IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 2, 2020, accepted May 28, 2020, date of publication June 1, 2020, date of current version June 11, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2999071

Adaptive Edge Detection Algorithm Based on Improved Grey Prediction Model

ZHEN ZHENG, BINGTING ZHA, HAILU YUAN, (Member, IEEE), YOUSHI XUCHEN, YANLIANG GAO, AND HE ZHANG
School of Mechanical Engineering, Nanjing University of Science and Technology, Nanjing 210094, China

Corresponding authors: Bingting Zha (zhabingting@njust.edu.cn) and He Zhang (hezhangz@mail.njust.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51709147, in part by the Central University Special Funding for Basic Scientific Research under Grant 309171B8805, in part by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYCX19_0262, and in part by the Key Basic Research Projects of Basic Strengthening Plan under Grant 2017-JCJQ-ZD-004.

ABSTRACT Existing edge detection algorithms suffer from inefficient edge localization, noise sensitivity, and/or relatively poor automatic detection capability. Contemporary edge detection algorithms can be improved by targeting these problems to help bolster their performance. Grey system theory can be used to resolve the small data and poor information issues in the local information of uncertain systems. An automatic edge detection algorithm was developed in this study based on a grey prediction model to remedy these problems. Noise characteristics in grey images are used to deploy a noise-filtering algorithm based on local features. A mask with twenty-four edge direction information points (345°) was established based on edge line texture features. By compressing the amplitude of the sequence, the randomly oscillated grey prediction sequence can be converted into a smooth, new sequence. The discrete grey model (1,1) (DGM(1,1)) was established based on this new grey prediction sequence to obtain the grey prediction maximum value. A grey prediction image with enhanced edges was obtained by replacing the pixel value in the original image with the maximum grey prediction value. A grey prediction subtraction image with edges separated from non-edge points was also obtained by subtracting the original image from the grey prediction image. The optimal separation threshold in the grey prediction subtraction image can be determined via the global adaptive threshold selection method. The neighborhood search method was then deployed to remove stray points and burrs from the image after the target was separated from the background, creating the final edge image. Experiments were performed on a computer-simulated phantom to find that both the subjective visual effects and objective evaluation criteria are better under the proposed method than several other competitive methods. The proposed edge detection algorithm shows excellent edge detection ability and is highly robust to noise, though the grey prediction model needs further improvement to optimize the run time.

INDEX TERMS Image processing, image edge detection, grey system theory, grey prediction model, adaptive threshold.

I. INTRODUCTION

The "edge" is the part of a given image where the gray value changes drastically [1], [2]. The edge represents rich and useful information, making edge detection an important pre-processing step in many artificial intelligence applications [3]–[5], [16].

Previously, several edge detection algorithms have been proposed. Some are based on the step nature of image edges and operate via gradient operators, such as those established by Roberts [6], Duda and Hart [7], and Canny [8], the best and most popular edge detection operator is the Canny, which performs the edge search according to the three criteria: signal-to-noise ratio, positioning performance, and uniqueness [8], [9]. It is still sensitive to noise, however, and may result in high-frequency signal loss or edge over-detection. Other attempts have been made to improve the traditional edge-detection-operator-based algorithms [10], [11]; for example, Rong *et al.* [12] introduced the concept of gravitational field intensity to replace the image gradient, which improved the noise-sensitivity performance and edge detection capability of the original Canny operator.

The associate editor coordinating the review of this manuscript and approving it for publication was Huazhu Fu.

The sub-pixel edge detection algorithm works by targeting sub-pixel points (i.e., points smaller than the image pixels) for enhanced noise-sensitivity and edge positioning capabilities. There are three types of sub-pixel edge detection algorithms currently in use: moment methods, interpolation methods, and fitting methods. Unfortunately, it is difficult to find a suitable fitting model. The massive amounts of calculation these methods require render them unsuitable for real-time applications [13], [14].

Rotation invariance is an important property of the moment. Moment-based methods can be used to effectively achieve edge detection. Types of moment include space moments, gray moments, and Zernike moments. It is usually necessary to select a large template to improve the edge positioning capability. The edges interfere with each other, however, when there are complex edges inside the template. The adaptive selection of the proper moment template is problematic for this reason [15]–[18].

Other techniques have been designed to strike a balance between edge localization ability and noise-sensitivity ability, such as wavelet transform, mathematical morphology, and neural network methods.

An appropriate multi-scale morphological operator must be selected based on mathematical morphology, at which point the edge positioning ability is strong and the amount of calculation is small, but the edges appear to be rough. To improve the stability of the edge-detector, an appropriate filter can be applied as a pre-operation [19]–[21]. Zheng *et al.* [19] designed an edge refinement method for this purpose which is effective to some extent but not robust to noise. Edge detection algorithms based on wavelet transform or neural network methods show better edge localization ability and noise-sensitivity performance, but their calculation burden is so large that the processing time is too long for practical application. In the automated artificial intelligence context, scenes must be imaged and the images processed in real time; it is difficult to obtain a priori knowledge of the target, so neural networks are not quick enough for effective edge detection [22]–[26].

Grey system theory was introduced in 1982 by Deng [27] in 1982 to solve the problems of uncertain systems with ''small data'' and ''poor information''. After nearly four decades of development, the grey system theory has been successfully applied in various fields such as economics, military, agriculture, geography, and control system [27]–[30]. A computer does not recognize whether a pixel is an edge in a local area when performing edge detection, so when it processes the image as the computer processes information: with ''incomplete global information''. Grey system theory is very well-suited to edge detection for this reason [2]. At present, the edge detection algorithms based on the grey system theory mainly use three types of grey models, namely grey correlation model, grey prediction model, and grey correlation entropy model.

Ma *et al.* [31] first applied grey relation analysis (GRA) to the edge detection context successfully in a 2003 study, but their technique required manual selection of thresholds and was sensitive to noise. Li *et al.* [32] proposed a surface image edge detection algorithm based on GRA in 2012, but their technique also requires manual selection of the appropriate separation threshold. Qiao *et al.* [33] proposed an automatic edge detection method based on GRA and human visual characteristics in 2016; the detection effect is better than traditional edge detection operators, but the edge continuity is poor and it is prone to over-detection.

Zhang *et al.* [34] established the grey relation entropy theory based on the theory of Shannon entropy in 1996. Zheng *et al.* [2] proposed an automatic edge detection algorithm based on grey entropy in 2019 which does have excellent edge detection performance, but is prone to double edges and has an overly time-consuming noise filtering algorithm.

He *et al.* [35] first utilized the grey prediction model for edge detection in 2005; the algorithm can effectively detect the edges of a given image but it is sensitive to noise. Xie *et al.* [36] proposed a micro-fiber edge detection algorithm based on the traditional grey prediction model (GM (1,1)) and Niblack model in 2016. A strong and weak edge connection algorithm provides good edge continuity, but the method is prone to edge over-detection for complex images and is not robust to salt and pepper noise.

An adaptive edge detection algorithm based on an improved grey prediction model was developed in this study. The structure of this paper is as follows. Section II introduces related work on grey prediction models and grey prediction sequence selection. Section III presents an adaptive edge detection algorithm based on an improved grey prediction model. Section IV reports a series of edge detection experiments we performed on original images and noisy images to test the proposed method, as well as the disadvantages and future research directions related to the proposed method. Section V provides a brief summary and conclusion.

II. RELATED WORK

A. IMPROVED GREY PREDICTION MODEL

The grey prediction model is an important branch of grey system theory. Its purpose is to predict data that is irregular but changes monotonically. The GM (1,1) model is the fundamental and most widely used grey prediction model; many prediction models have been derived based on the original GM (1,1) such as the mean GM (1,1) model (EGM), the original difference GM (1,1) model (ODGM), mean difference model (EDGM), discrete GM (1,1) model (DGM), and grey Verhulst model [37].

The original form of the GM (1,1) model is essentially a difference equation [37], [38]. Because the gray values in the 3 × 3 neighborhood window are discrete data and non-homogeneous exponential sequences, the algorithm proposed in this paper uses the DGM (1,1) model [39]. If the gray values of the pixel in the image neighborhood are very close, the development coefficient of the grey prediction model will be very close to zero; this causes the denominator of the time response sequence of the GM (1,1) model to appear to be zero. The gray values of the pixels in the neighborhood of the image are random within 0-255, and likely have a series of oscillating changes [38], but the reduction formula of the DGM (1,1) model is a strictly homogeneous exponential function [39]. Thus, the predicted data cannot be combined with the oscillating characteristics of the original data and DGM (1,1) model application directly results in substantial error. The original data must be improved using a smoothing operator to meet the requirements of the grey prediction model [38].

The grey prediction model uses a small amount of valid data and grey uncertainty data to predict future data through the accumulation of a sequence. A grey prediction model with a sequence of random oscillations is usually established in the form of an exponential with a monotonical increase. However, the simulated data calculated by the exponential expression does not conform to the original random oscillation sequence, resulting in poor prediction accuracy. It is necessary to compress the amplitude (the difference between the maximum value and the minimum value of the sequence) of the original random oscillation sequence to smooth the original sequence and improve the prediction accuracy [40].

Assume the original prediction data sequence is $X^{(0)} = \left\{x^{(0)}(k) \,|k = 0, 1, \ldots, n\right\}$, and $x^{(0)}(k) \geq 0$, the maximum difference of data in the original prediction data sequence is $Df = X_{max} - X_{min}$, where,

$$\begin{cases} X_{max} = \max\{x(k)|k = 1, 2, \cdots, n\} \\ X_{min} = \min\{x(k)|k = 1, 2, \cdots, n\} \end{cases} \tag{1}$$

The proposed smoothing operator is:

$$x^{(0)}(k)\, d = \frac{(x^{(0)}(k) + 3Df) + (x^{(0)}(k+1) + 3Df)}{8} \tag{2}$$

This operator can compress the amplitude of the data sequence, which effectively improves the smoothness of the modeling sequence.

Proof: The prediction sequence after the smoothing operation is $X^{(0)}D = \{x^{(0)}(k)\, d|k = 0, 1, \cdots, n$

If

$$\begin{cases} XD_{max} = x^{(0)}(k_1)\, d = \max\{x(k)|k = 1, 2, \cdots, n\} \\ XD_{min} = x^{(0)}(k_2)\, d = \min\{x(k)|k = 1, 2, \cdots, n\} \end{cases}$$

and the maximum difference of data in the sequence is

$$TfD = XD_{max} - XD_{min}$$

then substitution into Eq. (2) yields:

$$\begin{aligned} TfD &= \frac{(x^{(0)}(k_1) + x^{(0)}(k_1+1)) - (x^{(0)}(k_2) + x^{(0)}(k_2+1))}{8} \\ &\leq \frac{2Df}{8} && (3) \\ Df &\geq 4TfD && (4) \end{aligned}$$

Therefore, the smoothing operator (Eq. (2)) can compress the amplitude of the data sequence, which effectively smooths the modeling sequence.

The predicted data is accumulated to force a monotonical increase and minimize randomness. The accumulated sequence is $Y^{(1)} = \{y^{(1)}(k)\,|k = 0, 1, \cdots, n$, where $y^{(1)}(k) = \sum_{i=1}^{k} y^{(0)}(k)n, k = 0, 1, \cdots, n$, and $\hat{\beta} = [\beta_1, \beta_{2,}]^T$ is the parameter list. The DGM (1,1) model is established as $y^{(1)}(k+1) = \beta_1 y^{(1)}(k) + \beta_2$. $\hat{\beta} = (B^T B)^{-1} B^T Y$ is determined via least squares method [38], where

$$y^{(0)}(k) = x^{(0)}(k)d \tag{5}$$

$$Y = \begin{bmatrix} y^{(1)}(2) \\ y^{(1)}(3) \\ \vdots \\ y^{(1)}(n) \end{bmatrix}, \quad B = \begin{bmatrix} y^{(1)}(2) & 1 \\ y^{(1)}(3) & 1 \\ \vdots & \vdots \\ y^{(1)}(n) & 1 \end{bmatrix} \tag{6}$$

The discrete grey prediction model DGM (1,1) is obtained as follows:

$$\begin{aligned} \hat{y}^{(1)}(k) &= y^{(1)}(1) = x^{(0)}(1)\, d \\ \hat{y}^{(1)}(k+1) &= \beta_1^k y^{(0)}(1) + \frac{(1-\beta_1^k)\beta_2}{1-\beta_1}, \quad k = 1, 2, \cdots, n \end{aligned} \tag{7}$$

After subtracting the data, the predicted data is:

$$\hat{y}^{(0)}(k+1) = \beta_1^{k-1}(y^{(0)}(1)(\beta_1 - 1) + \beta_2); \quad k = 1, 2, \cdots, n \tag{8}$$

According to Eq. (5) and Eq. (8),

$$\begin{aligned} x^{(0)}(k) &= 8y^{(0)}(k-1) - x^{(0)}(k-1) - 6Df \\ x^{(0)}(k-1) &= 8y^{(0)}(k-2) - x^{(0)}(k-2) - 6Df \\ &\vdots \\ x^{(0)}(3) &= 8y^{(0)}(2) - x^{(0)}(2) - 6Df \end{aligned} \tag{9}$$

Iterative calculation of Eq. (9) yields

$$\begin{aligned} &x^{(0)}(1) = const, \quad k = 1 \\ &x^{(0)}(2) = 8y^{(0)}(1) - x^{(0)}(1) - 6Df, \quad k = 2 \\ &x^{(0)}(k) = 8y^{(0)}(k-1) - y^{(0)}(k-1) + y^{(0)}(k-2) - \cdots, \\ &\qquad k \geq 3 \end{aligned} \tag{10}$$

Let $C = x^{(0)}(2)$ be the initial value of the oscillation sequence prediction model, then

For $k > 2$, when k is odd, the predicted value can be obtained according to Eq. (8) and Eq. (10).

$$x^{(0)}(k) = \frac{8\beta_1^{k-3}(y^{(0)}(1)(\beta_1 - 1) + \beta_2)(1 - (-1)^{k-2}\beta_1^{2-k})}{1 + \beta_1^{-1}} - C - 6T$$

When k is even, the predicted value can be obtained according to Eq. (8) and Eq. (10).

$$x^{(0)}(k) = \frac{8\beta_1^{k-3}(y^{(0)}(1)(\beta_1 - 1) + \beta_2)(1 - \beta_1^{2-k})}{1 + \beta_1^{-1}} + C$$

Let $F = \frac{8y^{(0)}(1)(\beta_1 - 1)+\beta_2}{1+\beta_1^{-1}} = const$, then Eq. (11) is written by combining odd and even cases; this is the improved grey prediction model used in this study.

$$x^{(0)}(k) = F\beta_1^{k-3}\left(1-(-1)^{k-2}\beta_1^{2-k}\right)+(-1)^k C + 6(1-(-1)^k)Df \qquad (11)$$

B. GREY PREDICTION SEQUENCE SELECTION

According to Eq. (11) (Section II(A)), the grey prediction model based on the grey prediction sequence must pass through the initial pixel $x^{(0)}(1)$. The prediction results differ when different initial points of the grey prediction model are selected for the grey prediction sequence. For example, if the grey prediction sequence $S_1 = [30, 20, 163, 50, 200]$, the predicted value of S_1 is $F_1 = 197$, and $\Delta F_1 = 197 - 163 = 34$. The sequence $S_2 = [133, 50, 163, 20, 40]$ can be obtained by changing the initial position of S1; the predicted value is $F_2 = 109$, and $\Delta F_2 = 109 - 163 = -54 < \Delta \text{F}_1$. Initial points should be selected on both sides of the grey prediction consequence to minimize any prediction error in the grey prediction model.

The grey prediction model generally requires at least 4 data points for modeling. Here, we input 5 data points to the grey prediction model. To suit the 24 edge texture directions, 24 types of grey prediction sequences were designed in the 5×5 neighborhood window representing 0°, 15°, 30°, 45°, $\ldots$, 180°, 195°, $\ldots$, 330°, and 345°. The mask is shown in Fig. 1. The corresponding prediction sequences are $0 - 345°$. Red pixels are the initial points of the grey prediction sequence, black pixels are the transition points of the grey prediction sequence, and green pixels are the end points of the grey prediction sequence.

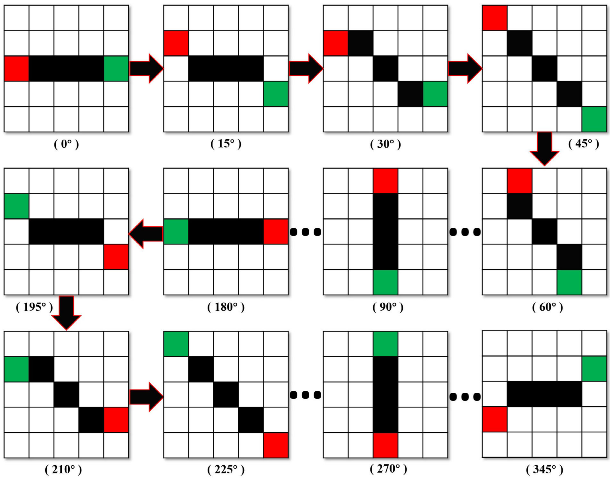


FIGURE 1. Grey prediction sequence mask.

These sequences represent 24 distinct edge texture directions, which contain more information than single-direction grey model sequences. The prediction results are thus more accurate and the model is robust to low noise.

III. ADAPTIVE EDGE DETECTION ALGORITHM DESIGN

A. NOISE FILTERING ALGORITHM

Figures are easily polluted by noise during generation and transmission, so denoising should be performed prior to edge detection. Salt and pepper noise is the most common and has a relatively large impact on image quality [40]–[42]. In this study, denoising filtering pre-processing was performed for salt and pepper noise specifically.

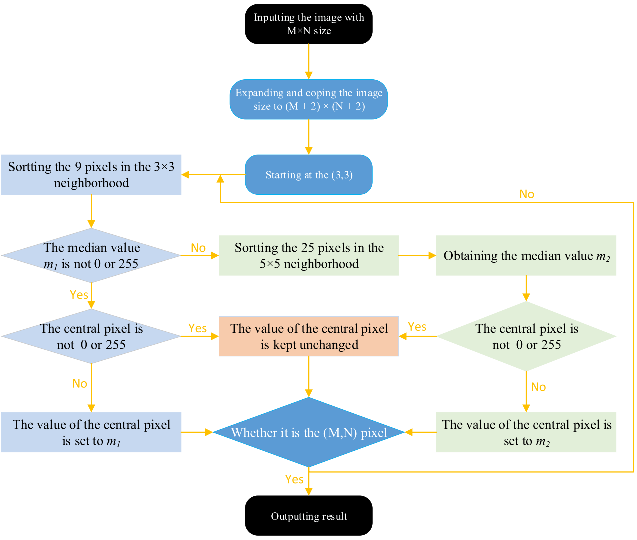

Inputting the image with M×N size
Expanding and coping the image size to (M + 2) × (N + 2)
Starting at the (3,3)
Sorting the 9 pixels in the 3×3 neighborhood
No
The median value m_1 is not 0 or 255
No
Sorting the 25 pixels in the 5×5 neighborhood
Obtaining the median value m_2
Yes
The central pixel is not 0 or 255
Yes
The value of the central pixel is kept unchanged
Yes
The central pixel is not 0 or 255
No
No
The value of the central pixel is set to m_1
Whether it is the (M,N) pixel
The value of the central pixel is set to m_2
Yes
Outputting result

FIGURE 2. Processing flow diagram of the proposed denoising algorithm.

Salt and pepper noise is an extreme value in the image which appears directly to the human eye as black or white. In greyscale images, the values of salt and pepper pixels are 0 or 255. A fast denoising algorithm was designed in this study based on this feature. The processing flow diagram of the proposed fast denoising algorithm is shown in Fig. 2.

(1) Sort the 9 pixels in the 3×3 neighborhood. If the median value m_1 of the 9 pixels in 3×3 neighborhood is not 0 or 255, then there are fewer than 5 noise points in the 3×3 neighborhood window.

(2) If the central pixel is not equal to 0 or 255, then the value of the pixel is kept unchanged; if the central pixel point is equal to 0 or 255, the pixel is judged as a noise point and its value is set to m_1.

(3) The value of salt and pepper noise is 0 or 255. When the median of the sorted sequence is 0 or 255 in the 3×3 neighborhood, it means that at least 5 pixels are 0 or 255 in 3×3 neighborhood and it is likely that the 3×3 neighborhood is seriously polluted by salt and pepper noise. Extending the processing neighborhood window to 5×5 and sortting the 25 pixels in the 5×5 neighborhood windows to determine the median value of the sequence m_2.

(4) If the center pixel point is 0 or 255, then the center pixel point is judged as a noise point and its value is set to m_2. If the central pixel point is not 0 or 255, then the central pixel point is a non-noise point and its value remains unchanged.

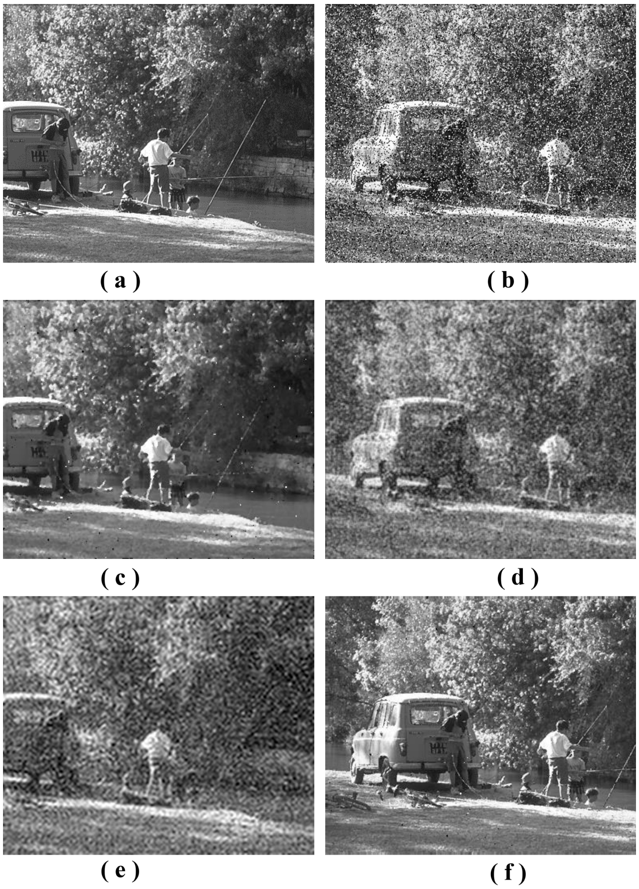


FIGURE 3. Filtering effect of various algorithms. (a) Original image, (b) image with 0.3 salt and pepper noise, (c) median filter method with 3 × 3 template, (d) mean filtering method with 3 × 3 template, (e) DCT method, (f) proposed method.

To test the proposed algorithm, we compared it against other commonly utilized quick algorithms as shown in Fig. 3. Figure 3(a) shows the original image and Fig. 3(b) shows the image with 0.3 salt and pepper noise; Fig. 3(c) shows the image processed by the median filter method with a 3 × 3 template, Fig. 3(d) shows the image processed by the mean filtering method with a 3 × 3 template, Fig. 3(e) shows the image processed by the Discrete Cosine Transform (DCT) filtering method [43], and Fig. 3(f) shows the image filtered by the proposed algorithm.

As shown in Fig. 3, the image becomes blurred when 0.3 salt and pepper noise is added. The average filtering method and DCT filtering method have the worst filtering effect; the image remains blurry and crowded with noise. After the median filtering process, a portion of the image is still blurred, but the noise processing effect is better than average filtering or DCT. The proposed filtering algorithm shows the best denoising effect and most intact original image among the algorithms tested in this study.

Peak Signal to Noise Ratio (PSNR) is an important index for evaluating image filtering effects [2]. With salt and pepper noise density of 0.05, 0.1, 0.3, and 0.5, the PSNR values of the images shown in Figs. 3(c)-(f) are listed in Table 1.

TABLE 1. Comparison of various filtering algorithm PSNR values.

Noise ratio	Median filter method	Mean filtering method	DCT method	Proposed filtering algorithm
0.05	31.6975	30.5767	30.0223	41.4179
0.1	31.5913	30.1688	29.7829	39.5443
0.3	31.1624	29.5194	29.5232	35.4089
0.6	29.6570	29.3668	29.3988	31.2945

The proposed algorithm has higher PSNR than the algorithms we tested, especially when processing low-noise images.

B. DATA PREPROCESSING

The relative simulation percentage error (*RSPE*) calculated by Eq. (12) of the grey prediction sequence is shown below. RSPE is infinite when the gray value $x^{(0)}\,(1)$ is zero, so this must be carefully avoided. In addition, all grey prediction sequence data should satisfy the ratio check condition. The gray values of the image pixels can be preprocessed to accomplish this:

$$RSPE\,(k)=\left|\frac{\hat{x}^{(0)}\,(k)-x^{(0)}\,(k)}{x^{(0)}\,(k)}\times 100\%\right|,\quad k=1,2,\cdots,n \tag{12}$$

where $x^{(0)}\,(k)$is the original value, $\hat{x}^{(0)}\,(k)$ is the predicted value, and n is the length of the grey prediction sequence.

The specific data transformation method involves performing a positive offset with respect to the data to prevent a 0 value [44]. The offset is p. Logarithmic processing is performed to satisfy extreme ratio conditions as shown in Eq. (13).

$$z^{(0)}\,(k)=\ln(x^{(0)}\,(k)+p) \tag{13}$$

The value range of $(x^{(0)}\,(k)$ is [0,255]. As $z(t) = \ln(t)$ is a monotonically increasing function, the grey prediction sequence satisfies the extreme ratio condition:

The value range of $z^{(0)}\,(k)$ is $(e^{-\frac{2}{1+n}}, e^{\frac{2}{1+n}})$ [44], so Eq. (14) holds.

$$\frac{\ln(255+p)}{\ln p}=e^{\frac{2}{1+n}} \tag{14}$$

Five data points were used to form a grey prediction sequence in this study, so $n = 5$ was plugged into Eq. (14) to obtain a p of 61.94. The transformation model is thus:

$$z^{(0)}\,(k)=\ln(x^{(0)}\,(k)+61.94) \tag{15}$$

C. EDGE DETECTION ALGORITHM

The edge is where the gray level of the image changes drastically. When the grey prediction model uses severely fluctuating data, *RSPE* will be large. When changing from low to high, as shown in Fig. 1 (a), the predicted value results in an overestimation at the turning point of the trend. When changing from high to low, as shown in Fig. 1(b), the predicted value results in an underestimation at the turning point of the trend. However, when the grey prediction model

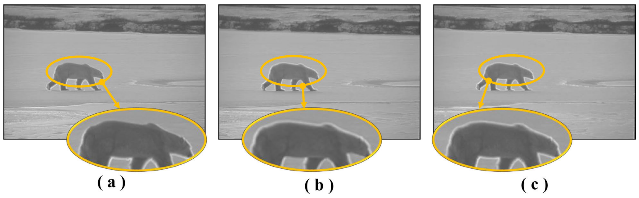


FIGURE 4. Edge enhancement images processed with different masks. (a) Grey prediction image using 12 prediction sequences (s_1 to s_{12}), (b) grey prediction image using 12 prediction sequences (s_{13} to s_{24}), (c) grey prediction image using 24 sequences (s_1 to s_{24}).

uses data from non-edge areas, the data does not fluctuate greatly, *RSPE* is small, and the predicted value is stable. In this study, once we selected an appropriate initial grey prediction sequence value and obtained the prediction image via the grey prediction model, we effectively obtained an edge image by subtracting the original image from the prediction image [36], [45].

To prevent predicted value underestimation (Fig. 1(b)), we used the 24 edge line directions described in Section II (B) to establish 24 grey prediction models to obtain 24 grey prediction values. The texture direction of the 24 edges is $(0 - 345°)$, so *RSPE* is largest when the maximum value is selected, at which point the edges are enhanced. However, for the non-edge area, the 24 predicted values are almost the same as the actual values (i.e., present no such enhancement).

Figure 4(a) shows the grey prediction image using 12 prediction sequences$(0 - 165°)$, Fig. 4(b) shows the grey prediction image using 12 prediction sequences $(180 - 345°)$, and Fig. 4(c) shows the grey prediction image using 24 sequences $(0 - 345°)$. The edges of the bears in the three images have mostly white edge lines while the original image has black edge lines, indicating that the edges have been enhanced. Among the three images, some do now show enhanced edges from top to bottom (such as the back of the bear); in Fig. 4(b), the edges are not enhanced from the bottom to the top (such as the abdomen of the bear) though the edges in both directions in Fig. 4(c) are enhanced. In effect, the proposed method results in significant edge enhancement.

The processing region of the algorithm is a 5 × 5 neighborhood window. The grey prediction model cannot predict the two rows above the image, two rows below the image, two columns on the left of the image and two columns on the right of the image, so we copied the pixels of these regions of the image to the original image pixel value.

According to the original image I_0, the grey prediction image I_p can be obtained by the grey prediction model. We obtained an edge direction data matrix *Label* corresponding to the maximum prediction value. Because the proposed grey prediction model enhances edge points, the value of these edge points in the subtraction image I_d is larger than the value of non-edge points. I_d is calculated as follows:

$$I_d = I_p - I_o \tag{16}$$

The optimal separation threshold *Th* is obtained by iterative method to separate the target from the background [46]. The adaptive threshold selection method used in this study is operated as follows.

(1) The initial value of the separation threshold $T_1 = \frac{T_{max}+T_{min}}{2}$ is determined where T_{max} is the maximum value of I_d and T_{min} is the minimum value of I_d.

(2) T_1 divides I_d into part A and part B. The average greyscale $\mu_1$of part A and the average greyscale μ_2 of part B are calculated to obtain a new separation threshold $T_2 = \frac{\mu_1+\mu_2}{2}$.

(3) The values of T_1 and T_2 are compared and the resolution is set to T_0. If $|T_1 - T_2| < T_0$, T_2 is the optimal separation threshold *Th*; otherwise, T_1 is replaced with T_2 and Steps (2) and (3) are repeated until the condition $|T_1 - T_2| < T_0$ is met.

Finally, all pixels are updated and the edge image is obtained according to Eq. (17). If the pixels are determined to be edge points, their values are set to 1 and *edge* (*i, j*) is updated to 1 in the edge image; if the pixels are non-edge points, their values are set to 0 and *edge* (*i, j*) is updated to 0.

$$edge\ (i,j) = \begin{cases} 1, & I_d(i,j) \geq Th \\ 0, & I_d(i,j) < Th \end{cases} \tag{17}$$

When $edge(i,j) = 0$, if there are points $edge(i + k_1, j + k_2), k_1 = \pm 1, 0, k_2 = \pm 1, 0$ in the 3 × 3 neighborhood window and $Label(i,j) = Label(i + k_1, j + k_2), k_1, k_2 = \pm 1, 0$, then $edge(i,j) = 1$ and the broken edges are connected. The gray value of the pixels connected to the edge line is then close to the edges and there is a ''burr'' on the edge processed by the above steps.

The deburring methods used in this study involve setting one pixel as a starting point and searching outwards for edge points in the 4-connected neighborhood. If there are edge points, the search continues outward. When there are less than 3 points found, it is judged as a ''burr'' and the starting point is deleted. In addition, the area pixels with fewer than 25 connections are deleted via 8-connected neighborhood search method. The small edges and noise are removed and the final edge image is obtained.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

We compared the proposed algorithm in this paper with state-of-the-art algorithms to validate its performance and to explore its flaws and potential room for improvement.

A. ACTUAL IMAGE EDGE DETECTION

In order to verify the performance of the proposed algorithm, we compared it to other state-of-the-art algorithms including the improved Canny algorithm proposed by Rong *et al.* [12], the traditional GM (1,1) edge detection algorithm (threshold $th = 5$) proposed by Wan *et al.* [38], the traditional GRA edge detection algorithm (threshold $k = 0.1$) proposed by Li *et al.* [32], the morphological method proposed by Zheng *et al.* [19], and the Zernike moment method proposed by Peng *et al.* [16]. Fig. 5 shows the edges detected by various methods on the Berkeley Segmentation Data Set (BSDS500) [47], [48].

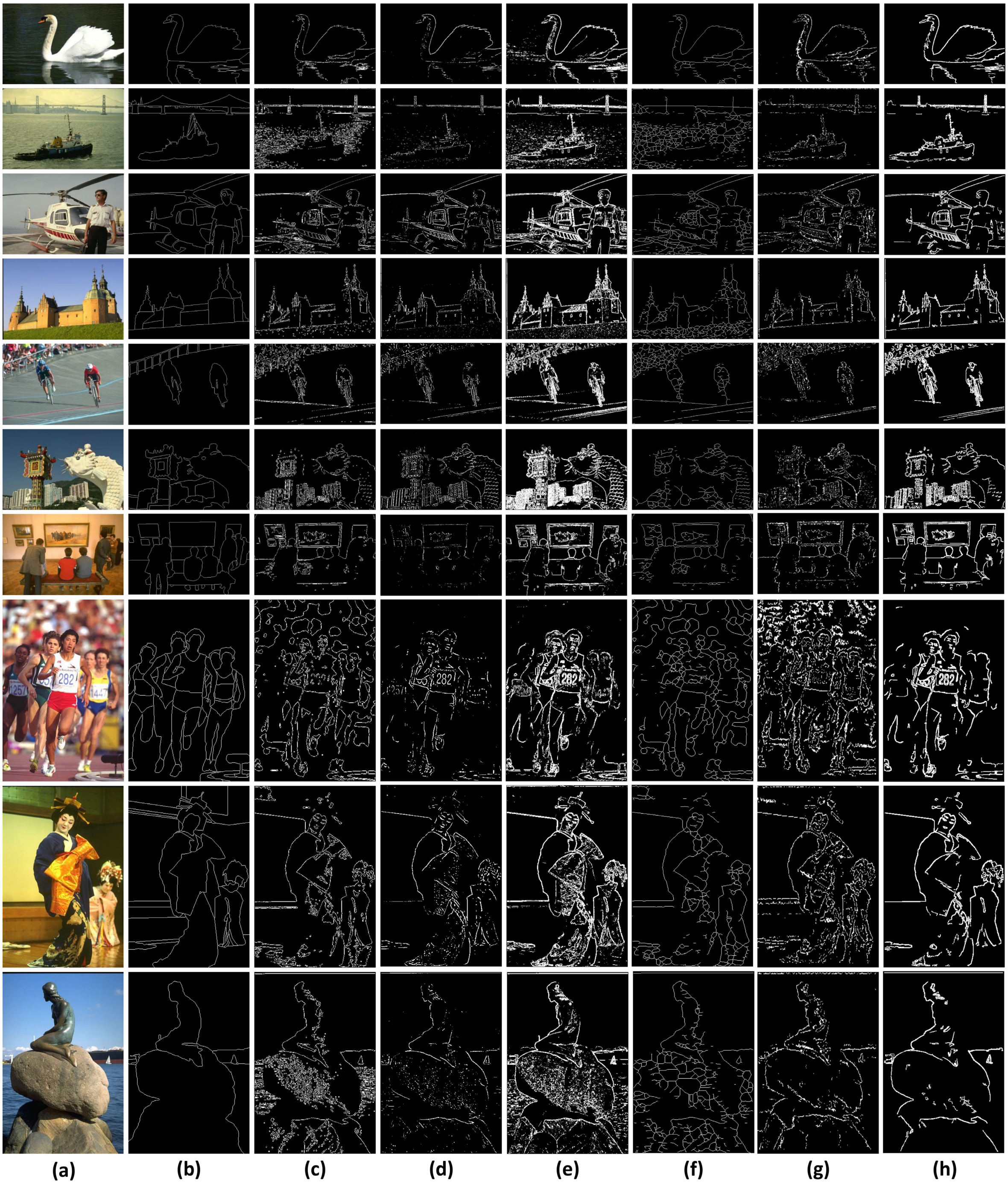


FIGURE 5. Image edge detection results. (a) Original image, (b) ground truth by human subjects, (c) edge produced by RONG+Canny, (d) edge produced by WAN+GM (1,1), (e) edge produced by LI+GRA, (f) edge produced by ZHENG+Morphology, (g) edge produced by PENG+Moment, (h) edge produced by proposed algorithm.

As shown in Fig. 5, for simple images (such as those in the first and fourth rows), edges were detected successfully and accurately by various methods. For images with relatively complex scenes (such as those in the third, fifth, ninth, and tenth rows), CANNY+RONG, LI+GRA and ZHENG+ Morphology all detected unneeded background areas, WAN+GM(1,1) missed some edges, and PENG+Zenike detected discontinuous edges and background points. These problems are more serious when the scene is complex or when the grey level of the image does not markedly change (such as the images in the second, seventh, and eighth rows). As shown in Fig. 5(h), the proposed algorithm accurately detects most of the edges; edges are rarely over-detected and edge continuity performance is excellent. This indicates that the proposed method has better edge detection ability than the others we tested, though there some double edges do occur (similar to the LI+GRA).

We also compared the *F-Measure* results of various algorithms and the proposed algorithm as per the *precision* and *recall* information from Eq. (17). The edge detection method is more effective when its *F-Measure* is higher [51]. We chose $k = 2$ in this study to obtain the results on the on BSDS500 as shown in Fig. 6.

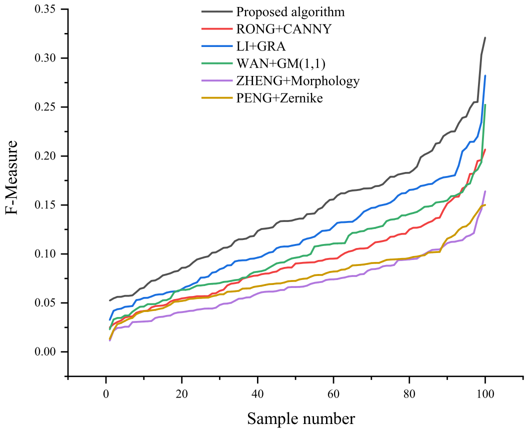


FIGURE 6. **F-measure on original image.**

As shown in Fig. 6, the *F-measure* of the proposed method are higher than other algorithms. In conclusion, the ability of edge detection of the proposed algorithm is better than other edge detection algorithm we compared.

Continuity is an important attribute of edges. In the image segmentation, target recognition, and other edge-based image processing contexts, the continuity of edges crucial significance – it directly affects the image segmentation effects and of target recognition results. The continuity of detected edges must be carefully evaluated. When there are many edge points and few continuous edges, the space range that each edge cross is large and edge continuity is strong. A continuity evaluation index based on the average number of pixels contained in the edge line was designed here accordingly:

$$\rho = \frac{\varepsilon_1 n_s}{\varepsilon_s n_1} \tag{18}$$

where ε_1 is the number of edge points, ε_2 is the number of edge points detected by the ground truth (Figs. 5(b)), n_1 is the number of edge lines, and n_2 is the number of edge lines detected by the ground truth (Figs. 5(b)).

We calculated a continuity evaluation index for the 10 figures in Fig. 5. A continuity evaluation index comparison is shown in Fig. 7. The continuity evaluation indexes of the proposed algorithm were consistently higher than the other algorithms we tested, which suggests that the continuity of the edges detected by the proposed algorithm is better than other compared state-of-the-art algorithms on original images.

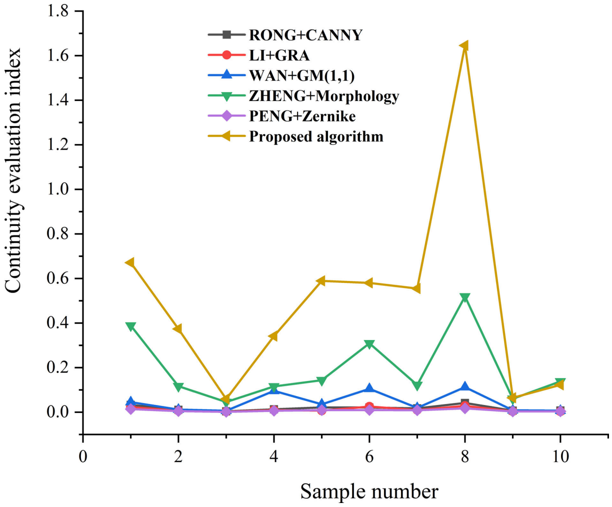


FIGURE 7. **Continuity evaluation index of various edge detection algorithms on original images.**

B. NOISE IMAGE EDGE DETECTION

We added salt and pepper noise (with density of 0.05) to the digital image used in Section IV(A) and verified the proposed algorithm by comparison to the other algorithms discussed above. The results are shown in Fig. 8.

Under the influence of high level (with density of 0.05) noise, as shown in Figs. 7(c)-(f), RONG+Canny, LI+GRA, WAN+GM (1,1), and ZHENG+Morphology were overwhelmed by noise. As shown in Fig. 7(g), PENG+Moment detected most of the edges but many discontinuous edges occurred. The edge detection effects shown in Fig. 7(h) are similar to Fig. 5(h), which indicates that the proposed algorithm is robust to noise.

The *F-Measure* curve of various edge detection algorithms is shown in Fig. 9. The *F-measure* values of the proposed algorithm are also consistently higher than the other algorithms we tested. These results suggest that the proposed algorithm outperforms the others for images that are polluted by noise. In addition, as shown in Fig. 10, the *F-measure* of the noisy images is very close to that of the original image, indicating that the proposed algorithm has strong noise-sensitivity ability.

A continuity evaluation index of various edge detection algorithms was obtained as shown in Fig. 11. The continuity evaluation index of the proposed algorithm is also consistently higher than those of the other algorithms we tested. These results suggest that the continuity of the edges detected by the proposed algorithm is better than other state-of-the-art algorithms on noisy images.

C. FUTURE RESEARCH

There are two edge lines located on the target and the background, respectively. The mask spans across these edge lines when predicting the points on them. As shown in Fig. 10, the proposed algorithm performs accurate detections regardless of whether the starting point is P1 or P2, as it effectively

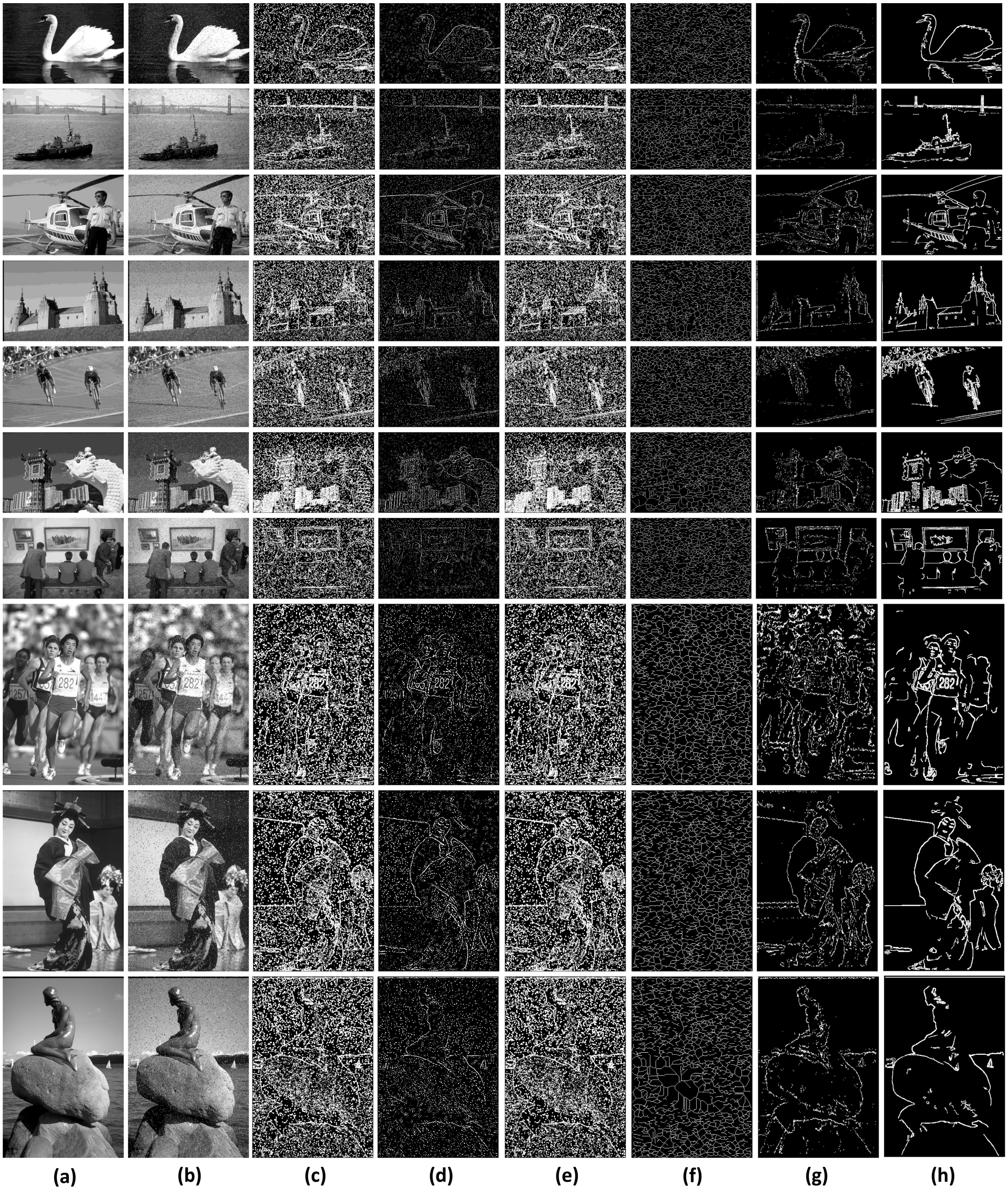


FIGURE 8. Noise-containing image edge detection results. (a) Original image, (b) ground truth by human subjects, (c) edge produced by RONG+Canny, (d) edge produced by WAN+GM (1,1), (e) edge produced by LI+GRA, (f) edge produced by ZHENG+Morphology, (g) edge produced by PENG+Moment, (h) edge produced by proposed algorithm.

enhances the edges. In other words, the proposed algorithm can detect double edge lines. Edges are usually are located on the target, however, the proposed algorithm still needs reworking to ensure it can detect double edges. The most effective approach to this is to first extract the region of interest (ROI) of the image, then determine whether the detected edge is on the ROI; if it is, keep it, otherwise, delete it. Many rapid ROI extraction methods are currently available [49].

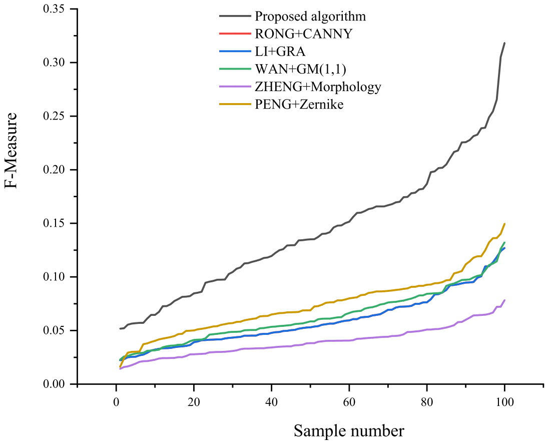


FIGURE 9. F-measure of noise-containing image edge detection.

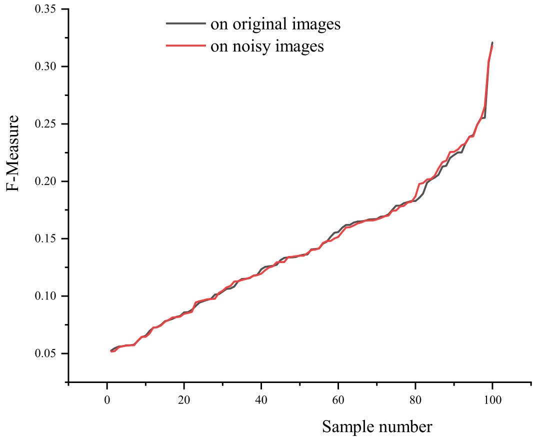


FIGURE 10. F-measure comparison between noisy and original images.

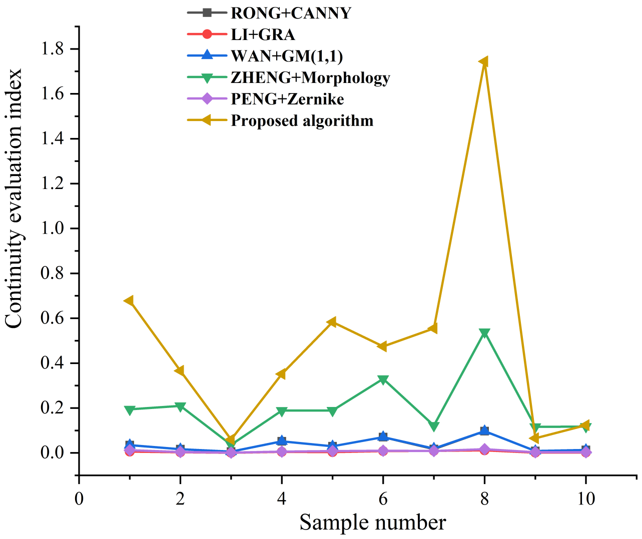


FIGURE 11. Continuity evaluation index of various edge detection algorithms on noisy images.

In the future, we plan to integrate ROI extraction into the proposed algorithm.

The threshold selection of the proposed algorithm is global, the edge detection capability for any areas in a given image with insufficient lighting is relatively poor (Fig. 13, where the light on the left side of the palace is weak) and may result in edge "leakage". We also plan to improve the threshold selection of the proposed algorithm in the future, such as by changing the global threshold selection to local threshold selection to reduce the impact of uneven lighting [50].

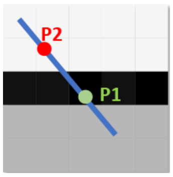


FIGURE 12. Double edge line schematics.

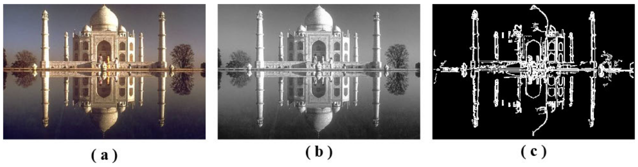


FIGURE 13. Edge detection image under uneven lighting conditions.

Running time is an important index for evaluating an algorithm. The algorithm has better performance when it takes less time to complete the same task as other algorithms. Big O notation reflects the time complexity of algorithm. The time complexity of each algorithm and the running time for edge detection on the image shown in Fig. 3 are listed in Table 2.

TABLE 2. Time complexity and running time of each algorithm.

	TIME COMPLEXITY	RUNNING TIME(/S)
RONG+CANNY	$O(n^2)$	0.971
LI+GRA	$O(n^2)$	2.343
WAN+GM(1,1)	$O(n^2)$	41.211
ZHENG+Morphology	$O(n^2)$	1.285
PENG+Zernike	$O(n^2)$	3.729
Proposed algorithm	$O(n^2)$	67.242

As shown in Table 2, the time complexity of the six algorithms tested in this study is similar. However, the proposed algorithm and WAN + GM (1,1) use the least squares parameter estimation in the grey prediction model, making them somewhat less efficient than the other four algorithms. In addition, a mask with 24 directions was used in this study to accurately obtain the edge direction – this further increases the running time of the proposed algorithm. In the future, the proposed algorithm must be optimized to adapt to online detection (e.g., by optimizing the mask to reduce the number of grey prediction operations, or optimizing the grey prediction model parameter estimation method) to accelerate the grey prediction operations.

The continuity of the edges detected by the proposed algorithm is better than the other algorithms tested here, but did result in some missed edges in the test. This problem intensifies when processing images with many weak edges, so the proposed algorithm should to be improved with an edge connecting algorithm. Existing edge connection technologies include morphology-based edge connection algorithms and gradient-based edge connection algorithms. The pixel edge trend can be obtained on several masks with direction information when obtaining the maximum predicted value at any pixel. When there is at least one edge point in the edge trend in the 3 × 3 neighborhood window, the point is determined to be an edge point and set to one.

V. CONCLUSION

This paper proposes a novel image edge detection algorithm based on the grey prediction model. An improved grey prediction model was established based on the original grey prediction model and the gray value of the pixels in the neighborhood window. A grey prediction sequence mask with 0-345 ° edge texture direction was then established. The adaptive threshold was selected via global iteration method, then desultory points and burrs were removed via neighborhood search method to complete the automatic adaptive edge detection of grey images.

We performed a series of experiments to test the proposed algorithm's edge detection ability and noise-sensitivity ability. The results show that it outperforms several other similar algorithms and can effectively perform automatic adaptive edge detection. It also showed excellent edge positioning ability and noise-sensitivity ability, which can provide reference for artificial intelligence algorithms such as target recognition and image recognition. However, further research is yet needed to overcome its lack of robustness to uneven illumination and double-edge problems.

REFERENCES

[1] H. Zhu, H. Fan, Z. Shu, Q. Yu, X. Zhao, and P. Gan, "Edge detection with chroma components of video frame based on local autocorrelation," *IEEE Access*, vol. 7, pp. 48543–48550, 2019.

[2] Z. Zheng, B. Zha, Y. Xuchen, H. Yuan, Y. Gao, and H. Zhang, "Adaptive edge detection algorithm based on grey entropy theory and textural features," *IEEE Access*, vol. 7, pp. 92943–92954, 2019.

[3] F. Han, B. Liu, J. Zhu, and B. Zhang, "Algorithm design for edge detection of high-speed moving target image under noisy environment," *Sensors*, vol. 19, no. 2, p. 343, Jan. 2019.

[4] S. Agrawal and B. K. Dean, "Edge detection algorithm for *Musca −−Domestica* inspired vision system," *IEEE Sensors J.*, vol. 19, no. 22, pp. 10591–10599, Nov. 2019.

[5] X. Wang, "Laplacian operator-based edge detectors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 5, pp. 886–890, May 2007.

[6] L. G. Roberts, *Machine Perception of Three-Dimension Solids, Optimal and Electro-Optimal Information Processing*. Cambridge, MA, USA: MIT Press, 1965, pp. 99–197.

[7] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York, NY, USA: Wiley, 1973, pp. 271–272.

[8] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[9] X. Zhang and W. Zhang, "Application of new multi-scale edge fusion algorithm in structural edge extraction of aluminum foam," *IEEE Access*, vol. 8, pp. 15502–15517, 2020.

[10] A. K. Cherri and M. A. Karim, "Optical symbolic substitution: Edge detection using Prewitt, Sobel, and Roberts Operators," *Appl. Opt.*, vol. 28, no. 21, pp. 4644–4648, Nov. 1989.

[11] H. Xu, X. Xu, and Y. Zuo, "Applying morphology to improve canny operator's image segmentation method," *J. Eng.*, vol. 2019, no. 23, pp. 8816–8819, Dec. 2019.

[12] W. Rong, Z. Li, W. Zhang, and L. Sun, "An improved canny edge detection algorithm," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Tianjing, China, Aug. 2014, pp. 577–582.

[13] J. Ye, G. Fu, and U. P. Poudel, "High-accuracy edge detection with blurred edge model," *Image Vis. Comput.*, vol. 23, no. 5, pp. 453–467, 2005.

[14] K. Jensen and D. Anastassiou, "Subpixel edge localization and the interpolation of still images," *IEEE Trans. Image Process.*, vol. 4, no. 3, pp. 285–295, Mar. 1995.

[15] F. Da and H. Zhang, "Sub-pixel edge detection based on an improved moment," *Image Vis. Comput.*, vol. 28, no. 12, pp. 1645–1658, Dec. 2010.

[16] S. H. Peng, W. Q. Su, X. Hu, C. H. Liu, Y. Wu, and H. Nam, "Subpixel edge detection based on edge gradient directional interpolation and Zernike moment," in *Proc. Int. Conf. Comput. Sci. Softw. Eng.*, May 2018, pp. 106–116.

[17] Y. H. Li, J. Huo, M. Yang, and G. Y. Zhang, "Algorithm of locating the sphere center imaging point based on novel edge model and Zernike moments for vision measurement," *J. Mod. Opt.*, vol. 66, no. 2, pp. 219–227, 2019.

[18] S. G. Vengurlekar, D. Jadhav, and S. Shinde, "Object detection and tracking using zernike moment," in *Proc. Int. Conf. Commun. Electron. Syst. (ICCES)*, Coimbatore, India, Jul. 2019, pp. 12–17.

[19] Z. Zheng, B. Zha, H. Yuan, and Y. Xuchen, "Edge detection algorithm based on morphology and grey relation analysis," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Tianjin, China, Aug. 2019, pp. 945–950.

[20] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Trans. Image Process.*, vol. 2, no. 2, pp. 176–201, Apr. 1993.

[21] X. G. Fu and H. Jiang, "A multi-scale morphological algorithm for AFM micrograph edge detection," in *Proc. Int. Conf. Ind. Technol. Manage. Sci.*, Tianjin, China, 2015, pp. 942–945.

[22] Z. Zhang, Y. Liu, T. Liu, Y. Li, and W. Ye, "Edge detection algorithm of a symmetric difference kernel SAR image based on the GAN network model," *Symmetry*, vol. 11, no. 4, p. 557, Apr. 2019.

[23] S. Pala, S. Jayan, and D. G. Kurup, "An accurate UWB based localization system using modified leading edge detection algorithm," *Ad Hoc Netw.*, vol. 97, Feb. 2020, Art. no. 102017.

[24] P. A. Flores-Vidal, P. Olaso, D. Gómez, and C. Guada, "A new edge detection method based on global evaluation using fuzzy clustering," *Soft Comput.*, vol. 23, no. 6, pp. 1809–1821, Mar. 2019.

[25] Z. Fu, S. Song, X. Wang, J. Li, and H.-M. Tai, "Imaging the topology of grounding grids based on wavelet edge detection," *IEEE Trans. Magn.*, vol. 54, no. 4, pp. 1–8, Apr. 2018.

[26] Z. Hu, "Edge detection method based on lifting B-Spline dyadic wavelet," *Int. J. Performability Eng.*, vol. 15, no. 5, pp. 1472–1481, 2019.

[27] J. L. Deng, "Grey information space," *J. Grey Syst.*, vol. 1, no. 2, pp. 103–117, 1985.

[28] S. Liu, Y. Yang, N. Xie, and J. Forrest, "New progress of grey system theory in the new millennium," *Grey Syst., Theory Appl.*, vol. 6, no. 1, pp. 2–31, Feb. 2016.

[29] Z. Zheng, B. T. Zha, and H. Zhang, "Target recognition method of laser line scanning imaging fuze based on DHGF algorithm," *Chin. J. Lasers*, vol. 45, no. 7, pp. 147–154, 2018.

[30] Z. Zhou and N. Sang, "Image edge detection algorithm based on grey system theory," *Int. J. Digit. Content Technol. Appl.*, vol. 6, no. 23, pp. 782–789, Dec. 2012.

[31] M. Ma, Y. Y. Fan, S. Y. Xie, C. Y. Hao, and X. W. Li, "A novel algorithm of image edge detection based on gray system theory," *J. Image Graph.*, vol. 10, no. 8, pp. 35–38, 2003.

[32] G. Li, Y. Fang, and Y. L. Tong, "Surface image edge detection algorithm based on grey relational analysis," *Adv. Mater. Res.*, vol. 461, pp. 343–346, Feb. 2012.

[33] B. Qiao, L. Jin, and Y. Yang, "An adaptive algorithm for grey image edge detection based on grey correlation analysis," in *Proc. 12th Int. Conf. Comput. Intell. Secur. (CIS)*, Dec. 2016, pp. 470–474.

[34] Q. S. Zhang, X. J. Guo, and J. L. Deng, "Grey relation entropy method of grey relation analysis," *Syst. Eng. Theory Pract.*, vol. 8, no. 8, pp. 8–12, 1996.

[35] R. G. He, D. S. Huang, and J. B. Chen, "Image edge detection based on grey predicton model," *J. Northwestern Polytech, Univ.*, vol. 23, no. 1, pp. 15–18, Feb. 2005.

[36] S. Xie, Y. Xie, W. Yang, and G. Wang, "Edge detection using advanced grey prediction model," in *Proc. Int. Conf. Netw., Sens. Control (ICNSC)*, Chicago, IL, USA, Apr. 2010, pp. 486–490.

[37] Q. Wang, T. Wang, and K. Zhang, "Image edge detection based on the grey prediction model and discrete wavelet transform," *Kybernetes*, vol. 41, no. 5/6, pp. 643–654, Jun. 2012.

[38] Y. Wan, Z. L. Shi, and S. L. Sun, "Development of micro fiber edge detection system based on grey system theory," in *Proc. 3rd Int. Conf. Wireless Commun. Sensor Netw.*, Wuhan, China, Dec. 2016, pp. 496–499.

[39] N. M. Xie and S. F. Liu, "Discrete GM (1, 1) and mechanism of grey forecasting model," *Syst. Eng.-Theory Pract.*, vol. 1, pp. 93–99, Jan. 2005.

[40] B. Zeng and S. F. Liu, "Prediction model of stochastic oscillation sequence based on amplitude compression," *Syst. Eng.–Theory Pract.*, vol. 32, no. 11, pp. 2493–2497.

[41] B. Zeng, S. F. Liu, and N. M. Xie, "Prediction model of interval grey number based on DGM (1, 1)," *J. Syst. Eng. Electron.*, vol. 21, no. 4, pp. 598–603, Aug. 2010.

[42] J. Barba, H. Jeanty, P. Fenster, and J. Gil, "The use of local entropy measures in edge detection for cytological image analysis," *J. Microsc.*, vol. 156, no. 1, pp. 125–134, Oct. 1989.

[43] X. Zhang, H. Liao, X. Du, and B. Xu, "A fast hybrid noise filtering algorithm based on median-mean," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Aug. 2018, pp. 2120–2125.

[44] M. E. Yüksel, "Edge detection in noisy images by neuro-fuzzy processing," *AEU-Int. J. Electron. Commun.*, vol. 61, no. 2, pp. 82–89, Feb. 2007.

[45] J. Pan, X. Yang, H. Cai, and B. Mu, "Image noise smoothing using a modified Kalman filter," *Neurocomputing*, vol. 173, pp. 1625–1629, Jan. 2016.

[46] N. Fukushima, Y. Kawasaki, and Y. Maeda, "Accelerating redundant DCT filtering for deblurring and denoising," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 4175–4179.

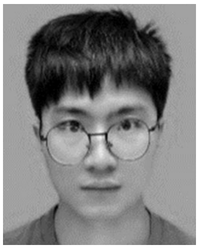
[47] M. Fornasier and H. Rauhut, "Iterative thresholding algorithms," *Appl. Comput. Harmon. Anal.*, vol. 25, no. 2, pp. 187–208, Sep. 2008.

[48] V. Premachandran, B. Bonev, X. Lian, and A. Yuille, "PASCAL boundaries: A semantic boundary dataset with a deep semantic boundary detector," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Santa Rosa, CA, USA, Mar. 2017, pp. 73–81.

[49] M. Borsotti, P. Campadelli, and R. Schettini, "Quantitative evaluation of color image segmentation results," *Pattern Recognit. Lett.*, vol. 19, no. 8, pp. 741–747, Jun. 1998.

[50] L. Ma, B. Du, H. Chen, and N. Q. Soomro, "Region-of-interest detection via superpixel-to-pixel saliency analysis for remote sensing image," *IEEE Geosci. Remote Sens. Lett.*, vol. 13, no. 12, pp. 1752–1756, Dec. 2016.

[51] Z. Ye, H. Chen, W. Liu, and J. Zhang, "Automatic threshold selection based on particle swarm optimization algorithm," in *Proc. Int. Conf. Intell. Comput. Technol. Autom. (ICICTA)*, Hunan, CHINA, Oct. 2008, pp. 36–39.

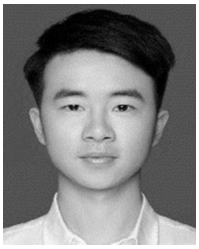


ZHEN ZHENG was born in Hubei, China. He is currently pursuing the Ph.D. degree in mechanical engineering with the Nanjing University of Science and Technology (NJUST), Nanjing, China. His current research interests include laser detection, laser target recognition, and image processing.

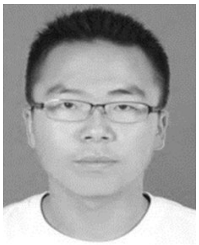
BINGTING ZHA was born in Jiangxi, China. She received the Ph.D. degree in armament science and technology from the Nanjing University of Science and Technology (NJUST), Nanjing, China, in 2015.

She is currently an Associate Professor with the School of Mechanical Engineering, NJUST. Her current research interests include underwater laser detecting and laser applications.

HAILU YUAN (Member, IEEE) was born in Shanxi, China. She is currently pursuing the Ph.D. degree in armament science and technology with the Nanjing University of Science and Technology (NJUST), Nanjing, China. Her current research interests include underwater laser scanning imaging technology and laser target detection.



YOUSHI XUCHEN was born in Zhejiang, China. He is currently pursuing the Ph.D. degree in armament science and technology with the Nanjing University of Science and Technology (NJUST), Nanjing, China. His current research interests include laser detection and laser anti-dust interference.



YANLIANG GAO was born in Hebei, China. He is currently pursuing the M.S. degree in armament science and technology with the Nanjing University of Science and Technology (NJUST), Nanjing, China. His current research interests include laser detecting and laser applications.

HE ZHANG was born in Henan, China. He received the Ph.D. degree in measurement technology and instruments from the Nanjing University of Aeronautics and Astronautics, Nanjing, China.

He is currently a Professor with the School of Mechanical Engineering, Nanjing University of Science and Technology (NJUST), China. He is the Director of the Institute of Mechanical and Electrical Engineering, NJUST, and the Associate Director of the ZNDY National Defense Key Laboratory. His current research interests include mechatronics and weapon system applications. He is an Editorial Board Member of the *Journal of Detection and Its Control*.