A Method for Identifying Flaws on Product Surface Based on Spatial Connectivity Domain

QUANYOU ZHANG[1,2], YONG FENG[1], BAO-HUA QIANG[3,4], YAOHUI LI[2], AND QIONGJIE KOU[2]

[1]College of Computer Science, Chongqing University, Chongqing, Shapingba District 400044, China
[2]College of International Education, Xuchang University, Xuchang, Weidu 461000, China
[3]Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China
[4]Guangxi Key Laboratory of Optoelectronic Information Processing, Guilin University of Electronic Technology, Guilin 541004, China

Corresponding author: Yong Feng (zimican1s@163.com)

ABSTRACT In this paper, we exploit a method for identifying flaws on product surface based on spatial connectivity domain. A number of algorithms for detecting local features exist that were established to enhance the efficiency and accuracy of identifying interest features, such as AKAZE, BFSIFT, BRIEF, BRISK, ORB, SURF, SIFT and PCA-SIFT algorithm. But the data of flaws on product surface which is similar and consistent with the background intensity became a dilemma to detect the feature of image. In terms of identifying flaws on product surface, the above algorithms are not effective and accurate. Our aim is to enhance the accuracy of detecting the feature of flaws on product surface, so that the product with flaws could be accurately identified in industrial production. Therefore, we propose a method to identify flaws on product surface based on spatial connectivity domain. Compared with some other algorithms, such as the extracting texture algorithm, the detecting local feature algorithm and the identifying edge algorithm, our proposed method is more effective and accurate in detecting the local feature flaws on product surface of auto parts in automotive manufacturing factory.

INDEX TERMS Connectivity domain, feature, image, auto parts, flaw.

I. INTRODUCTION

We present our latest results as a contribution to detect the local feature of product surface of auto parts in automotive manufacturing. We explore a method for identifying the feature of flaws on product surface based on spatial connectivity domain. The experiment result indicates that the proposed method is more effective than some other methods, such as the accelerated KAZE(AKAZE), binary robust invariant scalable key points (BRISK) and scale-invariant feature transform (SIFT), in distinguishing features of image of product surface for auto parts inspection.

There are currently a number of algorithms for identifying features of the image, such as AKAZE algorithm, BRISK algorithm, oriented features from accelerated seg-

ment test (FAST) algorithm, rotated binary robust independent elementary feature (BRIEF) algorithm, oriented FAST and rotated BRIEF(ORB) algorithm, speeded-up robust features(SURF) algorithm and SIFT algorithm [1]. In 2004, Lowe and Kong [2] proposed the SIFT algorithm. Although images were rotated, reflected and transformed, the feature of images could be detected via SIFT algorithm. However, the 128-vector descriptors need to be compared, which are complexly computed and matched. An algorithm named BRIEF [3] adopts the FAST keypoint detector and binary robust independent feature descriptor, which have yielded better results in real-time applications. Afterwards, an algorithm named ORB [4] utilizes the FAST and BRIEF algorithm to detect features of image. It performs well even with noise, rotation or affine transform. In 2011 a extracting feature algorithm named BRISK [5] was proposed at the International Conference on Computer Vision (ICCV 2011). It is

also a binary feature description algorithm with characters of rotation-invariant, scale-invariant, better robustness, etc. In image registration, BRISK algorithm is faster than SIFT algorithm, and it is also suitable for matching the large fuzzy image. In 2012 a more stable feature detection algorithm named KAZE was proposed at the European Conference on Computer Vision (ECCV). The algorithm constructs a stable nonlinear spatial scale [6]. It solved the problem of boundary blur and detail loss in image processing. KAZE algorithm [7] is a novel feature detection and description method which is invariant to rotation, scale, limited affine transform. One year later, the accelerated version of KAZE algorithm named AKAZE [8] was proposed. It adopts sparse feature vector as key point feature description vector and fastens image processing, so as to enhance the speed of feature extraction and registration.

Some computer experts have also carried out some practical project, and researched how to apply the feature detection algorithm. For example, in the medical breast checking, the SIFT algorithm was used to detect micro-calcification points of breast X-ray film, which accelerates the speed of manual recognition. In medical treatment, a new 3D-SIFT-Flow method was proposed, which takes advantage of dense correspondence using the informative and robust feature [9]. It is effective and efficient for improving the registration accuracy of large tissue deformation and blurry boundary in CT liver image. Mazur *et al.* [10] applied and optimized SIFT-based dense pixel tracking on 0.35 T cine-MR images during image-guided radiation therapy. Tracking results showed that this method is a high fidelity model in assessing intra-fractional motion on cine-MR acquisitions. In term of detection and description features of image, the core operation of algorithms is to filter the original image to create scale-space [11] images.

In the industry production process, it is necessary to inspect products to make sure that the products come up the standards of quality. For example, in an auto parts factory, Figure 1 is a part of screenshot of product, which was produced by the machine, as shown in Figure 2. In Figure 1(a), it is a qualifying product with flawless, while in Figure 1(b) that is a defective product. Usually, workers detect the product surface by naked eyes whether the product is smooth or not. In Figure 1(b), it is a defective product with two protruding blobs on the product surface, which are signed with blue circle. Although there are some small protruding blobs [12] on the product surface, whose smoothness come up the standards of quality, they are not flaws. Because it took us much time and the labor force to inspect product surface by the artificial processing. This is an encountered problem for inspecting product of auto parts in automotive manufacturing.

The paper is organized into the following sections. In Sect. II, the contribution of the paper is presented. In Sect. III, the problem is analyzed and tentatively solved by several methods. In Sect. IV, the theoretical basis of the proposed algorithm is presented. In Sect. V, description and steps of the proposed algorithm are defined. Finally, the
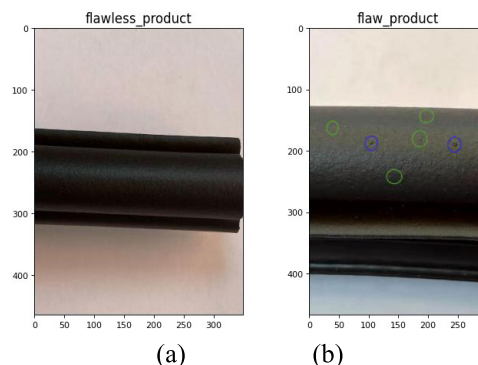
FIGURE 1. (a) A part of the qualifying product. (b) A part of the defective product.

FIGURE 2. A machine that produces auto parts.

discussion and conclusions are presented in Sect. VI and VII respectively.

II. CONTRIBUTION OF THE PAPER

According to analysis and experiments of extracting features of the image of product, we conclude that each flaw on the produce surface is composed of a certain number of pixels, and these pixels are connected to form a large area with similar gray value. In this paper, we exploit a method for identifying flaws on product surface based on spatial connectivity domain. In the light of characteristics of flaws on the product surface, we construct Gaussian scale-space [13] and detecting spot feature filter to process the image. Then, according to the distance and weight of adjacent spots, the large spot was stitched by spatial connectivity domain. Finally, in according to the requirements of product inspection, the spot of the flaw is identified, and the value of the flaw which exceed the threshold value (100) is outputted. On the GitHub website, the code of our proposed algorithm will be available at: https://github.com/zhangquanyou/Identifying-Flaws-on-Product-Surface.git

III. ANALYSIS OF PROBLEM

There are salient object detection(SOD) and general object detection(GOD) in image object detection [14]. But in Figure 1(b), it is difficult to identify the flaw object which signed blue circle label. Because the texture of background of the image is very similar, and the intensity of the feature region is consistent with the surrounding area. So identifying the flaw
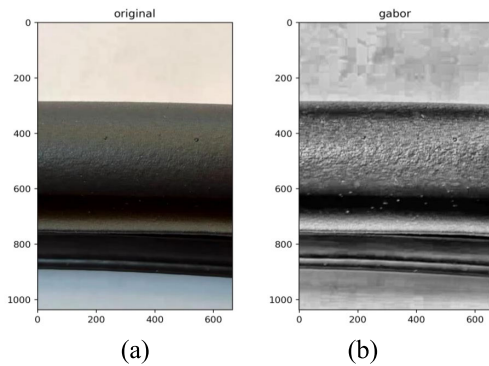
FIGURE 3. (a) Original image of product. (b) The texture image of product surface.

object is much more challenging than the traditional SOD and GOD. Whether or not to identify flaws by extracting texture algorithms or detecting local feature algorithms or identifying edge algorithms?

## A. SHORTCOMING OF EXTRACTING TEXTURE ALGORITHMS

In the aspect of extracting texture feature of image, there are Tamura texture analysis method, Gabor wavelet texture feature extraction method, local binary pattern (LBP) texture feature statistical method and so on [15]. In these methods, it is effective to analyze the surface features of objects by the spatial dependence of gray level. The Tamura texture [16] analysis method is effective to analyze the contrast, coarseness and directionality of the image, but it is poor in recognition of the linelikeness, regularity and roughness of the image. The algorithm of extracting texture feature based on Gabor [17] wavelet transform has better retrieval effect in frequency domain, and it can eliminate redundant information. But the higher the dimension of feature vector increases, the more slowly the retrieval speed becomes. The LBP [18] texture feature statistical method is a kind of analysis method with non-change of the gray level and rotation. It breaks through the difficulty of extracting large-scale texture features. Texture of object is the unique feature of object surface, which does not change after rotation or translation or affine transform or other transforms. During our experiment, in order to verify whether Gabor method is valid or not, we construct a Gabor filter to extract the texture feature of flaws on product surface. Figure 3 shows the effect of extracting texture by the Gabor algorithm. As shown in Figure 3, the original image on the left is processed by Gabor filter to get the Gabor image on the right. According to the Gabor image, the similarity of product texture is great, and flaws on the product surface could not be identified by texture algorithms.

B. SHORTCOMING OF DETECTING LOCAL FEATURE ALGORITHMS

According to Figure 1, flaws on the product surface belong to the local features of image. At present, the main popular

algorithms of detecting local image features include AKAZE algorithm, KAZE algorithm, BRIEF algorithm, ORB algorithm, SURF algorithm, BRISK algorithm and SIFT algorithm [19] and so on. Each algorithm has its own advantages and disadvantages. AKAZE [20] and KAZE [6] are nonlinear algorithms [32], which cost much time to process image and are not suitable for auto mobile industry production line. BRISK or SIFT or SURF or ORB algorithm [21] is a linear algorithm with fast processing speed. But according to the analysis of detected image features, it is easy to recognize the flawless parts as features. In our many experiments, the local feature algorithm was adopted in processing of identifying flaws on produce surface, resulting in the increasing probability of mistaken discrimination. The parameters of SIFT algorithm were set as below: threshold=0.4, edgeThreshold=10, sigma=1.6. In AKAZE and KAZE algorithm, the parameters were set as below: threshold=0.001, nOctaveLayers=4, nOctaves = 4. The parameters of BRISK algorithm was set as below: nOctaves=4, 9.75<threshold<13.67=10. The experiment results are shown in Figure 4.

According to the experiment results, we found that non-defective points were easily recognized as features of flaws on produce surface. At the same time, considering the automation process of industry production line, it is necessary to quickly inspect the local feature of flaws on product surface. We have absorbed the advantages of linear and nonlinear algorithms and detected features of flaws on product surface by the Gaussian scale-space and Difference of Gaussian (DoG) [22]. The features of flaws more salient than that of other non-defective domains, so that flaws on the product surface can be identified as a blob.
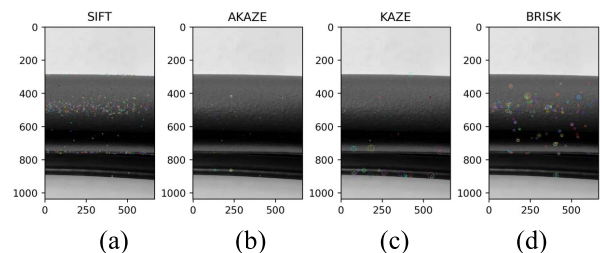
FIGURE 4. (a) Experiment result of SIFT algorithm. (b) Experiment result of AKAZE algorithm. (c) Experiment result of KAZE algorithm. (d) Experiment result of BRISK algorithm.

C. SHORTCOMING OF IDENTIFYING EDGE ALGORITHMS

In the aspect of identifying edge of object, there are Sobel operator, Scharr operator, Roberts operator and so on. The Roberts operator, which is known as cross differential algorithm, is used to detect edge lines by local difference calculation. When the edge of object is close to positive 45 degrees or negative 45 degrees, the algorithm is more effective. The disadvantage of algorithm is that the position of edge is not accurate, and the extracted edge lines are thicker. The Sobel operator [23] is often used in noisy images and gray gradient images. According to the gray weighting between the upper

and lower pixels and the left and right pixels, the Sobel operator detects the edge of object by the extreme value at the edge. The noisy image could be made smooth and the orientation information of edge is accurately preserved. The effect of Sobel operator's Edge localization is good, but it is prone to widen the edge with multi-pixels, and poor to extract the weak edge. Therefore, in order to effectively extract weak edges, the gap between pixel values must be increased. Scharr operator [24], which is an enhanced Sobel operator, increases the difference between pixel values of edge. Because the inside weighting of filter template is greater than the center of it, the accuracy of edge is higher than that of Sobel operator. Although the principle and usage of detecting edges are the same, the effect of weak edge is different. The experiment results are shown in Figure 5.
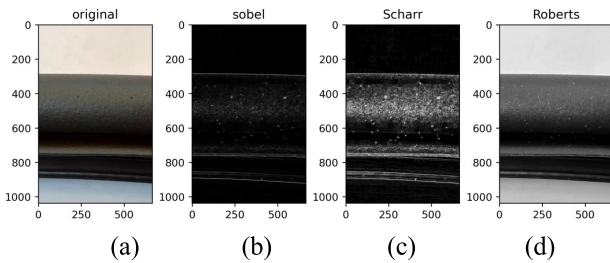
FIGURE 5. (a) Original Image of product. (b) Experiment result of Sobel algorithm. (c) Experiment result of Scharr algorithms. (d) Experiment result of Roberts algorithm.

In addition, there are Prewitt operators, Laplacian operators and so on. The Prewitt operator [25] is very effective in extracting edge of gray gradient image, but it ignores the influence of adjacent points on the current pixel points, which is similar to the Sobel operator. Laplacian operator [26] is very sensitive to noise, so that it is easy to lose the orientation information of some edges. The anti-noise ability of Laplacian is poor, resulting in some discontinuous extracting edges.

IV. THEORETICAL BASIS

A. DIFFERENCE OF GAUSSIAN AND LAPLACIAN OF GAUSSIAN

The Gaussian operator is a smoothing filter, which reduce the sensitivity to noise. The function of Gaussian is shown as below.

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp(-\frac{x^2 + y^2}{2\sigma^2}) \qquad (1)$$

The function of DoG [27] is used to fit the discrete pixels in scale-space. The DoG operator is shown as below.

$$DoG = G(x, y, \sigma_1) - G(x, y, \sigma_2) \qquad (2)$$

For a two-dimensional image $I(x, y)$, the response value of Laplacian of Gaussian (LoG) reaches the maximum when the scale $\sigma = \sqrt{r}$. If the circular spot in the grayscale image, black and white are reversed, the response value of LoG

reaches a minimum at $\sigma = \sqrt{r}$. The Laplacian operator is shown as below.

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \qquad (3)$$

The function of LoG [28] is shown as below.

$$LoG = \sigma^2 \nabla^2 G \qquad (4)$$

First order partial derivative of the Equation (1) with respect to independent variable σ is shown as below.

$$\frac{\partial G}{\partial \sigma} = \frac{-2\sigma^2 + x^2 + y^2}{2\pi\sigma^2} \exp(-\frac{x^2 + y^2}{2\sigma^2}) \qquad (5)$$

$$\frac{\partial G}{\partial \sigma} = \sigma \nabla^2 G \qquad (6)$$

When $\Delta\sigma \to 0$, the limit of Equation (6) in scale-space is shown as below.

$$\frac{\partial G}{\partial \sigma} = \lim_{\Delta\sigma \to 0} \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma}$$
$$= \frac{G(x, y, \sigma_1) - G(x, y, \sigma_2)}{(k - 1)\sigma} \qquad (7)$$

$$G(x, y,)\sigma_1) - G(x, y, \sigma_2) = (k - 1)\sigma^2\nabla^2 G \approx \sigma^2\nabla^2 \qquad (8)$$

Usually, DoG is considered as an approximate extreme value of LoG in order to conveniently compute. We also adopt the DoG operator to detect the local feature of image, which effectively accelerates computing.

B. HARRIS CORNER ALGORITHM

Harris Corner algorithm [29] is an extremely simple detecting algorithm. The algorithm adopts the moving window to calculate the gray level change and identify corner, edge, smooth region by the gray difference [30] of adjacent pixels. As the window moves [u,v], the gray change of pixels in the window is described as follows.

$$E(u, v) = \sum_{x,y} w(x, y)[I(x + u, y + v) - I(x, y)]^2 \qquad (9)$$

The Equation (9) can be expand and approximately computed by Taylor formula as follows.

$$E(u, v) \cong [u \quad v] M \begin{bmatrix} u \\ v \end{bmatrix} \qquad (10)$$

$$M = \sum_{x,y} w(x, y) \begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \qquad (11)$$

Calculating the Harris matrix [31] of each pixel in the image, then getting the eigenvalues of characteristic matrix. The two eigenvalues of uniform plane are small or approximate. The eigenvalues of edge are different, one of which is large while the other is small. The eigenvalues of corner both are large or approximately equal.

V. THE OVERALL WORKFLOW OF THE PROPOSED ALGORITHM

A. SPECIFIC STEPS OF THE PROPOSED ALGORITHM

The proposed algorithm includes seven steps, which are shown as follows.

1) The captured image is read into the program and processed into the gray-scale image. Then adopting normalization operation to make features of image invariant in scale, rotation and affine.

2) Constructing Gaussian scale-space and computing the image by convolution operation, then generating DoG by equation (2). The aim is to reduce noise effect and make features of image more significant by different scales of Gaussian.

3) Judging all pixels' gray values of the image, then using binarization operation by the interval of [minThreshold, maxThreshold]. It is important to filter out the feature of non-defective image for the next processing.

4) According to all pixels' gray values of the image, connecting feature pixels to make large connective domains. Based on binary image, using the findContours() function to extract the connective domain and calculating the center of each connective domain. The function of findContours is designed as below:

   findContours(image, contours, hierarchy, mode, method, offset=Point())

   image–Source. The image is the input data source, which is treated as binary.

   contours–Detected contours. Each contour is stored as a vector of points.

   hierarchy–Optional output vector, containing information about the image topology.

   mode–Contour retrieval mode, which can be set to CV_RETR_EXTERNAL or CV_RETR_LIST or CV_RETR_CCOMP or CV_RETR_TREE.

   method –Contour approximation method, which can be set to CV_CHAIN_APPROX_NONE or CV_CHAIN_APPROX_SIMPLE or CV_CHAIN _APPROX_TC89_L1 or CV_CHAIN_APPROX_TC 89_KCOS.

   offset–Optional offset by which every contour point is shifted.

5) Building block area: According to the minDist distance between the close points on the edge of two connective domains, judging whether the close points are classified into one group or not.

6) Calculating the size of each block. If the value of block is greater than the threshold (100), it would be considered as a flaw.

7) Outputting the value of flaws' area and the coordinate of center points in the image.

B. DESCRIPTION OF THE PROPOSED ALGORITHM

According to our proposed algorithm, calculating time complexity of the algorithm step by step as below:

Algorithm Process (I, σ, δ, minDis, Threshold)

Input: Image I, **Parameter** $\sigma = 5$, δ, $minDis=0.1$, $Threshold=5$

Output: Position$S$, Value$V$

    1: Import an original image I: I→Gray(I)→Q(I)
                    #normalization
    2: for σ in sigma: # $\sigma \rightarrow$ Gaussian and DoG
       L(x,y;σ) = f(x,y) $\otimes$ G(x,y;σ)
       Q(I)   #normalization
       while(I[P] > δ){
          If I(,)>minThreshold #Binarization,
                        minThreshold=5
                I(,)=1
             else I(,)=0
    3: do{
          I$_2$ = findContours(I)}   #Constructing
                                        Connectivity
    4: For l$_1$,l$_2$ in I$_2$:
       If I$_2$(l$_1$,l$_2$) >minDist   #Judge the distance
                                         of Connectivity
       I$_2$(l)=I$_2$(l$_1$,l$_2$)   #Coupling
    5: For k in I$_2$:

Grey-scale Map: m*n
Normalization: m*n
Scales of Gaussian: m*n*kszie
Binarization: m*n
Constructing Connectivity: m*n
Building block area: m*n*d
Judging the area by threshold: L

In the expression above, m*n indicates the size of image, kszie indicates the size of the filter of Gaussian, d indicates the longest diameter of the block area.

The total time complexity of the algorithm is O (m*n)+O (m*n*kszie) + O (m*n*d).

Because executing the algorithm only need one dimensional array and two dimensional arrays, the total space complexity of our algorithm is O(m*n) + O(n).

VI. EXPERIMENT AND DISCUSSION

Experiment environment: operation system (OS) Win 10; Development platform: Python3.8+OpenCV+Jupyter notebook; CPU: Intel(R) Core (TM) i7; GPU: GeForce RTX 2060SUPER; Memory:8 G; Disk:1 T.

Sample 1 with 665 × 1037 PPI, 24-bit RGB and 92.9K was captured by a HD camera. After processing the image of product surface via the proposed algorithm, the effect of flaws on product surface is shown in Figure 6. The red circle in the Figure 6 are flaws on product surface. In order to make the flaws obviously distinguish, we annotate it by drawing straight lines.
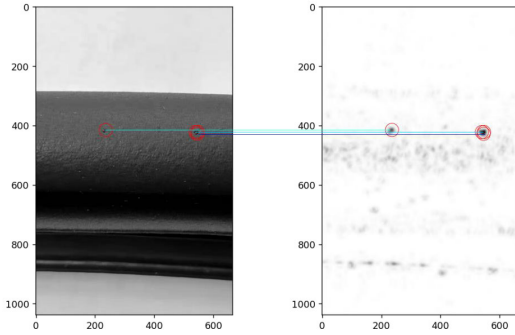
Sample 2 with 4032 × 3024 PPI, 24-bit RGB and 1.16M was also captured by a HD camera. After processing the image of product surface via the proposed algorithm,

the effect of flaws on product surface, as shown Figure 7 on the right. The red circle in the Figure 7 are flaws on product surface. In order to make the flaws obviously distinguish, we also annotate it by drawing straight lines.
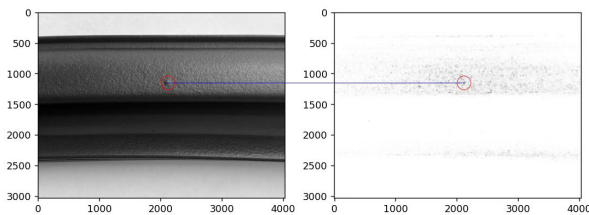

FIGURE 6. The effect of processed sample 1.

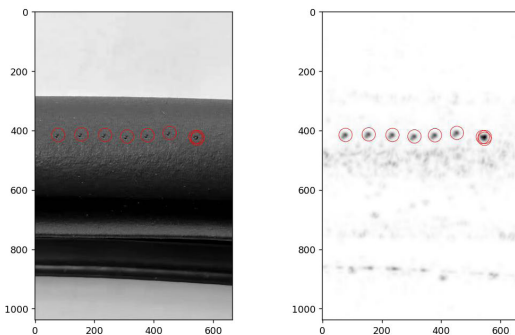**FIGURE 7.** The effect of processed sample 2.

FIGURE 8. The effect of processed sample 3.

Sample 3 with 662×996 PPI, 24-bit RGB and 179K was captured by a HD camera. the effect of flaws on product surface is shown in Figure 8. There are seven flaws on the product surface. The experiment result indicates the flaws are easily distinguished by the proposed algorithm.

In the experiment, we have compared our proposed algorithm with detecting local features algorithms, identifying edge algorithms and extracting texture algorithms. In the aspect of detecting features, the local feature algorithms [3]–[8] extract too many feature points to identify flaws on the product surface of auto parts, as shown in Table 1. Some points of non-defective surface are also extracted as key features, as shown in Sect. 3 Figure 4. In addition, the texture of

TABLE 1. Result of the proposed algorithm and other algorithm.

Algorithm Sample	Proposed ALGO	AKAZE	KAZE	BRISK	SIFT
Sample 1	2	28	41	174	503
Sample 2	1	6875	9000	14530	23155
Sample 3	7	30	50	182	511

TABLE 2. Test of the different lightness and angle.

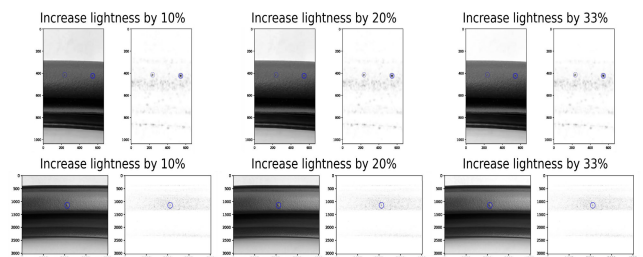| Sample	Lightness(%)			Angle(°)			
	20%	33%	34%	3°	10°	90°	180°
Sample 1	Y	Y	N	Y	Y	Y	Y
Sample 2	Y	Y	N	Y	Y	Y	Y

FIGURE 9. The sample with the increment of lightness (10%, 20%, 33%), the flaws on the product surface by the proposed algorithm.

flaws on the product surface is very similar to the background texture, so it is very difficult to extract feature of flaws by the extracting texture algorithm [17], [18]. The defective product could not be identified, as shown in Sect. 3 Figure 3. We also experiment to identify flaws on product surface via the identifying edge algorithm [23], [24]. It is invalid to extract the feature of surface image of product, as shown in Sect. 3 Figure 5. So we propose a method for identifying flaws on product surface based on spatial connectivity domain. The method is used to accurately identify a number of flaws on the product surface, as shown in this Sect. Figure 6,7. Moreover, the speed of the proposed algorithm is also suitable to detect product of auto parts in factory assembly line.
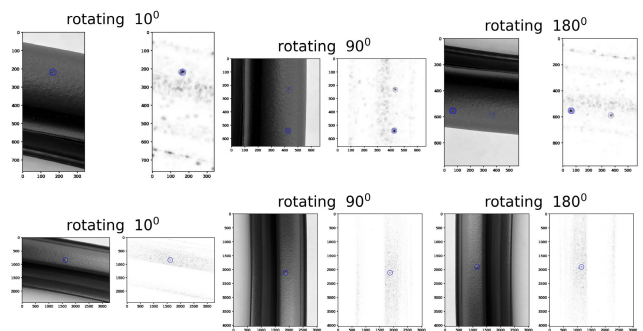

FIGURE 10. The sample with the different angle rotating (10°, 90°, 180°), the flaws on the product surface by the proposed algorithm.

The lightness or angle of view could affect the detection robustness. It is important to evaluate and calibrate the algorithm under different light conditions. In order to verify

the detection robustness of the proposed algorithm, we have changed the lightness and angle of samples to experiment in different lightness or each angle condition, as shown Table 2. We found that when lightness is increased by over 33%, the proposed algorithm is unavailable. But the sample with the increment of lightness ranges from 0 to 33%, our algorithm is available, as shown Figure9. Furthermore, the proposed algorithm is available when the sample was rotated to adjust the arbitrary angle, as shown Figure10.

VII. CONCLUSION

During the production process of inspecting product, the manual processing costs much time and labor force. It is an encountered problem to identify the features of flaw on the product surface. Because the texture of image is very similar, and the intensity of the feature region is consistent with the surrounding area. So it is much more challenging than the traditional algorithms, such as extracting texture algorithms, detecting local feature algorithms and identifying edge algorithms. We have compared Tamura texture analysis method, Gabor wavelet texture extraction method, LBP texture statistical method, AKAZE algorithm, KAZE algorithm, BRISK algorithm, SIFT algorithms, Sobel algorithm, Scharr algorithm, Roberts algorithm. These algorithms are invalid to identify flaws of the product by extracting the image feature of the product surface. So we exploit a method for identifying flaws on product surface based on spatial connectivity domain. The proposed method can accurately identify a number of flaws on the product surface. According to experiments of samples, the effect of identifying flaws on product surface is good. Moreover, the speed of the proposed algorithm is also suitable to inspect the product of auto parts in factory assembly line.

REFERENCES

[1] S. A. K. Tareen and Z. Saleem, "A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK," in *Proc. Int. Conf. Comput., Math. Eng. Technol. (iCoMET)*, Mar. 2018, pp. 1–10.

[2] D. Lowe and X. Y. Kong, "Distinctive image features from scale-invariant keypoints," in *Proc. IEEE/WWIC/ACM Int. Conf. Web Intell.*, Nov. 2004, pp. 91–97.

[3] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "BRIEF: Binary robust independent elementary features," in *Computer Vision* (Lecture Notes in Computer Science), vol. 6314. Berlin, Germany: Springer, 2010, pp. 778–792.

[4] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2564–2571.

[5] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 2548–2555.

[6] P. F. Alcantarilla, A. Bartoli, and A. J. Davison, "KAZE features," in *Proc. 12th ECCV*, vol. 7577, 2012, pp. 214–227.

[7] J.-X. Zhang, D. Li, M. Zhang, and M.-F. Hong, "3D reconstruction based on KAZE quasi-dense match algorithm," *Basic Clin. Pharmacol. Toxicol.*, vol. 125, p. 145, Dec. 2019.

[8] C. S. Prakash, P. P. Panzade, H. Om, and S. Maheshkar, "Detection of copy-move forgery using AKAZE and SIFT keypoint extraction," *Multimedia Tools Appl.*, vol. 78, no. 16, pp. 23535–23558, Aug. 2019.

[9] Y. Xu, C. Xu, X. Kuang, H. Wang, E. I.-C. Chang, W. Huang, and Y. Fan, "3D-SIFT-Flow for atlas-based CT liver image segmentation," *Med. Phys.*, vol. 43, no. 5, pp. 2229–2241, May 2016.

[10] T. R. Mazur, B. W. Fischer-Valuck, Y. Wang, D. Yang, S. Mutic, and H. H. Li, "SIFT-based dense pixel tracking on 0.35 T cine-MR images acquired during image-guided radiation therapy with application to gating optimization," *Med. Phys.*, vol. 43, no. 1, pp. 279–293, Jan. 2016.

[11] R. Cheng, Y. Jin, K. Narukawa, and B. Sendhoff, "A multiobjective evolutionary algorithm using Gaussian process-based inverse modeling," *IEEE Trans. Evol. Comput.*, vol. 19, no. 6, pp. 838–856, Dec. 2015.

[12] Z. Lin, J. Jia, W. Gao, and F. Huang, "A hierarchical coarse-to-fine perception for small-target categorization of butterflies under complex backgrounds," *J. Intell. Fuzzy Syst.*, vol. 38, no. 3, pp. 3463–3487, Mar. 2020.

[13] D.-P. Fan, G.-P. Ji, G. Sun, M.-M. Cheng, J. Shen, and L. Shao, "Camouflaged object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2777–2787.

[14] C.-K. Yeh, H.-K. Chu, M.-J. Chang, and T.-Y. Lee, "Anamorphic image generation using hybrid texture synthesis," *J. Inf. Sci. Eng.*, vol. 34, no. 1, pp. 123–134, Jan. 2018.

[15] N. Pal, A. Kilaru, Y. Savaria, and A. Lakhssassi, "Hybrid features of tamura texture and shape based image retrieval," in *Proc. 5th Int. Conf. Adv. Comput., Netw., Inform. (ICACNI)*, 2017.

[16] H. Yang, L. Shen, Y.-D. Yao, H. Wang, and G. Zhao, "Finger vein image inpainting with Gabor texture constraints," *IEEE Access*, vol. 8, pp. 83041–83051, 2020.

[17] D. Bhattacharya, N. Sinha, S. Prasad, P. K. Pal, J. Saini, and S. Mangalore, "A new statistical framework for corpus callosum sub-region characterization based on LBP texture in patients with parkinsonian disorders: A pilot study," *Frontiers Neurosci.*, vol. 14, p. 477, May 2020.

[18] I. A. Kazerouni, G. Dooly, and D. Toal, "Underwater image enhancement and mosaicking system based on A-KAZE feature matching," *J. Mar. Sci. Eng.*, vol. 8, no. 6, p. 449, Jun. 2020.

[19] D. Li, Q. Xu, W. Yu, and B. Wang, "SRP-AKAZE: An improved accelerated KAZE algorithm based on sparse random projection," *IET Comput. Vis.*, vol. 14, no. 4, pp. 131–137, Jun. 2020.

[20] S. K. Sharma and K. Jain, "Image stitching using AKAZE features," *J. Indian Soc. Remote Sens.*, vol. 48, no. 10, pp. 1389–1401, Oct. 2020.

[21] M. Russell, J. J. Zou, G. Fang, and W. Cai, "Feature-based image patch classification for moving shadow detection," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 9, pp. 2652–2666, Sep. 2019.

[22] X. Jiang, J. Jiang, A. Fan, Z. Wang, and J. Ma, "Multiscale locality and rank preservation for robust feature matching of remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6462–6472, Sep. 2019.

[23] G. Domínguez-Rodríguez, J. A. González-Sánchez, J. Rosado-Carrasco, and G. Canto, "Grain boundaries and phases identification of metallographic images by a normalized Sobel operation and the edge thinning process for further numerical simulation," *Met. Mater. Int.*, vol. 26, no. 9, pp. 1306–1317, Sep. 2020.

[24] Z. Yu, J. Wang, and G. Lu, "Optimized self-adapting contrast enhancement algorithm for wafer contour extraction," *Multimedia Tools Appl.*, vol. 78, no. 22, pp. 32087–32108, Nov. 2019.

[25] R.-G. Zhou, H. Yu, Y. Cheng, and F.-X. Li, "Quantum image edge extraction based on improved Prewitt operator," *Quantum Inf. Process.*, vol. 18, no. 9, p. 261, Sep. 2019.

[26] W. Waheed, G. Deng, and B. Liu, "Discrete Laplacian operator and its applications in signal processing," *IEEE Access*, vol. 8, pp. 89692–89707, 2020.

[27] L. Fei, G. Lu, W. Jia, S. Teng, and D. Zhang, "Feature extraction methods for palmprint recognition: A survey and evaluation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 2, pp. 346–363, Feb. 2019.

[28] G. Xu, R. Song, F. Wang, and H. Yuan, "Research on SIFT polarization image registration method based on matching optimization," in *Proc. 5th Conf. Frontiers Opt. Imag. Technol. Appl.*, Nov. 2018, Art. no. 1083206.

[29] C. Wang, Z. Zhang, and X. Zhou, "An image copy-move forgery detection scheme based on A-KAZE and SURF features," *Symmetry*, vol. 10, no. 12, p. 706, Dec. 2018.

[30] W. Zeng, W. Xie, Z. Yang, and X. Lu, "A universal sample-based background subtraction method for traffic surveillance videos," *Multimedia Tools Appl.*, vol. 79, nos. 31–32, pp. 22211–22234, Aug. 2020.

[31] G. Lin, Y. Tang, X. Zou, J. Cheng, and J. Xiong, "Fruit detection in natural environment using partial shape matching and probabilistic Hough transform," *Precis. Agricult.*, vol. 21, no. 1, pp. 160–177, Feb. 2020.

[32] P. F. Alcantarilla and T. Solutions, "Fast explicit diffusion for accelerated features in nonlinear scale spaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1281–1298, Oct. 2011.
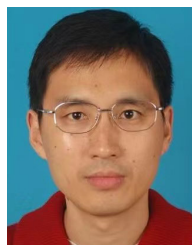
BAO-HUA QIANG received the Ph.D. degree. He is currently with Guilin University of Electronic Technology. From May 2007 to August 2007, he was an Academic Visitor at the University of Illinois. He is a Professor and a Doctor. His expertise include web information processing and intelligent search. His research interests include web information processing, intelligent search, and network information integration.

QUANYOU ZHANG is currently pursuing the Ph.D. degree. He had been studying in Canada as an Academic Visitor with the University of Ottawa, from July 2017 to December 2017. He is currently with Chongqing University and Xuchang University. He is an Associate Professor. His expertise includes image processing. His main research interests include image data processing and algorithms.

YAOHUI LI received the degree from Huazhong University of Science and Technology, in 2015, and the Ph.D. degree. He had been studying in Canada as an Academic Visitor with the University of Ottawa, from July 2017 to December 2017. He is an Associate Professor with Xuchang University. His expertise includes global optimization algorithm.

YONG FENG received the Ph.D. degree. He is currently with Chongqing University. His expertise include artificial intelligence and data retrieval. He is a Professor and a Ph.D. Adviser. His research interests include big data analysis and data mining, artificial intelligence and big data processing, deep learning, and system development of data retrieval.
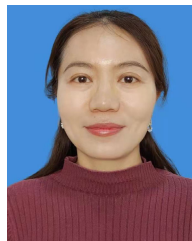
QIONGJIE KOU received the degree from Henan University, in 2007, and the Master of Education degree. She is with Xuchang University. Her expertise includes technology of education. Her research interests include technology of education and information processing.

• • •