

Research Article

Defect Detection in Tire X-Ray Images Using Weighted Texture Dissimilarity

Qiang Guo,^{1,2,3} Caiming Zhang,^{2,3} Hui Liu,^{1,3} and Xiaofeng Zhang⁴

¹*School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China*

²*School of Computer Science and Technology, Shandong University, Jinan 250100, China*

³*Shandong Provincial Key Laboratory of Digital Media Technology, Jinan 250014, China*

⁴*School of Information and Electrical Engineering, Ludong University, Yantai 264025, China*

Correspondence should be addressed to Qiang Guo; guoqiang@sdufe.edu.cn

Received 17 September 2015; Accepted 18 February 2016

Academic Editor: Pietro Siciliano

Copyright © 2016 Qiang Guo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Automatic defect detection is an important and challenging problem in industrial quality inspection. This paper proposes an efficient defect detection method for tire quality assurance, which takes advantage of the feature similarity of tire images to capture the anomalies. The proposed detection algorithm mainly consists of three steps. Firstly, the local kernel regression descriptor is exploited to derive a set of feature vectors of an inspected tire image. These feature vectors are used to evaluate the feature dissimilarity of pixels. Next, the texture distortion degree of each pixel is estimated by weighted averaging of the dissimilarity between one pixel and its neighbors, which results in an anomaly map of the inspected image. Finally, the defects are located by segmenting this anomaly map with a simple thresholding process. Different from some existing detection algorithms that fail to work for tire tread images, the proposed detection algorithm works well not only for sidewall images but also for tread images. Experimental results demonstrate that the proposed algorithm can accurately locate the defects of tire images and outperforms the traditional defect detection algorithms in terms of various quantitative metrics.

1. Introduction

Due to unclean raw materials and undesired manufacturing facilities used in the tire manufacturing process, tire components may be contaminated by various defects, such as metallic or nonmetallic impurities (e.g., steel threads, screws, and plastic fragments), bubble, and overlap. When a vehicle with the defective tire is at high speeds, these defects often lead to a blowout of the tire. Therefore, the nondestructive defect detection technique based on X-ray imaging is essential for tire quality assurance. The traditional quality inspection process is mostly performed by human inspection, which often occurs from inaccurate and undetected inspection results due to visual fatigue and leads to low efficiency with high labor costs [1]. As a result, computer vision based detection techniques have become an important and efficient tool to improve quality of the products and increase manufacturing efficiency [2].

Automatic quality inspection of industrial products has been a popular research topic in the image processing and computer vision communities. Many methods based on different theories, such as texture analysis and spectral analysis, have been proposed to address the limitations of manual inspection. In the texture based detection methods, the detection of defects is carried out by comparing texture features of different image patches. Therefore, a key issue for such methods is texture feature extraction. Latif-Amet et al. [3] used the subband cooccurrence matrices (CM) to characterize texture features of multiscale subbands of inspected images. A major disadvantage of CM is the high computational complexity for large-size images. In [4], Tajeripour et al. proposed a defect detection method which applies the local binary pattern (LBP) to extract texture features. However, for low-quality images, LBP has poor performance in the feature description.

Meanwhile, there also exist many transform-based methods for defect detection. Due to its capability of singularity analysis and noise immunity, the wavelet transform is well suited for finding the location of anomalies in textured images. Tsai and Hsiao [5] proposed a wavelet-based method for automatic surface inspection, which generates an image with enhanced local anomalies by reconstructing the selected wavelet coefficients according to a synthesis strategy. Serdaroglu et al. [6, 7] use independent component analysis (ICA) and topographic ICA to generate feature vectors of wavelet subbands. Then the defects are detected according to the Euclidean distance between feature vectors. In addition, the Gabor transform is also a popular tool for extracting local frequency information of textured images. In [8], Kumar and Pang utilized Gabor filters to detect fabric defects, in which a foreground image is firstly extracted and then the defects are located by segmenting it straightforwardly. However, this method is sensitive to the choice of filter parameters. Although the optimized filters have been developed for defect detection in [9], they have a high computational cost. Unlike the above methods that use fixed basis functions to represent images, Tsai et al. [10] proposed a method for defect detection in solar modules, which uses ICA to learn a set of basis functions from defect-free training images. Each image under inspection can be represented as a linear combination of the learned basis functions. By analyzing the reconstruction error between the image under inspection and the reconstructed image from representation coefficients, defects can be detected. Similarly, some detection methods based on sparse representation have been introduced in different application fields [11, 12]. These methods use a sparse constraint to learn an adaptive representation dictionary from test images. Unfortunately, the learning process is computationally expensive.

Most of existing automatic inspection techniques are focused on textile [1–4], solar wafer [10, 13], and flat steel [14]. Recently, a few research works for tire defect detection have been reported in the literature. In [15], Guo and Wei proposed a detection method based on the image component decomposition (ICD) technique, which exploits the local total variation filtering and the vertical mean filtering to separate defects from inspected images. From the point of view of edge detection, Zhang et al. [16, 17] detected defects by using the multiscale geometric transform and the edge detection operator. Based on the dictionary representation technique, Xiang et al. [18] proposed a dictionary based detection method for tire defects by analyzing the distribution of representation coefficients. However, these methods for tire defects detection were designed only for tire sidewall images. Consequently, all of them fail to work for tire tread images due to their complex structures. Although a density projection based method [19] was presented for detecting tire tread images, this method only provides the orientation information of defects and cannot accurately locate them.

The reason for the weakness of the above methods is that they do not effectively capture texture distortions of images. To address this problem, a simple and efficient detection method is proposed in this paper, which takes advantage of the feature similarity of tire images. Specifically, for an

inspected image, the proposed method firstly estimates the texture distortion degree of each pixel by weighted averaging of the dissimilarity between this pixel and its neighbors within a local window and produces an anomaly map of the inspected tire image. Then the defects are located by segmenting this anomaly map with a simple thresholding function. Experimental results on tire X-ray images show that the proposed method can effectively detect the defects both in the tire sidewall and in the tire tread. To the best of our knowledge, this is the first work that can accurately locate the defects on tire thread images. In addition, as the computational core of the proposed algorithm, computing the feature dissimilarity of image pixels can be implemented independently in parallel, which makes the proposed method feasible for tire online inspection.

2. Characteristics of Tire Images

Due to imperfect raw materials and manufacturing process, tire sidewall and tire tread may contain various types of defects such as impurity, bubble, and overlap. Figure 1 shows the main defects of tire sidewall and tire tread in the quality inspection. Figure 1(a) displays a tire sidewall image that contains a metallic impurity, in which the region of impurity is darker than its neighbors. On the contrary, a bubble on the tire sidewall shown in Figure 1(b) is brighter than the surrounding pixels. Figure 1(c) is an example of the overlap which results in an irregular texture. Figures 1(d) and 1(e) further show two tire tread images, which, respectively, contain an impurity and an overlap.

From Figure 1, it is observed that tire tread images contain more complicated textures with lower contrast than tire sidewall images, which leads to difficulties in detecting defects in the tire tread. Apart from this observation, it is further noted that both tire sidewall images and tire tread images are dominated by a texture which manifests a high regularity and defects locally break the regularity of a texture. This regularity generally means a similarity between pixel features. Therefore, the defects can be located by analyzing the feature dissimilarity of a pixel and its surrounding ones. In the following section, the proposed detection method based on the feature dissimilarity analysis is described in detail.

3. The Proposed Method

The similarity between pixels of a textural image implies that there exists an implicit dependency between one pixel and its neighbors. Therefore, a pixel can be represented as a weighted linear combination of the surrounding pixels. Based on the above considerations, we propose a detection method by using weighted texture dissimilarity to measure perceptual texture distortion, in which the distortion of a pixel is defined as the dissimilarity between the original feature value and the represented one of pixels. Figure 2 shows a block diagram of the proposed method, which consists of three main steps: extracting texture features, evaluating structural dissimilarity, and segmenting the defects. Texture features of each pixel are extracted using the local kernel regression (LKR) descriptor

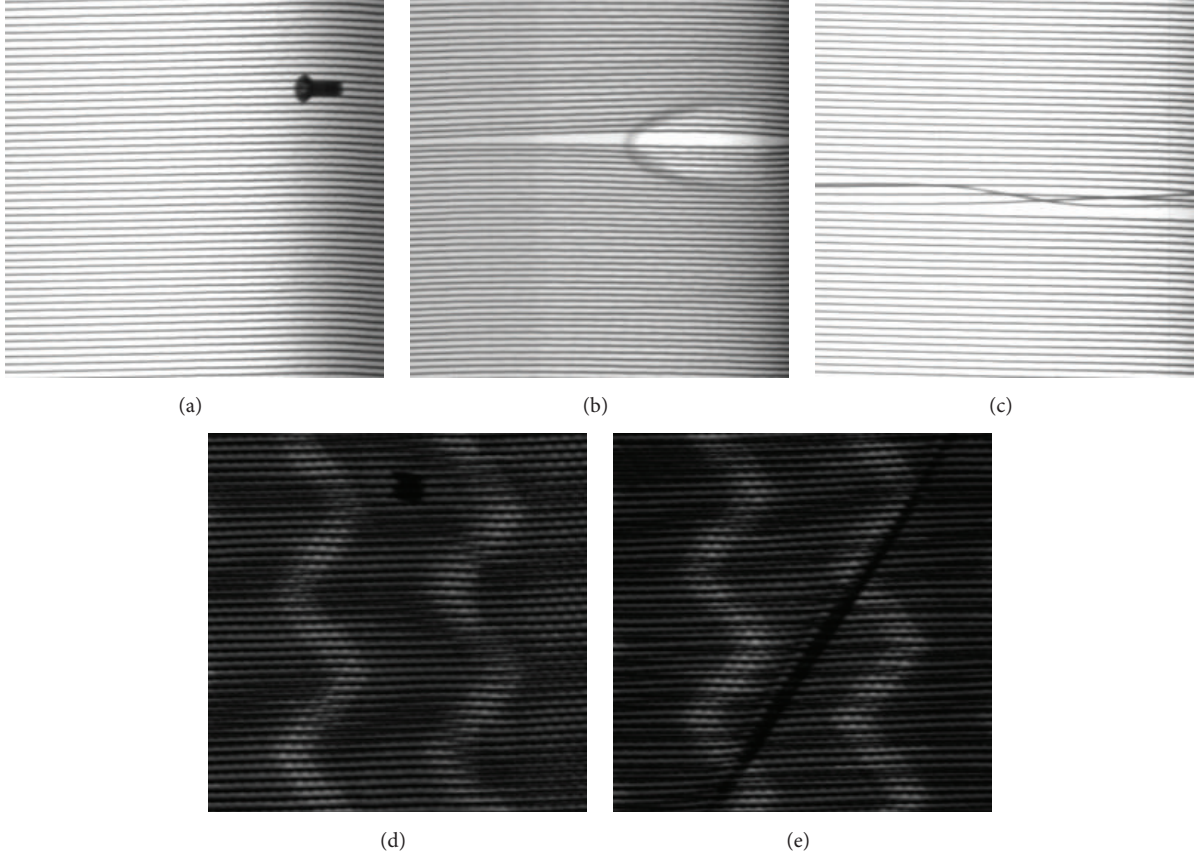


FIGURE 1: Examples of defective images. (a) Impurity in tire sidewall, (b) bubble in tire sidewall, (c) overlap in tire sidewall, (d) impurity in tire tread, and (e) overlap in tire tread.

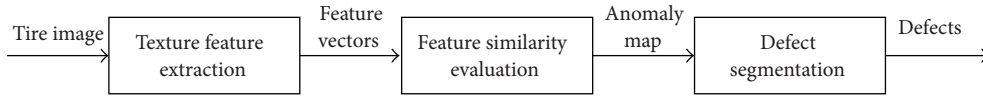


FIGURE 2: Block diagram of the proposed method.

[20] and represented as vectors of features. Then the anomaly of each pixel is determined by weighting the dissimilarity between this pixel and its local neighbors. Finally, the defects are located by segmenting the anomaly map with a simple thresholding process.

In order to determine the distortion of each pixel, we introduce an anomaly value for each pixel to quantize its distortion. Specifically, the anomaly value of a pixel I_i in an image I is defined as follows:

$$A(I_i) = \frac{\sum_{j \in W_i} w_{i,j} D(I_i, I_j)}{\sum_{j \in W_i} w_{i,j}}, \quad (1)$$

where $w_{i,j}$ is a weight determined from the input data, W_i is a spatial window centered at pixel I_i , and $D(I_i, I_j)$ is a feature dissimilarity metric between pixels I_i and I_j . The key issue in calculating the anomaly value $A(I_i)$ is how to determine the feature dissimilarity metric $D(I_i, I_j)$. Therefore, we will describe the feature descriptor adopted to represent

the texture structure of inspected images first, the definitions of $D(I_i, I_j)$, and $w_{i,j}$ afterwards.

3.1. Texture Feature Extraction. Different from the intensity-based metric for similarity measurement, the proposed method uses the feature correlation as a quantitative measure of the similarity between pixels and their neighbors, which is motivated by the following two reasons: on the one hand, texture feature extraction of inspected images is essential for measuring the perceptual texture distortion, which is an efficient way to avoid the negative influences of illumination changes for similarity measurement. On the other hand, we use the correlation metric to calculate patch similarity which has the advantage of being insensitive to outliers.

Due to its robustness to noise and other perturbations, we adopt the LKR descriptor to derive a set of feature vectors of an image, which acquires the textural structures of images by analyzing the local gradient information. Unlike scale-invariant feature transform (SIFT) and histogram of gradient

(HOG) which use the quantization of oriented gradients to reduce computational cost, LKR computes the texture feature between oriented gradients without the quantization step, which leads to a better feature description power. In general, LKR is more invariant to the shift and rotation transformations than the conventional feature descriptors based on gradients and key points. For a pixel at position $\mathbf{x} = [x_1 \ x_2]^T$, the LKR feature vector is defined as a self-similarity between this pixel and its neighbors, which is derived by the following form:

$$F(\mathbf{x}) = \exp\left(-(\mathbf{x}_i - \mathbf{x})^T \mathbf{C}_i (\mathbf{x}_i - \mathbf{x})\right), \quad i \in [1, \dots, n], \quad (2)$$

where $\mathbf{x}_i = [x_1 \ x_2]^T$ is the spatial coordinates, n is the number of pixels in a local window Ω_i with size $\sqrt{n} \times \sqrt{n}$, and \mathbf{C}_i is a gradient covariance matrix. In the numerical calculation, \mathbf{C}_i is estimated by averaging a collection of spatial gradient vectors within the local window and can be written as follows:

$$\mathbf{C}_i \approx \begin{bmatrix} \sum_{\mathbf{x}_j \in \Omega_i} z_{x_1}(\mathbf{x}_j) z_{x_1}(\mathbf{x}_j) & \sum_{\mathbf{x}_j \in \Omega_i} z_{x_1}(\mathbf{x}_j) z_{x_2}(\mathbf{x}_j) \\ \sum_{\mathbf{x}_j \in \Omega_i} z_{x_1}(\mathbf{x}_j) z_{x_2}(\mathbf{x}_j) & \sum_{\mathbf{x}_j \in \Omega_i} z_{x_2}(\mathbf{x}_j) z_{x_2}(\mathbf{x}_j) \end{bmatrix}, \quad (3)$$

where $z_{x_1}(\cdot)$ and $z_{x_2}(\cdot)$ are the first derivatives along x_1 and x_2 directions, respectively.

From (3) it is seen that \mathbf{C}_i can be interpreted as averaging geodesic distances in a patch. Thus it is robust to noise and other perturbations. In addition, as a key part of LKR, \mathbf{C}_i can capture the singularities of images, which makes LKR suitable for describing the texture feature of images. For complete details, we refer the interested reader to the work of Takeda et al. [20]. In order to validate the effectiveness of LKR, the LKR magnitudes (for a LKR feature vector $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, we define its LKR magnitude as the square root of the sum of the squares of its elements; i.e., $\|\mathbf{x}\|_{\text{LKR}} = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$) of two tire defective images are displayed in Figure 3. It can be observed that the LKR descriptor can effectively capture local texture distortion of tire images.

3.2. Feature Dissimilarity Evaluation. In general, the computation of pixel-based dissimilarity metric $D(I_i, I_j)$ is quite sensitive to noise so that the resulting anomaly values of pixels are unstable in the presence of noise. Instead of making use of a pixel for similarity measurement, the proposed method exploits a patch-based metric to improve the robustness of measuring similarity (patch-based metric has been extensively used in the image processing community. Its robustness against noise has also been demonstrated in the literature. See [21, 22] for some application examples). Thus the anomaly model (1) can be generalized as

$$A(I_i) = \frac{\sum_{j \in W_i} w_{i,j} D(P_i, P_j)}{\sum_{j \in W_i} w_{i,j}}, \quad (4)$$

where P_i and P_j are the patches of pixels centered at I_i and I_j , respectively. For the simplification of notations, we use $D_{i,j}$ instead of $D(P_i, P_j)$.

In (4), the dissimilarity metric $D(\mathbf{x}, \mathbf{y})$ plays a significant role in computing anomaly. In this paper, we exploit a correlation metric based on the LKR feature descriptor to measure the dissimilarity between image patches, which is defined as follows:

$$D(P_i, P_j) = D_{i,j} = \exp\left(-\frac{\text{Corr}(F(P_i), F(P_j))}{\sigma_1}\right), \quad (5)$$

where

$$\text{Corr}(\mathbf{x}, \mathbf{y}) = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\sqrt{\langle \mathbf{x}, \mathbf{x} \rangle} \sqrt{\langle \mathbf{y}, \mathbf{y} \rangle}} = \frac{\mathbf{x}^T \mathbf{y}}{\|\mathbf{x}\| \|\mathbf{y}\|}, \quad (6)$$

$F(\cdot)$ denotes a LKR feature extraction operator, and σ_1 is an adjustment parameter. The correlation defined in (6), also called cosine similarity, is a measure of the cosine angle between \mathbf{x} and \mathbf{y} because the inner product $\langle \mathbf{x}, \mathbf{y} \rangle$ depends exclusively on the angle between two vectors. Due to the advantage of robustness to noise, the cosine similarity outperforms the conventional metrics for the similarity measurement, such as Euclidean (L2) distance or Manhattan (L1) distance [23]. In fact, the cosine similarity is the uncentered Pearson correlation coefficient, which avoids subtracting the vector means and permits a simpler calculation. We evaluated the performance of the cosine similarity and Pearson correlation coefficient and found that the performance of cosine similarity is very close to Pearson correlation coefficient. Therefore, we use it to measure the similarity of LKR feature vectors.

In essence, (1) is equivalent to applying a weighted filter to the dissimilarity values between pixels. The simplest form of a weighted filter is the uniform weight that assigns the same weight to all data; for example, all weights set to 1. However, the uniform weight generally leads to an oversmoothed result. In the literature, there exists various weight functions such as Gaussian weight, Geman-McClure weight, and Hebert-Leahy weight [24]. For calculating the anomaly map, the weight value should be inversely proportional to the degree of dissimilarity. Thus, we adopt the following weight function due to its simplification:

$$w_{i,j} = \frac{1}{|D_{i,j} - D_{\min}|^2 + \varepsilon}, \quad (7)$$

where $D_{\min} = \min(D_{i,j})$, $j \in W_i$, and $\varepsilon = 0.05$ is a small constant that avoids division by zero.

For each pixel I_i , we can yield its anomaly value by substituting (5) into (4). The smaller the value of $A(I_i)$ is, the smaller the possibility of I_i belonging to a defect is. Therefore, we can use $A(I_i)$ as a dissimilarity measure to distinguish defect pixels from the inspected image. In essence, A defines an anomaly map of the image. Anomaly maps of six tire tread images are displayed in Figures 6(b) and 7(b). It can be noted that the entire defect regions are clearly highlighted in the anomaly maps, which facilitates an accurate segmentation of defects by using a simple segmentation algorithm.

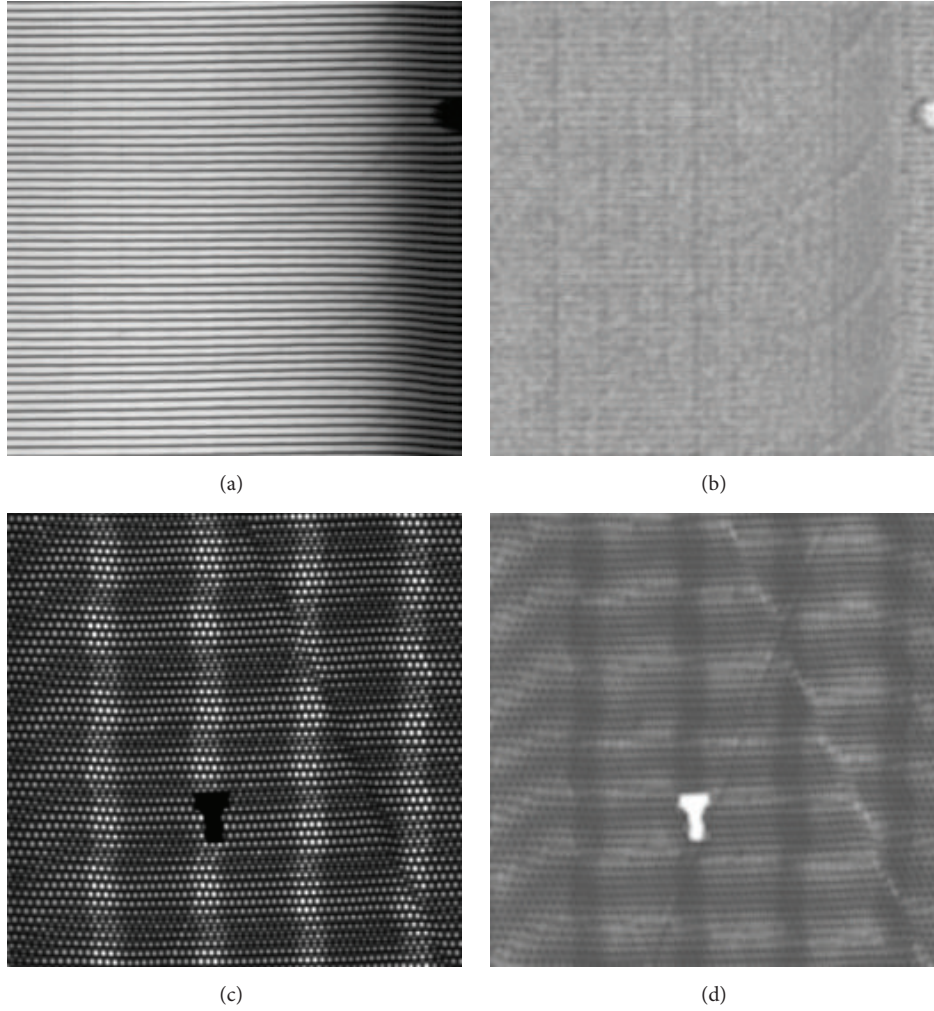


FIGURE 3: Examples of LKR. (a) Tire sidewall image, (b) LKR magnitude of image (a), (c) tire tread image, and (d) LKR magnitude of image (c).

3.3. Defect Segmentation. After generating the anomaly map, defects can be located by way of a thresholding process. Specifically, if the anomaly value of a pixel is larger than a given threshold λ , this pixel is considered to be a part of a defect and its value is set to 1. On the contrary, it is set to 0 if its anomaly value is below the threshold. This simple thresholding process can be formulated as follows:

$$\hat{I}_i = \begin{cases} 1, & A(I_i) > \lambda, \\ 0, & A(I_i) \leq \lambda. \end{cases} \quad (8)$$

The output of the above process is a binary image, in which white pixels denote defective regions and black pixels denote defect-free regions.

Although the defects can be detected by using the above thresholding process, there exists a small amount of defect-free pixels which are detected as defective. To resolve this problem and improve the robustness of the proposed algorithm, the pixels contained within defective regions further determined the identity of a defect by the local variance

analysis. The local variance of a pixel I_i is calculated over a local window W_i centered at I_i , which can be formulated as

$$\text{Var}(I_i) = \sum_{j \in W_i} \delta_j (I_j - \bar{I}_i)^2, \quad (9)$$

where δ_j is a normalized Gaussian weight and \bar{I}_i is the local mean defined as

$$\bar{I}_i = \sum_{j \in W_i} \delta_j I_j. \quad (10)$$

By integrating the local variance into (8), the thresholding function is then modified as

$$\hat{I}_i = \begin{cases} 1, & A(I_i) > \lambda, \text{ Var}(I_i) < \tau, \\ 0, & \text{otherwise,} \end{cases} \quad (11)$$

where τ is a given threshold.

The complete operational procedure of the proposed detection method can be algorithmically summarized in Algorithm 1.

Input: Inspected image \mathbf{I}
Output: Binary image $\hat{\mathbf{I}}$ with white pixels representing the defective region
(1) $F(\mathbf{I}) \leftarrow$ Extract LKR features of the inspected image \mathbf{I} by (2);
(2) **for** each pixel $I_i, i \in [1, n]$ **do**
(3) $D_{i,j} \leftarrow$ Calculate the dissimilarity of I_i and its neighbors $I_j, j \in W_i$ by (5);
(4) $w_{i,j} \leftarrow$ Calculate the weight via (7);
(5) $A(I_i) \leftarrow$ Aggregate all $D_{i,j}, j \in W_i$ to generate the anomaly value by (4);
(6) **end for**
(7) $\hat{\mathbf{I}} \leftarrow$ Locate the defects by the thresholding function equation (11).

ALGORITHM 1: The proposed detection algorithm.

4. Experiments

In order to validate the effectiveness of the proposed algorithm, this section gives our experimental results performed on a test dataset that is provided by Linglong Tyre Co. Ltd. This dataset consists of 40 defective sidewall images, 20 defect-free sidewall images, 40 defective tread images, and 20 defect-free tread images, in which different types of the defects like impurity, bubble, and overlap are included. The size of all test images is 256×256 . The performance of the proposed method is evaluated by comparing with ICD-based method (ICDM) [15] and the improved wavelet-based method (IWaveM) [5]. It needs to point out that we improve the performance of the original wavelet-based method [5] by extracting local wavelet features from small image patches and introducing the vertical mean filtering as used in ICDM. These methods are implemented in MATLAB and performed on an Intel-i7 1.73 GHz computer system with 8 GB RAM. Although there are other methods for tire inspection in the literature, we cannot comprehensively quote them in the paper due to the page limitation. In addition, the codes of these methods are not available online. In all comparisons, the scale parameter for ICDM is set to 7, the Daubechies wavelet transform is exploited for IWaveM with four vanishing moments over three decomposition levels, and for the proposed method we utilize a local window with size 5×5 and set the patch size 7×7 , $\lambda = 0.5$, $\tau = 100$.

4.1. Performance Evaluation. The performance of these detection methods mentioned above is quantitatively evaluated by detection rate (DR) [4, 10], precision (P), recall (R), and F -measure (F) [25, 26], which are deeply related to the ROC analysis [27] and, respectively, defined as follows:

$$DR = \left(\frac{N_{tp} + N_{tn}}{N} \right) \times 100\%, \quad (12)$$

$$P = \frac{TPs}{TPs + FPs}, \quad (13)$$

$$R = \frac{TPs}{TPs + FNs}, \quad (14)$$

$$F = \frac{(1 + \alpha) \times P \times R}{\alpha \times P + R}, \quad (15)$$

TABLE 1: Detection rate (%) of different algorithms on test images.

Images	ICDM	IWaveM	Proposed
Sidewall	88.3	90.0	93.3
Tread	N/A	N/A	85.0

where N is the total number of test images, N_{tp} is the number of detected defective images, N_{tn} is the number of detected defect-free images, TPs are the number of true-positive pixels, FPs are the number of false-positive pixels, FNs are the number of false-negative pixels, and we set $\alpha = 0.5$ as in [28]. P corresponds to the anomaly pixels detection performance, while R is the fraction of detected anomaly pixels to ground-truth. As an overall performance measurement, the F -measure is the weighted harmonic mean of P and R [29].

In Table 1, the detection rates of different detection methods for test images are listed for quantitative comparisons. From this table we can observe that for tire sidewall images of the proposed method outperform ICDM and IWaveM in terms of DR; for tire tread images it also achieves high DR. However, ICDM and IWaveM are not applicable (N/A) to detect the defects of tire tread images. To further compare the performance of these detection methods, the average precision, recall, and F -measure on tire sidewall images are shown in Figure 4. It is obvious that the proposed method offers better detection performance than ICDM and IWaveM. In general, ICDM and IWaveM perform well on simple texture images. When the images are with complex structures, they do not work as well. The major reason is that the low contrast and the complicated texture of tire tread images reduce the saliency of defects, which causes them to produce unsatisfactory results.

For visual comparisons, Figure 5 shows the detection results of these detection algorithms on tire sidewall images, respectively. As shown in this figure, all detection methods can locate the defects of tire sidewall images, and the proposed method obviously outperforms ICDM and IWaveM, which is also justified in Table 1 and Figure 4. To demonstrate the effectiveness of our method detecting the defects of tire tread images, Figures 6 and 7 illustrate the anomaly maps of two types of defects (impurity and overlap) of tire tread images and their detection results produced by our method, respectively. It can be observed that the proposed anomaly

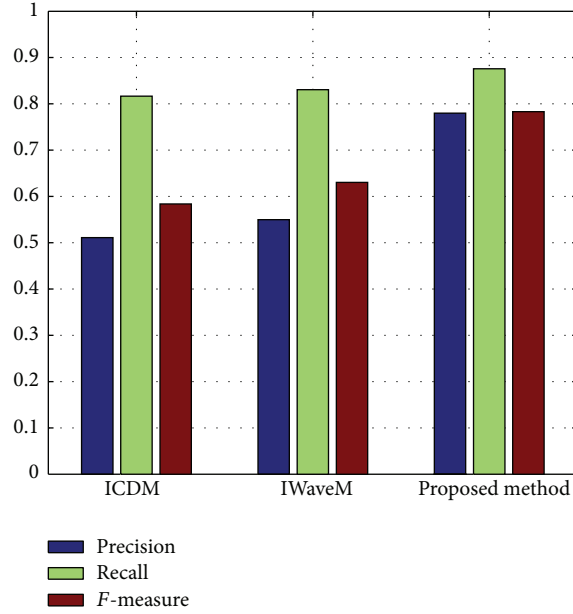


FIGURE 4: Performance evaluation for different methods on sidewalk images.

model is quite sensitive to texture distortion and the saliencies of defects are well highlighted in the anomaly maps.

4.2. Parameters Selection. There are four main parameters in the proposed method, the size of local windows W_i and σ_1 for feature dissimilarity evaluation and the thresholds λ and τ for defect segmentation. The parameter λ is used to segment the anomaly map which describes the feature dissimilarity of image pixels. Therefore, it ranges from 0 to 1. A small λ value is sensitive to noise and may produce false alarms in defect-free images, whereas a large λ value results in a loose control limit and may lead to miss a subtle defect. Figure 8 shows the performance curves of the proposed method with λ varying from 0.1 to 0.9. It can be observed that the proposed algorithm is insensitive to λ in the range [0.4, 0.6]. Thus, we choose $\lambda = 0.5$ in our experiments, which is a trade-off between 1 and 0. For different types of inspected images, the parameter τ can be determined by using a manual tuning method. We find that its performance curves are similar in shape to the ones shown in Figure 8. In all experiments, we empirically set $\tau = 300$ for tire sidewall images and set $\tau = 80$ for tire tread images, which yield a well overall detection performance.

The size of local window W_i has an important influence on the detection performance of our method. The optimal choice of this parameter depends on the size of defects. By observing defective images, we find that the pixels in the vertical direction are very similar and defects locally break the similarity of pixels. Therefore, a rectangular window W_i with the size $(2k + 1) \times 256$ is preferred. In order to evaluate the influence of varying the window size W_i , Figure 9 shows the change curves of three quantitative metrics P , R , and F with k varying from 0 to 10. It reveals that our method works

well when $k = 2$, that is, the size of local window being 5×256 . In consideration of detection effectiveness and computational cost, we choose the size of 5×256 for all experiments. In addition, we find that our algorithm is insensitive to σ_1 , and we empirically set $\sigma_1 = 0.15$.

4.3. Computational Cost. To evaluate the computational cost of three detection methods, we compare the running time on the test dataset. The CPU implementations of ICDM, IWaveM, and our method require 2.6 s, 1.2 s, and 17.3 s on the average, respectively, which are performed on Intel Core i7 3.60 GHz. The proposed method for LKR and the dissimilarity metric is computationally expensive. In general, the LKR step takes approximately 18% of the execution time, whereas 73% of the time is spent in calculating the dissimilarity metric. However, our method is suitable for parallel processing because LKR and the dissimilarity metric for each pixel can potentially be calculated independently. In the practical application, therefore, we can accelerate it on the GPU.

5. Conclusions

Automatic quality inspection is strongly desired by tire industry to take the place of the manual inspection. This paper presents an efficient detection method for automatic quality inspection, which takes advantages of feature similarity of tire images and captures the texture distortion of each pixel by weighted averaging of the dissimilarity between this pixel and its neighbors. Different from the existing tire defect detection algorithms that fail to work for tire tread images, the proposed detection algorithm works well not only for sidewall images but also for tread images. Experimental results performed on

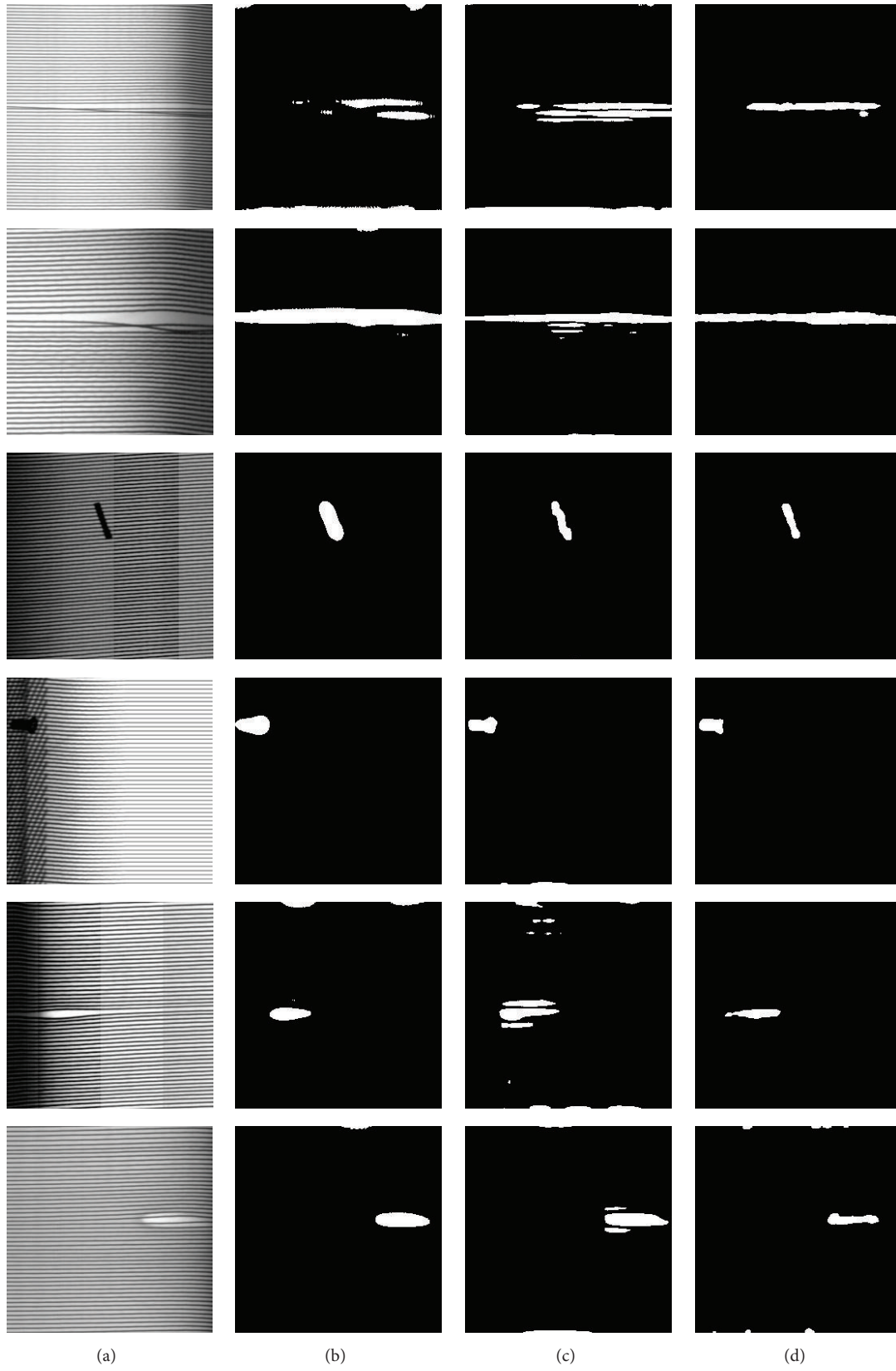


FIGURE 5: Experimental results for different defects of tire sidewall images. (a) Defective images, (b) ICDM, (c) IWaveM, and (d) the proposed method.

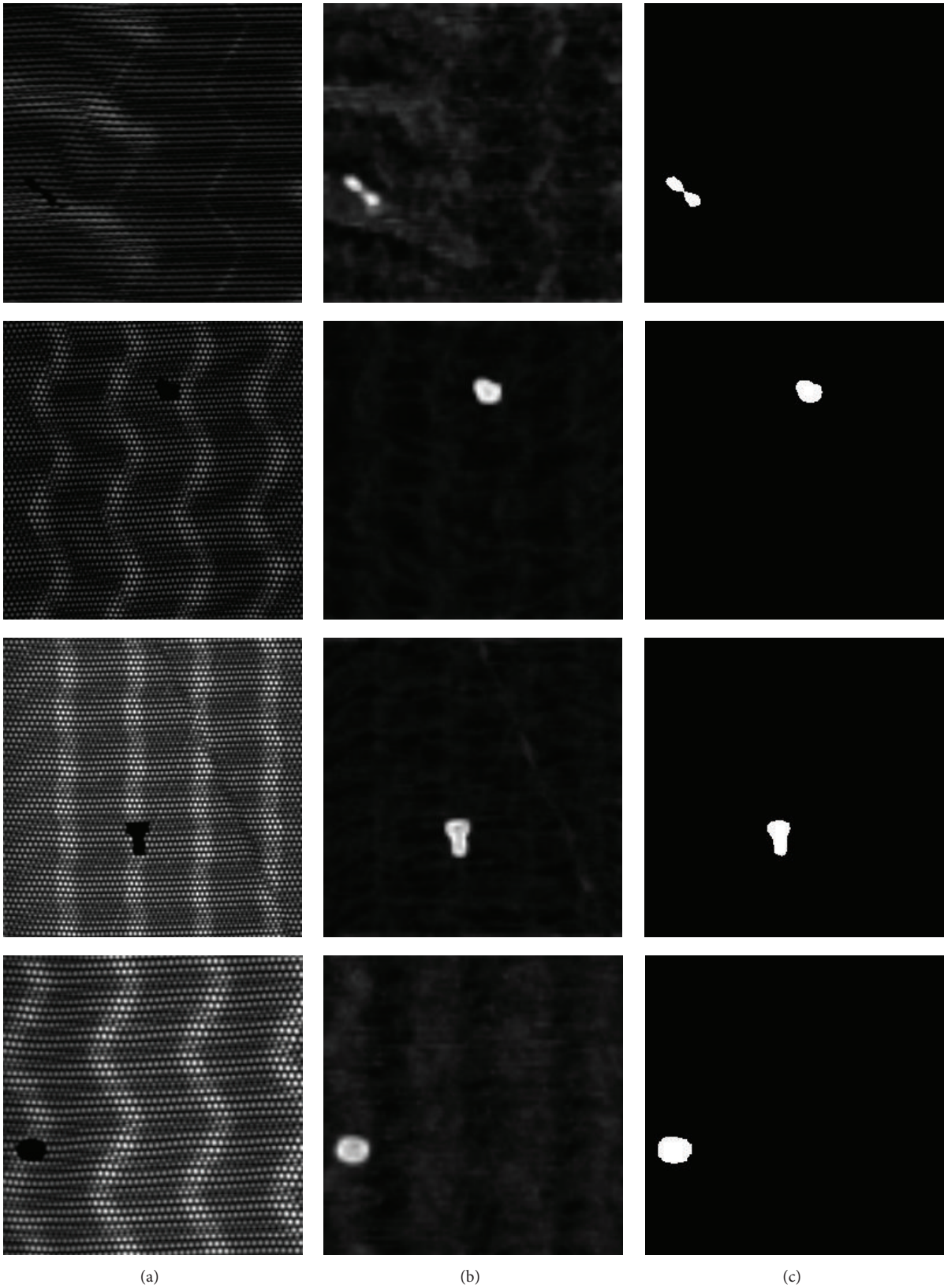


FIGURE 6: Experimental results for impurity defects of tire tread images. (a) Defective images, (b) anomaly maps, and (c) detection results.

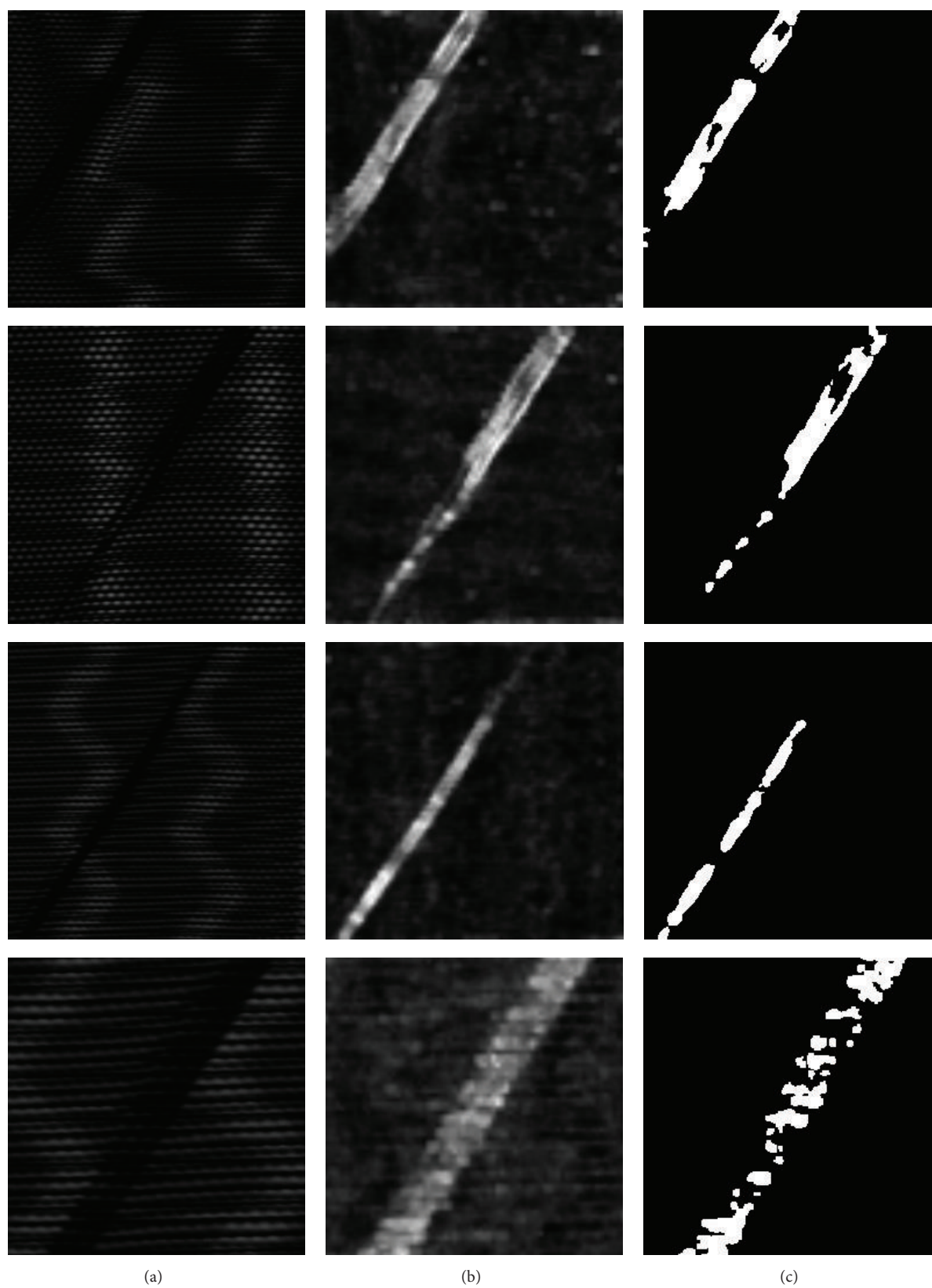


FIGURE 7: Experimental results for overlap defects of tire tread images. (a) Defective images, (b) anomaly maps, and (c) detection results.

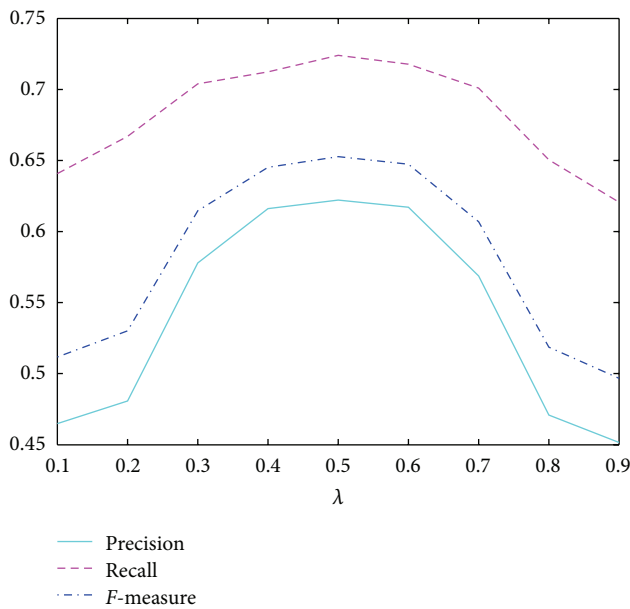


FIGURE 8: Influence of the threshold λ on test images.

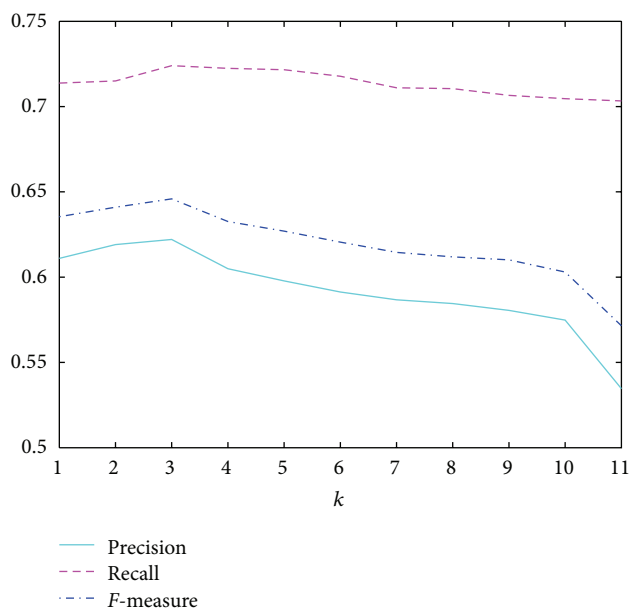


FIGURE 9: Influence of the size of W_i on test images.

real tire inspected images validate that the proposed detection method outperforms the current defect detection methods.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61202150, Grant 61272245, Grant 61373078, and Grant 61472220; Key Research

Development Program of Shandong Province under Grant 2015GGX101004; China Postdoctoral Science Foundation under Grant 2013M531600; and Program for Scientific Research Innovation Team in Colleges and Universities of Shandong Province.

References

- [1] H. Y. T. Ngan, G. K. H. Pang, and N. H. C. Yung, "Automated fabric defect detection—a review," *Image and Vision Computing*, vol. 29, no. 7, pp. 442–458, 2011.
- [2] A. Kumar, "Computer-vision-based fabric defect detection: a survey," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 1, pp. 348–363, 2008.
- [3] A. Latif-Amet, A. Ertüzün, and A. Erçil, "Efficient method for texture defect detection: sub-band domain co-occurrence matrices," *Image and Vision Computing*, vol. 18, no. 6, pp. 543–553, 2000.
- [4] F. Tajeripour, E. Kabir, and A. Sheikhi, "Fabric defect detection using modified local binary patterns," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 783898, 2008.
- [5] D. M. Tsai and B. Hsiao, "Automatic surface inspection using wavelet reconstruction," *Pattern Recognition*, vol. 34, no. 6, pp. 1285–1305, 2001.
- [6] A. Serdaroglu, A. Ertuzun, and A. Erçil, "Defect detection in textile fabric images using wavelet transforms and independent component analysis," *Pattern Recognition and Image Analysis*, vol. 16, no. 1, pp. 61–64, 2006.
- [7] A. Serdaroglu, A. Ertuzun, and A. Erçil, "Defect detection in textile fabric images using subband domain subspace analysis," *Pattern Recognition and Image Analysis*, vol. 17, no. 4, pp. 663–674, 2007.
- [8] A. Kumar and G. K. H. Pang, "Defect detection in textured materials using Gabor filters," *IEEE Transactions on Industry Applications*, vol. 38, no. 2, pp. 425–440, 2002.
- [9] A. Kumar and G. K. H. Pang, "Defect detection in textured materials using optimized filters," *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, vol. 32, no. 5, pp. 553–570, 2002.
- [10] D.-M. Tsai, S.-C. Wu, and W.-Y. Chiu, "Defect detection in solar modules using ICA basis images," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 122–131, 2013.
- [11] J. Zhou and J. Wang, "Fabric defect detection using adaptive dictionaries," *Textile Research Journal*, vol. 83, no. 17, pp. 1846–1859, 2013.
- [12] W. Zhou, M. Fei, H. Zhou, and K. Li, "A sparse representation based fast detection method for surface defect detection of bottle caps," *Neurocomputing*, vol. 123, pp. 406–414, 2014.
- [13] W.-C. Li and D.-M. Tsai, "Wavelet-based defect detection in solar wafer images with inhomogeneous texture," *Pattern Recognition*, vol. 45, no. 2, pp. 742–756, 2012.
- [14] S. Ghorai, A. Mukherjee, M. Gangadaran, and P. K. Dutta, "Automatic defect detection on hot-rolled flat steel products," *IEEE Transactions on Instrumentation and Measurement*, vol. 62, no. 3, pp. 612–621, 2013.
- [15] Q. Guo and Z. Wei, "Tire defect detection using image component decomposition," *Research Journal of Applied Sciences, Engineering and Technology*, vol. 4, no. 1, pp. 41–44, 2012.

- [16] Y. Zhang, T. Li, and Q. L. Li, "Defect detection for tire laser shearography image using curvelet transform based edge detector," *Optics & Laser Technology*, vol. 47, pp. 64–71, 2013.
- [17] Y. Zhang, T. Li, and Q.-L. Li, "Detection of foreign bodies and bubble defects in tire radiography images based on total variation and edge detection," *Chinese Physics Letters*, vol. 30, no. 8, Article ID 084205, 2013.
- [18] Y. Xiang, C. Zhang, and Q. Guo, "A dictionary-based method for tire defect detection," in *Proceedings of the IEEE International Conference on Information and Automation (ICIA '14)*, pp. 519–523, Hailar, China, July 2014.
- [19] C. Zhang, X. Li, Q. Guo, X. Yu, and C. Zhang, "Texture-invariant detection method for tire crack," *Journal of Computer-Aided Design & Computer Graphics*, vol. 25, no. 6, pp. 809–816, 2013 (Chinese).
- [20] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, 2007.
- [21] P. Chatterjee and P. Milanfar, "Patch-based near-optimal image denoising," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1635–1649, 2012.
- [22] Q. Guo, C. Zhang, Y. Zhang, and H. Liu, "An efficient SVD-based method for image denoising," *IEEE Transactions on Circuits and Systems for Video Technology*, 2015.
- [23] Y. Fu, S. Yan, and T. S. Huang, "Correlation metric for generalized feature extraction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 12, pp. 2229–2235, 2008.
- [24] J. D. Peter, V. K. Govindan, and A. T. Mathew, "Robust estimation approach for NL-means filter," in *Advances in Visual Computing: 4th International Symposium, ISVC 2008, Las Vegas, NV, USA, December 1–3, 2008. Proceedings, Part II*, vol. 5359 of *Lecture Notes in Computer Science*, pp. 571–580, Springer, Berlin, Germany, 2008.
- [25] N. İmamoğlu, W. Lin, and Y. Fang, "A saliency detection model using low-level features based on wavelet transform," *IEEE Transactions on Multimedia*, vol. 15, no. 1, pp. 96–105, 2013.
- [26] M.-M. Cheng, G.-X. Zhang, N. J. Mitra, X. Huang, and S.-M. Hu, "Global contrast based salient region detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 409–416, Providence, RI, USA, June 2011.
- [27] J. Davis and M. Goadrich, "The relationship between precision-recall and ROC curves," in *Proceedings of the 23rd International Conference on Machine Learning (ICML '06)*, pp. 233–240, ACM, New York, NY, USA, June 2006.
- [28] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.
- [29] T. Liu, Z. Yuan, J. Sun et al., "Learning to detect a salient object," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 2, pp. 353–367, 2011.

