Surface damage detection for steel wire ropes using deep learning and computer vision techniques

Xinyuan Huang [a], Zhiliang Liu [a,*], Xinyu Zhang [b], Jinlong Kang [a], Mian Zhang [a], Yongliang Guo [a]

[a] School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, China
[b] Glasgow College, University of Electronic Science and Technology of China, China

ARTICLE INFO

ABSTRACT

Steel wire rope (SWR) is of great importance to its many industrial applications. When SWR is damaged, it is likely to result in serious consequences. Therefore, it is important to do research in the field of SWR damage detection. Computer vision-based surface damage detection methods for SWR can operate with high detection accuracy and good adaptation for different types of SWR. Conventional machine learning methods with manual feature extraction have strong subjectivity. If the discriminant information cannot be extracted accurately, the detection accuracy decreases. To address this problem, this paper proposes an intelligent SWR damage detection method, based on a convolutional neural network, which has powerful learning ability and can automatically extract discriminant features by training surface images of the SWR. The experiment results show that the proposed method, based on deep learning, has a higher F1 score and a higher detection speed than four other conventional machine learning methods.

© 2020 Elsevier Ltd. All rights reserved.

1. Introduction

Industrial development has led to the wide use of steel wire rope (SWR) such as transportation, mine hoisting, and tourist ropeways. During the period of using SWR, various damage inevitably occurs, due to the deterioration of its physical properties over time. Therefore, in the practice of engineering, accurate monitoring of the condition of the SWR is essential. At present, the common manual inspection method presents problems of low efficiency and time consumption, as well as inadequate and failed detection. Therefore, automatic non-destructive testing for the SWR is an urgent research subject [1].

Non-destructive SWR testing methods can be classified as follow: the magnetic detection-based method [2], the acoustic emission detection-based method [3], the computer vision detection-based method [6], etc. Magnetic detection mainly uses magnetic flux leakage to judge local damage, such as a pitting fracture of SWR. Acoustic emission detection mainly detects fractures and elastic deformation of SWR that is caused by structural change. Computer vision inspection uses a camera to capture an optical image of the SWR surface and to detect the damage based on the features of fracture. At present, the most mature and widely used method is magnetic detection. Cao et al. [4] proposed a radial basis

function method to identify the electromagnetic excitation signal, and thus achieved its purpose of detecting a SWR fracture. Yan et al. [5] proposed a method for non-destructive testing of SWR, using iron core as a coil winding skeleton. By increasing the number of iron cores, the signal-to-noise ratio of the coil output signal is improved, so that it is easier to analyze damage. Although magnetic detection has some advantages, it is hard to operate, because the equipment structure is complex and results are easily affected by the liftoff value. In addition, for SWRs of different diameters, the same apparatus of the magnetic detection apparatus may not be applicable for magnetic flux leakage detection. Besides, it is difficult to apply the magnetic detection method to SWRs of small diameter (3–6 mm), such as steel cables in aircraft handling systems, because of the special structure and position of SWR [6]. The apparatus used for computer vision detection has a simple structure, and the same apparatus can be applied to SWRs of different sizes; further, the detection result is basically unaffected by the liftoff value. Therefore, computer vision detection has become a potential research direction in the field of non-destructive SWR testing.

Since the end of the 20th century, computer vision detection has been used for non-destructive testing of SWR. Vallan and Molinari [7] proposed a method to process SWR contours that relies on computer vision detection, and this method is applied to cableway inspection. Platzer et al. [8] analyzed and compared several feature extraction methods for SWR anomaly detection, aiming to find the

---

* Corresponding author.
   E-mail address: Zhiliang_Liu@uestc.edu.cn (Z. Liu).

optimal features to obtain better detection performance. Rodner et al. [9] proposed a classification method based on the Gaussian process for one-class classification, which can distinguish between images of defective and normal SWR. The detection result is better than the one-class classification detection method based on support vector data description. Platzer et al. [10] proposed a new strategy, using the hidden Markov model to locate SWR defects, and this method is better than the time-invariant one-class classification method. Sun et al. [11] proposed a method for detecting surface defects of steel cables that is based on support vector data description. This method uses a variety of methods to extract texture defect features, achieving 93% detection accuracy. Yaman and Karakose [12] proposed an elevator rope monitoring and detection method based on image processing and auto correlation which can effectively distinguish serious rope damages. Wacker and Denzler [13] proposed a method to reconstruct a three-dimensional SWR model using two-dimensional wire rope images to identify the SWR fracture. In summary, the above methods must manually extract features, which requires a large amount of prior knowledge, thereby making the feature construction process troublesome. Thus, the adaptability of these methods is poor. Once the processed image and application scenario are slightly changed, the detection performance is reduced [14].

Deep learning provides a solution to the above challenge. Krizhevsky et al. [15] proposed a kind of convolutional neural network (CNN) that can automatically extract features. In the ILSVR-2010 dataset, the classification accuracy of the CNN is much higher than that of conventional machine learning methods. The CNN is a neural network inspired by the visual cortex of animal brains, and its performance on object recognition is better than that of other neural networks. Soukup and Huber [16] applied the CNN to steel surface defect detection and improved the network recognition performance using a normalized method. Liu et al. [17] combined a sparse auto encoder and a deep belief network to form a deep neural network that can classify various cable faults without requiring preprocessing operations for the fault signals. Deng et al. [18] presented an approach of incipient cable failure recognition and classification, based on variational mode decomposition and CNN. Experiments on different classifiers, which include the decision tree, k-nearest neighbor (kNN), backpropagation neural network (BPNN) and support vector machine (SVM), show that the CNN outperforms the other classifiers in terms of accuracy. Zhou et al. [19] proposed a CNN-based health monitoring method to balance tail ropes in hoisting systems. This method focuses on some obvious defects, such as disproportional spacing, twisted rope, broken strand, and broken rope. In engineering practice, incipient damage detection is more meaningful for health management. In the above literature review, we have found some studies on the health monitoring of ropes and cables. However, we have found rare publications that adopt the computer vision detection method based on deep learning for SWR detection. This understanding is further confirmed by a very recent review paper [20], published in 2019. The SWR has unique characteristics (e.g. complex texture), which present new challenges for detection. The existing methods that are proposed for ropes and cables cannot be applied directly to other objects, and their effectiveness cannot be guaranteed for damage detection of SWRs.

In this paper, we propose a method to detect SWR defects that is based on computer vision. The proposed method has the characteristics of being insensitive to noise and background light changes. It is an end-to-end learning process, which is a type of deep learning process in which all parameters are trained jointly, rather than step by step. In this paper, the two "ends" refer to the raw SWR image and the category of SWR surface damage, respectively. The proposed method does not require the design of structural features and has good adaptability for SWR damage detection. In addition,

it can be applied to damage detection for SWRs of different sizes. The experimental results show that the performance of our method is better than that of another four conventional machine learning methods that are based on manual feature extraction.

Contributions of this paper can be summarized as follows.

(1) We explore the use of deep learning theory in SWR damage detection. The proposed method can automatically extract discriminant features from optical images of SWR, and it can provide an end-to-end detection solution.

(2) We design a prototype of the data acquisition apparatus for computer vision- based detection. The apparatus can capture high-quality images of the whole SWR from 360 degrees. The apparatus uses a global shutter to reduce the smear phenomenon and uses guide sleeve structure to stabilize the SWR during inspection. In addition, the slide equipment makes the apparatus adaptive to SWR of different sizes.

(3) We create two surface damage types, broken wire and wear, in the laboratory, and by using the data acquisition apparatus, we establish a data set of the SWR images for the two damage types. The experimental results from this data set show that the proposed method has superiority over the four conventional machine learning methods.

This paper is organized as follows: Section 1 is the background and introduction. Section 2 is the introduction of the CNN fundamentals. Section 3 introduces the proposed method. Section 4 is the experimental study and result analysis. Section 5 discusses the proposed CNN structure and the data acquisition apparatus. Section 6 concludes the paper.

2. The CNN fundamentals

The CNN is a multi-level neural network structure consisting mainly of convolutional layers, pooling layers and fully connected layers. Different activation functions, such as rectified linear units (ReLU) and sigmoid functions, can be used for different layers. The CNN may use dropout and normalization to avoid the over-fitting problem [23], which usually shows a good recognition result in the training set and performs poorly in the test set.

(1) The convolutional layer

The learning task of the convolutional layer for images is mainly realized by the convolution kernel. By controlling the size and step size of the convolution kernel, the transformation of input to output can be adjusted. Convolution kernel parameters and bias values in convolutional networks are usually obtained by random initialization. During the training process, these convolution kernel parameters and bias values are updated by backpropagation. The step size, a hyperparameter that can be set, defines how many pixels at a time are slipped on the convolution kernel in the input image. Setting a smaller step size leads to a larger receptive field, which can retain the feature information of many input images. However, this results in a larger output size and increases the computation cost. Fig. 1 shows the process of slip in the convolutional layer. The red square is the sliding window, and the orange square represents the position of the input where the previous sliding window stayed. The data swept through the sliding window is convoluted with the convolution kernel. After the convolution computation, the corresponding data, along with the bias, is transferred as the output. For demonstration, the bias value is set to zero. The convolution operation refers to the multiplication and addition of the input data in the range of the sliding window and the corre-
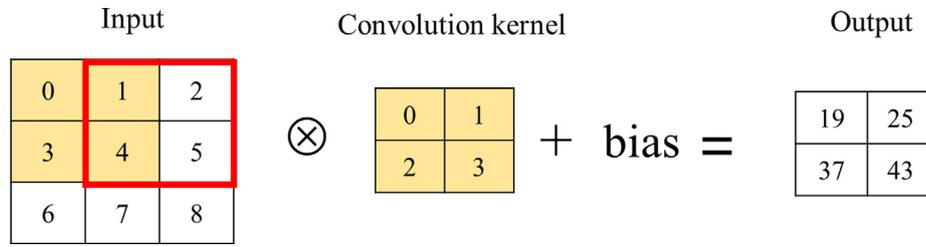
Fig. 1. Process of slip in the convolutional layer.

sponding elements in the convolution kernel. The weight data of the convolution kernel and the bias are learned by CNN training algorithms.

(2) The pooling layer

The pooling layer is used to reduce the size of the input image by down sampling. There are usually two ways to do pooling: mean-pooling and max-pooling. For a sliding window on an image, the max-pooling takes the maximum value of the sliding window. The mean-pooling return the mean value of all elements in the sliding window. As shown in Fig. 2, the data is pooled using a 2×2 sliding window. By consecutively selecting the maximum value in the 2×2 slip window, it can generate the pooling result. Scherer et al. [22] proved that it is better to process images by the max-pooling strategy than by the mean-pooling strategy. Therefore, all pooling layers in this paper are processed by the max-pooling strategy. The red and orange squares in Fig. 2 are similar to the squares described in the convolutional layer. The maximum value in each sliding window is an output element, and the maximum values of all the sliding windows constitute the output.

(3) The fully connected layer

The fully connected layer is composed mainly of multiple neurons of non-linear activation function, and the layer is used to weight and combine the local features acquired by the convolutional layer. As both the convolutional layer and the pooling layer adopt a sliding window strategy, each convolution kernel can only acquire one local feature of the image. However, for image recognition, there is no way to perform overall classification judgment by local features. To obtain the overall feature information, the fully connected layer is required to perform local feature weight combination. There are many options for the linear activation function of the fully connected layer. Nair and Hinton [22] confirmed that the ReLU function has better network performance than other activation functions, and they showed an approximate probability explanation. Therefore, each neuron in this paper is activated by the ReLU function.

When constructing deep neural networks, using more layer connections and more neuron learning units usually result in better learning ability. However, this may also lead to the over-fitting

problem. Often, the more complex the neural network, the more serious the over-fitting problem. To solve such a problem, Srivastava et al. [23] introduced dropout into the neural network. Randomly inactivating the neurons in the training process, according to a certain dropout ratio, can improve the training efficiency and the generalization ability of the model. Ioffe and Szegedy [24] also proposed a batch normalization method to accelerate the speed of deep learning by normalizing all input features. The method also weakens the influence of parameter updates in the former layer on the back layer by limiting the mean and variance of the hidden layer, so that the neural network can reach a deeper structure. Therefore, the neural network proposed in this paper also uses dropout and batch normalization to obtain better results.

3. Proposed surface damage detection method

In this paper, SWR surface damage detection is a three-class classification problem. Given the relative simplicity of the classification task, no complex CNN is required in network design. As such, we use a relatively large number of the convolutional and the pooling layers, and we use a relatively small number of the fully connected layers in networks. In the proposed CNN, the convolutional layers and the pooling layers are used mainly for feature extraction, which is automatic and does not need prior expert knowledge. The fully connected layers use the features to complete the final classification.

Fig. 3 shows the structure of the proposed CNN, where *conv*, *pool*, and *dense* represent the convolutional layer, the pooling layer, and the fully connected layer, respectively. The proposed CNN consists of three pairs of convolutional and pooling layers and two fully connected layers. The specification of the proposed CNN is shown in Table 1. The network structure of this paper is based on [25]. SWR surface damages (e.g. broken wire and wear) are usually local characteristics, so using smaller convolution kernel and smaller step size is more helpful to extract local damage information and improve classification performance. Therefore, the size of the convolution kernel in the proposed CNN is 3×3, and the stride is 1. We use the 'Dropout' function after the 'Flatten' function. The reason for this can be explained as follows. Hinton et al. [26] proved that the over-fitting problem can be reduced by using the "Dropout" function to prevent complex co-adaptations in the training data. On each presentation of each training set, each hidden unit is randomly inactivated from the network with a probability of 0.5 (set the dropout value to 0.5), so a hidden unit cannot rely on the presence of other hidden units. In this way, we achieve the maximum randomness. Thus, during the training phase, we also randomly inactivate 50% of the neurons by setting value of the 'Dropout' to 0.5. Table 2 shows that the proposed CNN attains a very high accuracy (close to 1), which reaffirms why we do not need a complex network in this paper.

The CNN can be regarded as a function $f(\mathbf{X}) = \mathbf{Y}$ that maps the input $\mathbf{X}$ to the output $\mathbf{Y}$. The coefficients of the function are the
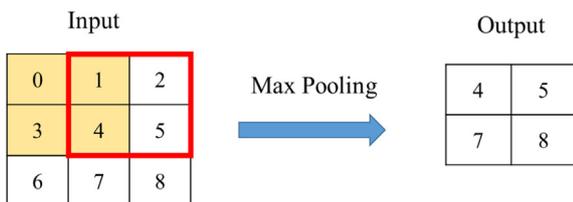
Fig. 2. Process of pooling in the pooling layer.
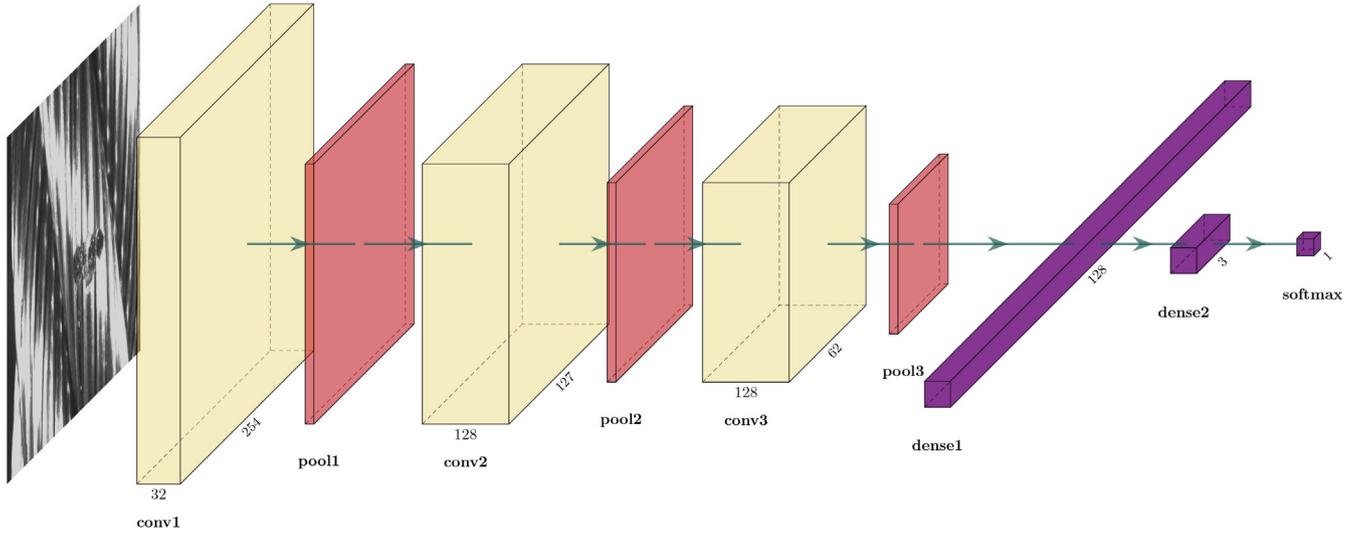
Fig. 3. Structure scheme of the proposed CNN.

Table 1
Specification of the proposed CNN.

Type	Size of slipped/step size or explanation	Input size
Conv1	3 × 3/1	256 × 256 × 3
Batch normalization		
Max Pooling	2 × 2/1	254 × 254 × 32
Conv2	3 × 3/1	127 × 127 × 32
Batch normalization		
Max Pooling	2 × 2/1	125 × 125 × 128
Conv3	3 × 3/1	62 × 62 × 128
Batch normalization		
Max Pooling	2 × 2/1	60 × 60 × 128
Flatten		
Linear	ReLU	1 × 1 × 128
Dropout		
Linear	ReLU	1 × 1 × 3
Classifier	Softmax	1 × 1 × 1

neural network parameters. The convolutional layer calculation in CNN is shown in formula (1).

$$\mathbf{Y} = f(\mathbf{W} \otimes \mathbf{X} + \mathbf{b}), \tag{1}$$

where $\mathbf{X}$ is the input feature map and $\mathbf{Y}$ is the output feature map. The output feature map $\mathbf{Y}$ of the current layer is the input feature map of the next layer. $\mathbf{W}$ is the convolution kernel, $\mathbf{b}$ is the bias vector, $\otimes$ signifies a convolution operation, and f is the ReLU activation function.

The data after convolution must be processed by batch normalization before being passed through the pooling layer. The batch normalization accelerates the learning rate by limiting the distribution of the data. Specifically, the batch normalization operation is shown in formulas (2)–(5).

$$\mu_B = \frac{1}{m} \sum_{i=1}^{m} x_i, \tag{2}$$

$$\sigma_B{}^2 = \frac{1}{m} \sum_{i=1}^{m} (x_i - \mu_B)^2, \tag{3}$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B{}^2 + \varepsilon}}, \tag{4}$$

$$y_i = \gamma \hat{x}_i + \varphi, \tag{5}$$

where x_i is one element in $\mathbf{X}$, μ_B is the mean of the input data, $\sigma_B{}^2$ is the variance of the input data, $\hat{x}_i$ is the result of normalizing the input data, y_i is one element in output data and all results of y_i compose $\mathbf{Y}$ that are inputted for the next layer calculation. ε is a small constant that prevents $\sigma_B{}^2$ from being equal to 0. γ and φ are the two parameters of the learning model. The pooling layer does not require learning parameters because it only consists of down sampling. The calculation of pooling layer is intended to take the maximum value of the sliding window.

The fully connected layer is calculated as shown in formula (6).

$$\mathbf{Y} = f(\mathbf{W}^\mathsf{T}\mathbf{X} + \mathbf{b}), \tag{6}$$

where $\mathbf{W}$ is the weight matrix and $\mathbf{b}$ is the bias vector, $\mathbf{X}$ is the input data, $\mathbf{Y}$ is the output data that are inputted for the next layer calculation, and f is the ReLU activation function.

In the training phase, the CNN must learn the optimal parameters through weight updates. Formulas (7) and (8) define the prediction value and the loss function, respectively. Minimizing the loss function makes the prediction value and the true value as close as possible.

$$\hat{\mathbf{y}} = f(\mathbf{W}^\mathsf{T}\mathbf{X} + \mathbf{b}), \tag{7}$$

$$J = -[\mathbf{y}\log\hat{\mathbf{y}} + (1 - \mathbf{y})\log(1 - \hat{\mathbf{y}})], \tag{8}$$

where $\mathbf{W}$ is the weight matrix and $\mathbf{b}$ is the bias vector, $\mathbf{X}$ is the input data, $\hat{\mathbf{y}}$ is the prediction value of the output, $\mathbf{y}$ is the true value, f is the softmax function, and J is the loss function. Parameters are

Table 2
F1 score and time consumption for the five comparison methods.

	kNN	SVM	Logistic regression	Random forest	Proposed method
F1 Score	0.94	0.96	0.94	0.96	**0.99**
Time Consumption (second)	9.67	8.96	6.07	6.10	**3.71**

updated by the backpropagation chain rule based on the gradient descent.

In general, the deep learning optimization strategy uses the stochastic gradient descent (SGD) method. However, the SGD presents problems, such as slow convergence rate and poor convergence effect. Duchi et al. [27] proposed the Adagrad algorithm to accelerate the learning process. Adagrad improves performance on sparse gradients by preserving the learning rate for each parameter. Tieleman and Hinton [28] proposed the RMSProp algorithm, which adaptively preserves the learning rate, based on the mean of the nearest magnitude of the weight gradient for each parameter. This means that the algorithm has excellent performance on non-steady state problems. Kingma and Ba [29] proposed the Adam algorithm, which combines the advantages of the Adagrad algorithm and the RMSProp algorithm and performs better in the CNN than other algorithms. The Adam algorithm specific parameter update is shown in formulas (9)–(13).

$$V_{dw} = \beta_1 V_{dw} + (1 - \beta_1)\frac{\partial J}{\partial w}, \tag{9}$$

$$V_{dw}{}^{\text{corrected}} = \frac{V_{dw}}{1 - \beta_1{}^t}, \tag{10}$$

$$S_{dw} = \beta_2 S_{dw} + (1 - \beta_2)\left(\frac{\partial J}{\partial w}\right)^2, \tag{11}$$

$$S_{dw}{}^{\text{corrected}} = \frac{S_{dw}}{1 - \beta_2{}^t}, \tag{12}$$

$$w = w - \alpha \frac{V_{dw}{}^{\text{corrected}}}{\sqrt{S_{dw}{}^{\text{corrected}} + \varepsilon}}, \tag{13}$$

where w is one element in **W**, V_{dw} is the momentum variable, S_{dw} is the exponentially weighted moving average variable, and the initial values of both variables are set to zero. $V_{dw}{}^{\text{corrected}}$ and $S_{dw}^{\text{corrected}}$ are the corresponding deviation corrections. β_1 and β_2, two hyperparameters, are set to 0.9 and 0.999, respectively. These two values are recommended by Kingma and Ba [29]. ε is a small constant that prevents the denominator from being 0, t is the number of time steps, and α is the learning rate. The formulas (9)–(13) are used to update the matrix **W** of an n-layer neural network. The procedures to update the other parameters (i.e. b, γ, φ) are similar.

Using the learning rate decay strategy based on the Adam algorithm can, to some extent, improve the performance of the model [30]. Learning rate decay means that the learning rate is reduced in proportion to each new training batch. Specifically, the learning rate decay used in this paper is shown in formula.

$$\alpha_{e+1} = \frac{\alpha_e}{(1+r)}, \tag{14}$$

where α_e represents the learning rate of the current iteration, α_{e+1} represents learning rate of the next iteration, the initial learning rate is set to 0.01, and r is set to 0.0005.

Through the formulas (7)–(14), the parameters are continuously iterated and updated, so that $\hat{y}$ and **y** are gradually approached, and finally, a model with the best classification performance is trained.

Based on the above proposed CNN, the proposed SWR surface damage detection method is illustrated in Fig. 4, where N_{epoch} is set to 20. The proposed method consists of the training phase and the testing phase. The training phase is used to train the CNN model. Once the CNN is established, a new image is fed into the trained network, and it can be classified as one of the three classes.

4. Experimental study

4.1. Experiment setup and data acquisition

As shown in Fig. 5(a), the data acquisition apparatus mainly comprises three parts: an optical camera, a light source, and a computer. The optical camera is a CMOS industrial fixed-focus camera, model of HT-SUA31GC, and the acquired image type is grayscale. A slide rail is arranged under the optical camera to adjust the physical distance between the camera and the SWR, which ensures that the camera can be adapted to SWR with different diameters. The light source uses a monochromatic strip light source with a diffuse plate. The light source and the camera are symmetrically distributed at an angle of 60 degrees. The symmetrical strip light source ensures that, when capturing images, the metal surface reflection is minimized. The computer is used to record and save all captured images during the process of inspection. The power supply is used to power the two 5-Watt light sources. A 10 cm guide sleeve is attached to the holder to prevent the blurring image problem during the movement of the SWR. As shown in Fig. 5(b), this design can effectively reduce the shaking of SWR and can improve the quality of the SWR image.

The flow chart of the data acquisition processes is illustrated in Fig. 6. The five steps can be described as follows: 1) put the SWR into the holder; 2) connect the camera to the computer; 3) open the data acquisition software in the computer, and prepare for data acquisition by setting specifications, such as time of exposure and brightness; 4) energize the light source; and 5) drag the SWR through the data acquisition apparatus for data acquisition.

The SWR used in our experiment has a diameter of 28 mm, a specification of 7×19, and a steel type of 304. Surface damages on this SWR are created by a grinding wheel. The damages differ in depth and width, and they are used to simulate real damages. The data acquisition apparatus moves along the SWR in the axial direction. For the wear damage, we created 38 damage cases with different depths, from 0.7 mm to 3 mm. For the broken wire damage, we created 57 damage cases with different numbers of broken wires, from 2 to 5. To collect more abnormal samples, we drag the SWR to perform data acquisition twice. Each time, there are changes of the position of the SWR relative to the camera for each time, making the captured images different. We collect 2757 images in total:1036 images are from the normal SWR, 699 images are from the worn SWR, and 1022 images are from the SWR with broken wires. The resolution of the images is 240×320 pixels. To prevent the data imbalance problem that may occur during the training process, we randomly select 660 images from every category to establish the training data set, and uniformly scale the images to 256×256 pixels. Fig. 7 gives examples of the captured images.

To improve the generalization ability of the model, we use the function called "ImageDataGenerator" in Keras deep learning framework to realize data enhancement. Data enhancement can make the data set as diverse as possible and may thus prevent the over-fitting problem from occurring in the proposed network. We can use data enhancement to expand the training set. Fig. 8 shows an example of the data enhancement results. In particular, the image used for training and testing is one of the following varieties: the original image, the image after being randomly rotated within 30 degrees of the original image, or the image after being offset to the left/right within 10% of the original image. We randomly select 660 images from each of the three categories for training and testing. From the 660 images, we select 360 images for training and use the rest for testing. We repeat the above steps ten times and record the testing results each time. Finally, we compute the average of the testing results for each of the ten times as the final results.
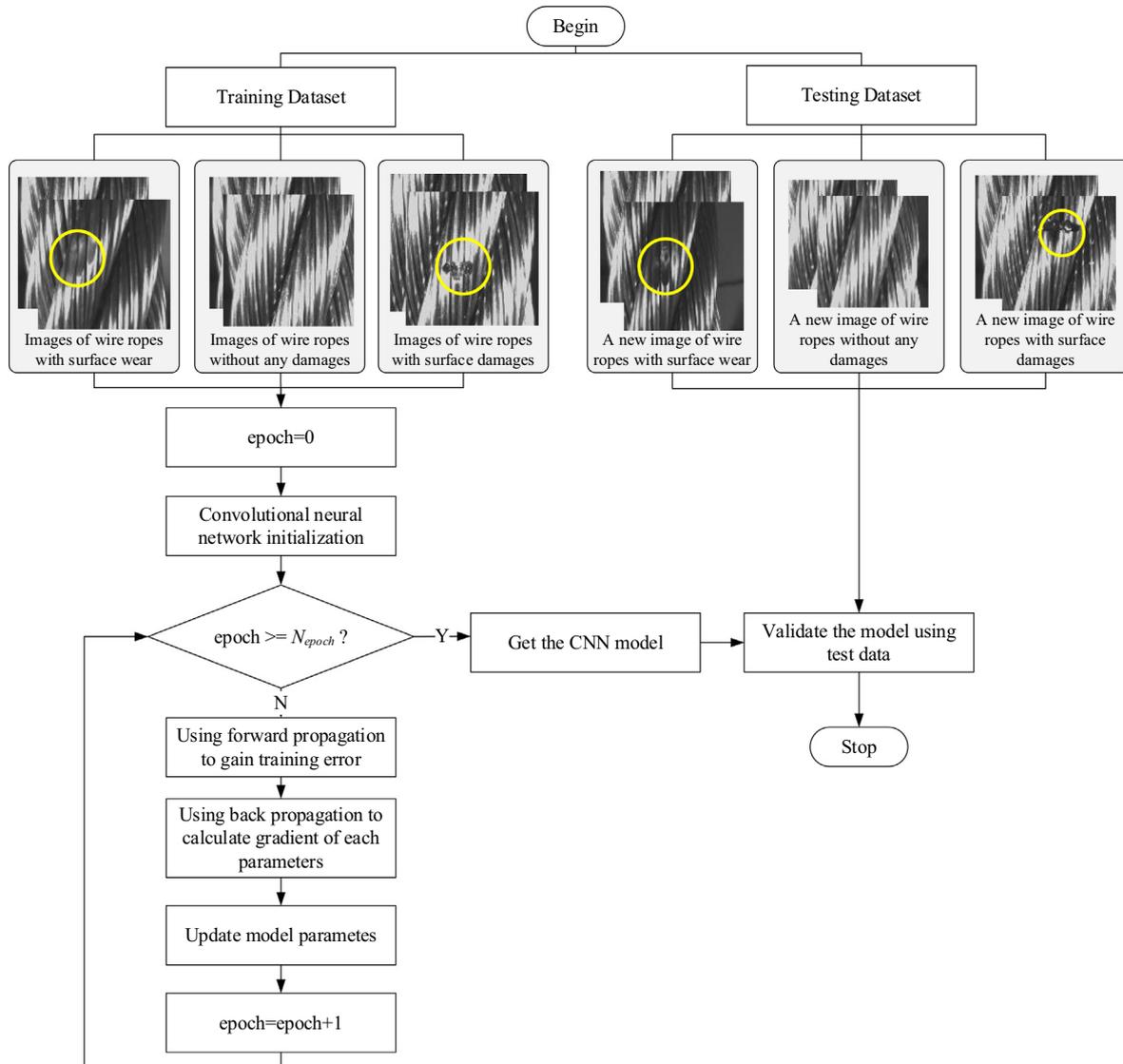
Fig. 4. Framework of the proposed method for SWR surface damage detection.
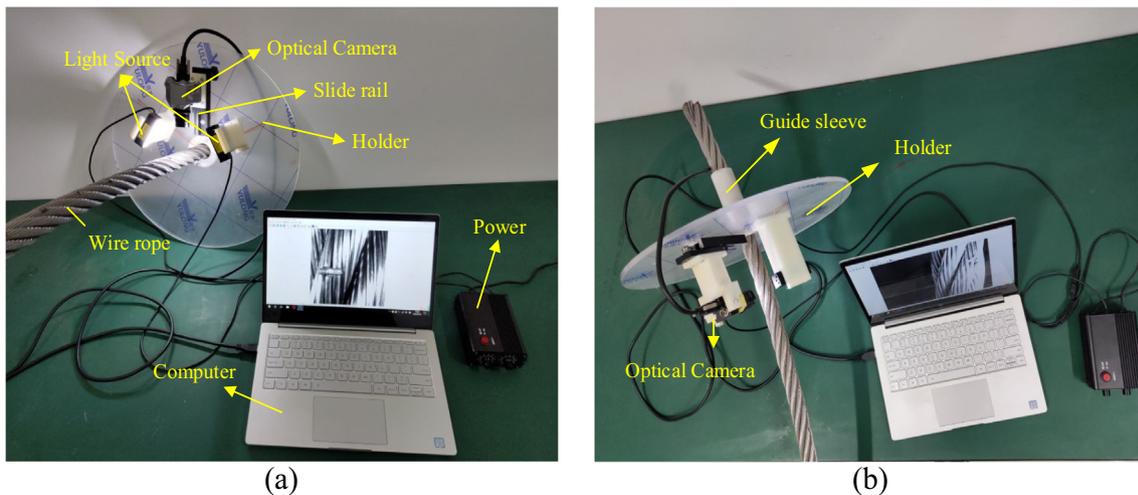
Fig. 5. The data acquisition apparatus from two views: (a) front view and (b) top view.
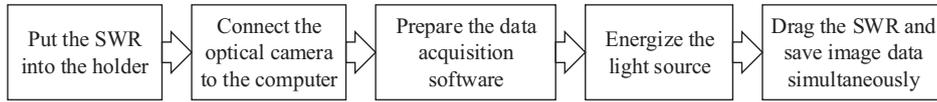
Fig. 6. Data acquisition process flowchart.
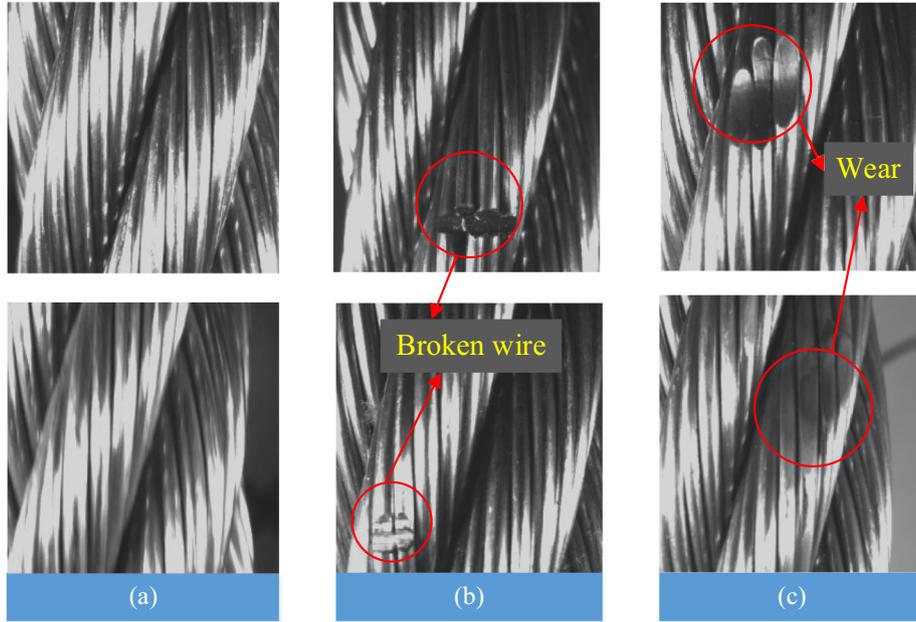
Fig. 7. The SWR images of three fault types: (a) normal, (b) broken wire, and (c) wear.
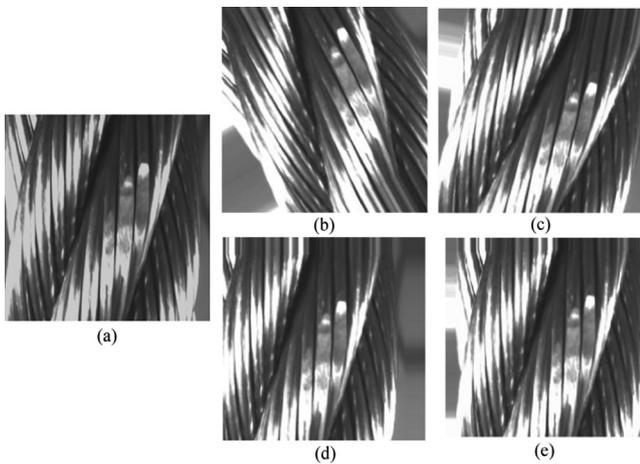
Fig. 8. Data enhancement demonstration: (a) the original image; (b) the image after rotation to the left; (c) the image after rotation to the right; (d) the image after offset to the left; and (e) the image after offset to the right.

4.2. Evaluation metric

The surface damage detection performance is evaluated by F1 score, which considers both precision and recall. The precision refers to the ratio of certain category individuals to total selected population. The recall refers to the ratio of certain selected category individuals to the total individuals in this category. Their definitions are shown in formulas (15)–(17). F1 score combines the precision and the recall to objectively reflect the generalization performance of classification methods.

$$\text{Precision} = \frac{TP}{TP + FP}, \tag{15}$$

$$\text{Recall} = \frac{TP}{TP + FN}, \tag{16}$$

$$\text{F1 score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{17}$$

where TP denotes the number of cases in which the detection is positive and the actual value is also positive, FP denotes the number of cases in which the detection is positive but the actual value is negative, FN denotes the number of cases in which the detection is negative but the actual value is positive, and TN denotes the number of cases in which the detection is negative and the actual value is also negative.

4.3. Detection performance comparison

In this section, we compare performance between the proposed method, which is capable of automatic feature extraction, and the conventional machine learning method, which require manual feature extraction. In this paper, four conventional machine learning methods, kNN [31], SVM [32], logistic regression [33], and random forest [34], are used for the comparison. For conventional machine learning, image features must be extracted prior to the model training and testing. Platzer et al. [8] found that the structural characteristics of the SWR generate some obvious gradient directions in the image. He also compared various feature extraction methods and found that the histogram of oriented gradient (HOG) feature is the best. Therefore, this paper uses the HOG feature as the input for the conventional machine learning methods. The HOG feature extraction uses a detection window unit size of

20×20 pixels, and the number of discrete orientation bins is set to four.

The four conventional machine learning methods are implemented by the Scikit-Learn Machine Learning Toolbox [35], where hyperparameter optimization is also available for the four methods. F1 scores and time consumption of the four methods and the proposed method are shown in Table 2. We first analyze the F1 score results. The experimental results show that the proposed method performs better, in terms of F1 score, than the four conventional machine learning methods. Moreover, even though the proposed method uses a simple network architecture, its F1 score attains the highest value of 0.99. This reflects that the CNN has a strong ability for feature extraction, and the automatically extracted features implemented by the CNN performs better than the manually extracted features developed by the field experts, such as the HOG features. We next analyze the time consumption results. Time consumption refers to prediction time for the test data set, and is collected by a computer with an i5-8500 CPU and GTX 1060-5G GPU. It can be seen from Table 2 that the four conventional machine learning methods take more time than the proposed method. There are two reasons for this: 1) the conventional machine learning methods have no way to use GPU for algorithm acceleration; and 2) the conventional machine learning methods take a lot of time to convert input images to HOG features. In our experiment, converting the input images to HOG features takes 5.74 s.

5. Discussions

5.1. Discussion on the proposed CNN structure

This section explores the performance of the CNN with different structures. We conduct a set of experiments that use different numbers of layers in the CNN. First, a CNN baseline structure named CNN0 is designed, and its network structure is shown in Table 3. The F1 score of the CNN0 is 0.969 and this score is considered to be a baseline performance. It is worth noting that, according to Table 2, even the structure of the CNN0 is simple, the CNN0 outperforms the three conventional machine learning methods, which are kNN, SVM, logistic regression, and random forest.

Based on the CNN0, five networks are generated by consecutively increasing the number of the fully connected layers with 128 neurons with the ReLU activation function. And the F1 scores of the five networks are calculated. Similarly, another five networks are also generated continuously by increasing the number of combinations of the convolutional and the pooling layers to the CNN0. Each combination sets the convolution kernel size to 3×3, the step size to 1, the number of the convolution kernels to 128, and the pooling layer size to 2×2. Then, the F1 scores of these five networks are also calculated.

The F1 scores of the above two groups, of the networks and the CNN0, are shown in Fig. 9. According to the figure, although the F1
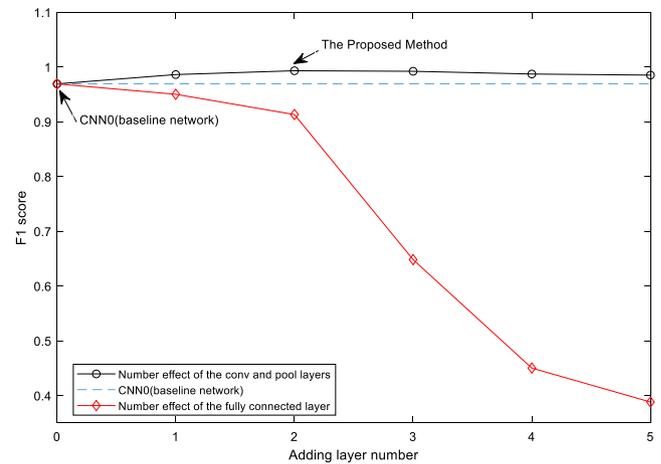
Fig. 9. F1 scores of the CNNs with different structures.

score first increases and subsequently decreases when we continuously add the combination of convolutional and pooling layers to the CNN0, the performance is still improved, relative to CNN0. By contrast, as the number of the fully connected layers increases, the performance drops sharply.

The main function of the fully connected layers is to combine local feature information with global feature information. Thus, it is not necessary to increase the number of fully connected layers. If too many layers are added to the network, it is easy to cause the over-fitting problem. The main function of the convolutional and pooling layers is to extract local surface damage features, and by increasing the depth of the convolutional and pooling layers, allow extraction of high-level features, thereby facilitating network learning and classification. However, when the convolutional layer and the pooling layer reach a certain depth, a further increase of the layers causes no effect on the classification performance. In addition, increasing network depth increases the computational cost, so continuing to increase the depth of the convolutional and the pooling layers after achieving a satisfactory performance is not recommended.

We use different sizes of the training data to explore whether the model is robust. Fig. 10 gives the testing results about the F1 score. In the x-axis of the figure, 60 denotes that we choose 60 images for training and 600 images for testing from each category; 120 means that we select 120 images for training and 540 images for testing from each category, and so on. We repeat the above data selection processes ten times, and we compute the average of the ten testing F1 scores as the final result. According to Fig. 10, F1 scores basically stabilize around 0.99 when the training size reaches 360. In the cases with small training data sizes, the proposed method can still maintain a good detection performance.

To compare the effects of different feature extraction methods, we use the t-SNE method [36] to reduce the feature dimensions. The t-SNE method is a nonlinear dimensionality reduction method, which is very suitable for high-dimensional data visualization. The t-SNE method converts affinities of data points to probabilities. The affinities in the original space are represented by Gaussian joint probabilities and the affinities in the embedded space are represented by Student's t-distributions [35]. When the Kullback-Leibler divergence between the distributions of high-dimensional data and low-dimensional data is minimized, which means that the data points of the high-dimensional distribution have been successfully mapped onto the low-dimensional space. When using conventional machine learning methods, we extract the HOG features of the images. Before entering the classifier for classification, the dimension of extracted features is 3600. When using a deep

Table 3
Specifications of the CNN0.

Type	Size of slipped/step size or explanation	input size
Conv1	$3 \times 3/1$	$256 \times 256 \times 3$
Batch normalization		
Max Pooling	$2 \times 2/1$	$254 \times 254 \times 32$
Flatten		
Linear	ReLU	$1 \times 1 \times 128$
Dropout		
Linear	ReLU	$1 \times 1 \times 3$
Classifier	Softmax	$1 \times 1 \times 1$
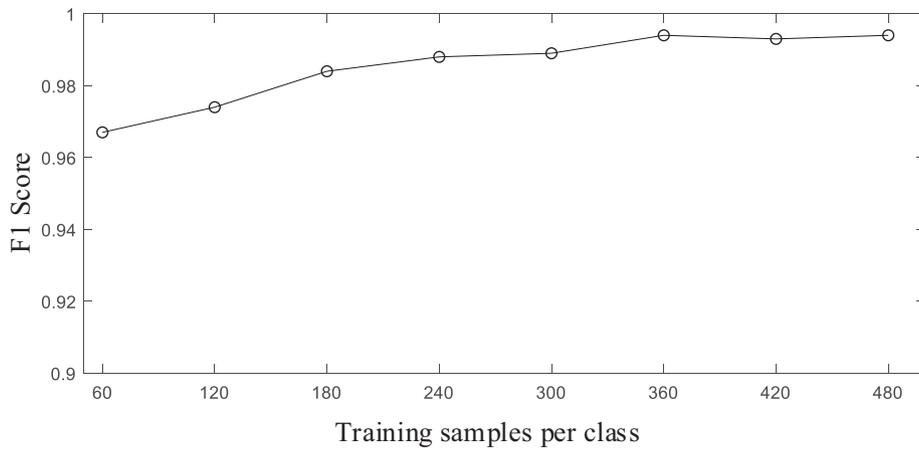
Fig. 10. F1 scores under different numbers of training samples.
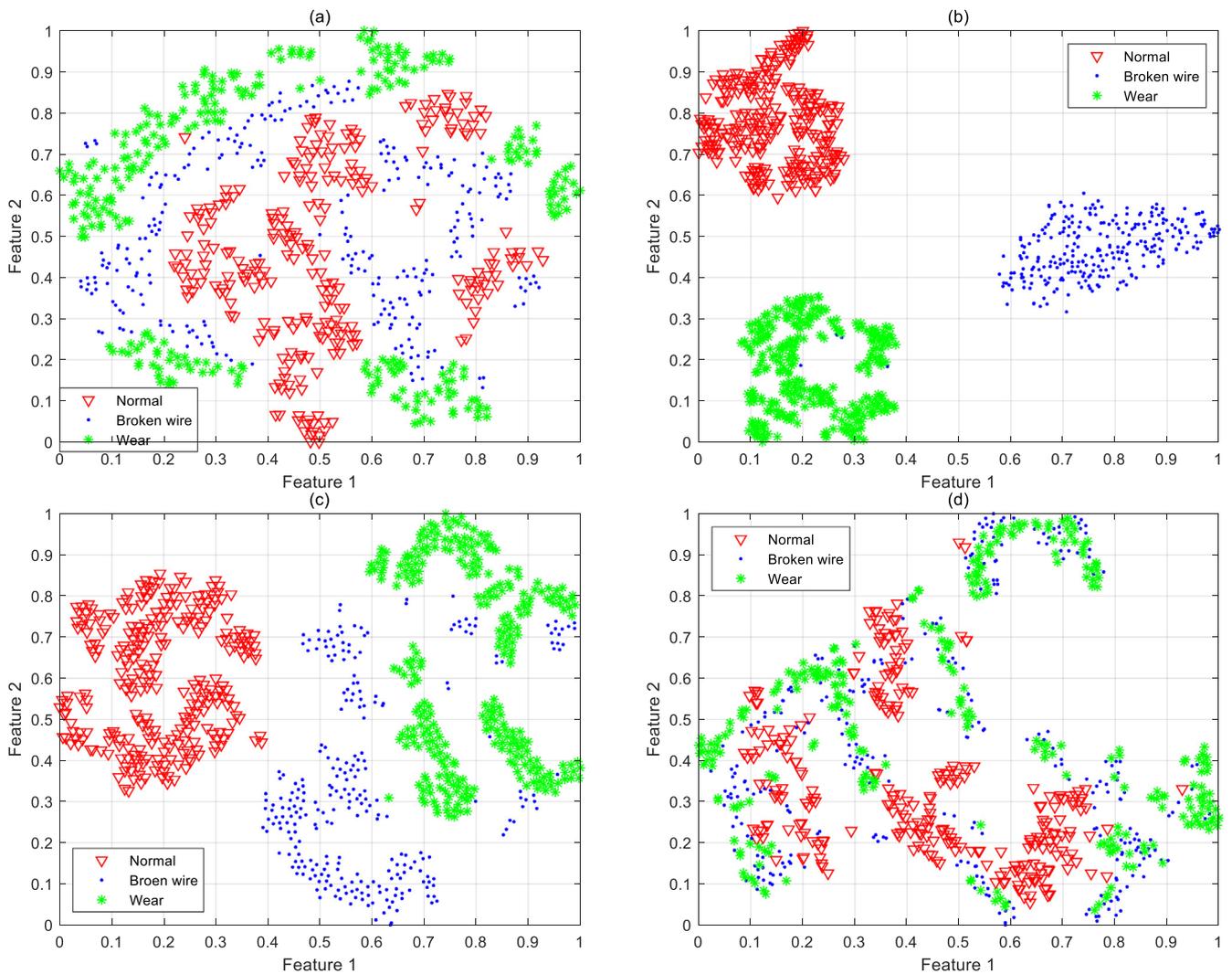
Fig. 11. Group scatter plots of different feature extraction methods: (a) the HOG features, (b) the proposed method, (c) CNN0, and (d) the network that adds 5 layers to the fully connected layer in Fig. 9.

learning method for feature extraction, before entering the classifier for classification, the extracted feature dimension is 128. We use the t-SNE method to reduce features into two-dimensional space for representation. Fig. 11 shows group scatter plots of the

features for the 900 testing samples. The HOG features show high nonlinear characteristics in Fig. 11(a), which leads to a big challenge for the classification. This again confirms the limitations of manual feature extraction methods. According to Fig. 11(b), the

features extracted from the proposed method are highly discriminant for the three classes, and a simple classifier can deal with this three-class classification problem because the boundary is clear. The proposed method performs slightly better than the comparison methods from Table 2; however, the features extracted by the proposed method perform much better than those extracted by the comparison methods from Fig. 11. This again explains the success of our method for SWR surface damage detection. According to Fig. 11(c), the decision boundary blurs, and this issue becomes even worse in Fig. 11(d). It shows that the proposed structure is well tuned.

The above comparison suggests that, in the future, when designing the CNN for surface damage detection of SWR, the depth of the convolutional and pooling layers should be preferentially increased, which help improve the classification performance. Further, the depth of the fully connected layer should be properly selected to prevent the over-fitting problem.

5.2. Discussion on the data acquisition apparatus

The main factors that influence the quality of SWR images are: 1) shaking during the moving of SWR, 2) the image smear phenomenon, and 3) the photo missing problem for the SWR with relatively high moving speed.

As for the problem of shaking during movement of the SWR, we were able to design the piece of equipment that functions like a guide sleeve to stabilize the SWR, and the structure of the guide sleeve is clarified in Fig. 5(b).

As the SWR is moving while capturing images, the image smear problem may occur, which negatively influences detection. To avoid incurring a smear in the process of data acquisition, we use the camera with a global shutter function for data acquisition. Global shutter and rolling shutter are two camera exposure modes. The rolling shutter uses a progressive scanning method for exposure, and the global shutter exposes the entire image. Fig. 12 shows the difference between the two exposure methods. As the rolling shutter exposure relies on progressive scanning, it results in differing exposure times of different rows of pixels. Thus, the problem of image smear occurs when shooting the moving object. But the global shutter exposure does not have such a problem. The camera collects about 45 images per second when we drag the holder at a speed of about 0.2 m/s.

If the SWR moves at a relatively high speed, it is possible to fail to capture all images of the SWR defects. This problem is called the "the photo missing problem" in this paper. By increasing the data acquisition speed of the camera, it may be possible to solve such a problem. The time consumed by the camera for data acquisition is divided into two parts: the first part is the exposure process; and the second part is the readout process, which reads the data from the sensor's registers and transmits it out. There are two types of methods commonly used in the image acquisition process: non-overlapped exposure and overlapped exposure. Fig. 13 shows the
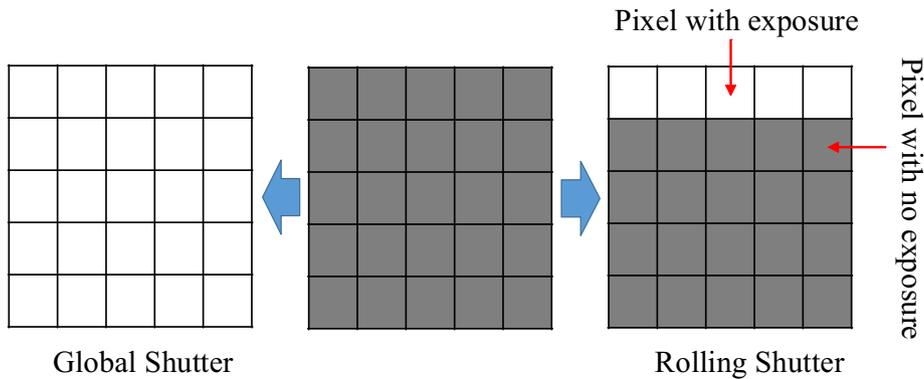
Fig. 12. Demonstration for rolling shutter and global shutter.
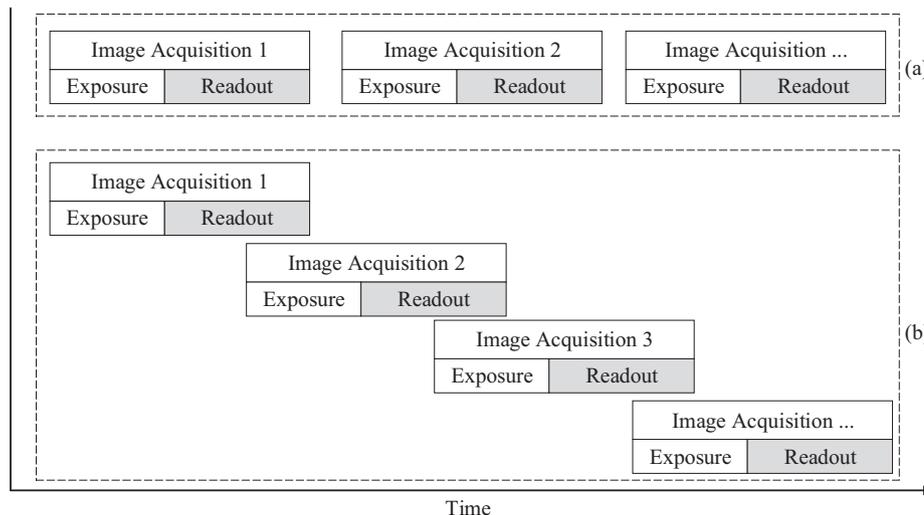
Fig. 13. Two camera exposure modes: (a) non overlapped exposure and (b) overlapped exposure.

difference between the two exposure modes. For the non-overlapped exposure mode, the exposure and readout processes are completed during each image acquisition cycle. For the overlapped exposure mode, the image data obtained in the previous frame can be read and transmitted when the next frame of image begins to be exposed. We use the non-overlapped exposure mode when we are collecting image data. The camera's exposure time is 5 ms and readout time is about 17 ms (45 frame per second). During the readout process, the SWR is about 0.26 cm forward, and the camera's shooting area is greater than 0.28 cm^2. We do this to enable us to guarantee that every defect of the SWR is captured.

In practical engineering applications, the SWR surface inspection may require a faster running speed. One can use a camera that supports the overlapped exposure mode. It is also possible to shorten the exposure time by increasing the light intensity, or use a higher frame rate camera to reduce the time required for the readout process. In summary, we propose reference solutions to alleviate the influence of the three factors on the quality of the SWR image. The quality of the SWR image can also be improved, to some extent, by these solutions.

6. Conclusions

In this paper, we propose a CNN-based method for SWR surface damage detection. We design a data acquisition apparatus to collect a set of images from normal and damaged SWRs. Unlike conventional machine learning methods, the proposed method can automatically obtain the discriminant features from the convolutional and the pooling layers. It overcomes the intrinsic limitations of the manual feature extraction methods. The experimental results show that the proposed method achieves a higher F1 score, of 0.99, than those obtained from the four conventional machine learning methods. In addition, the time consumed by the proposed method is the shortest.

The shortcoming of the proposed method is that the effectiveness of the method has been verified only in the laboratory scenario. Limited by the condition of the laboratory, only two different types of SWR surface defects are investigated, which does not mean that the proposed method can only deal with the two surface defects. The deep learning method has strong generalization and transfer learning capabilities. The proposed method can also deal with other SWR defects, provided that the associated training data set is available for such SWR defects. Moreover, in practical applications, defects not only appear on the surface, but also inside the SWR. The proposed method only focuses on detecting surface damages, and cannot be used to detect inner damages. SWR inner damage detection is a very challenging problem. To use the proposed method to achieve SWR inner damage detection, it is necessary to combine other imaging techniques, such as magnetic flux leakage, ultrasound guided wave, and acoustic emission.

CRediT authorship contribution statement

Xinyuan Huang: Methodology, Software, Writing - original draft. **Zhiliang Liu:** Conceptualization, Supervision. **Xinyu Zhang:** Validation, Writing - original draft. **Jinlong Kang:** Resources, Writing - review & editing. **Mian Zhang:** Resources, Writing - review & editing. **Yongliang Guo:** Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Juwei Zhang, Xiaojiang Tan, Pengbo Zheng, Non-destructive detection of wire rope discontinuities from residual magnetic field images using the hilbert-huang transform and compressed sensing, Sensors 17 (3) (2017) 608.
[2] Xiaolan Yan, Donglai Zhang, Shimin Pan, et al., Online nondestructive testing for fine steel wire rope in electromagnetic interference environment, NDT & E International 92 (2017) 75–81.
[3] N.F. Casey, J.L. Taylor, The evaluation of wire ropes by acoustic emission techniques, British Journal of Non-Destructive Testing 27 (6) (1985) 351–356.
[4] Qingsong Cao, Dan Liu, Yuehai He, et al., Nondestructive and quantitative evaluation of wire rope based on radial basis function neural network using eddy current inspection, NDT & E International 46 (2012) 7–13.
[5] Xiaolan Yan, Donglai Zhang, Fei Zhao, Improve the signal to noise ratio and installation convenience of the inductive coil for wire rope nondestructive testing, NDT & E International 92 (2017) 221–227.
[6] Huixian Sun, Visual detection method and application research based on texture analysis, PhD Thesis, National University of Defense Technology, Changsha, China, 2010.
[7] Alberto Vallan, Filippo Molinari, A vision-based technique for lay length measurement of metallic wire ropes, IEEE Transactions on Instrumentation and Measurement 58 (5) (2009) 1756–1762.
[8] Esther-Sabrina Platzer, Herbert Süße, Josef Nägele, et al., On the suitability of different features for anomaly detection in wire ropes, in: Computer Vision, Imaging and Computer Graphics: Theory and Applications, Springer, Berlin, Heidelberg, 2009.
[9] Erik Rodner, Esther-Sabrina Wacker Platzer, Michael Kemmler, et al., One-class classification for anomaly detection in wire ropes with gaussian processes in a few lines of code, in: Proceedings of the 12th IAPR Conference on Machine Vision Applications, Springer Verlag, Berlin, Heidelberg, 2011.
[10] Esther-Sabrina Platzer, Josef Nägele, Karl-Heinz Wehking, et al., HMM-based defect localization in wire ropes–A new approach to unusual subsequence recognition, Pattern Recognition, Lecture Notes in Computer Science, 5748, Springer, Berlin, Heidelberg, 2009, pp. 442–451.
[11] Sun Hui-xian, Zhang Yu-hua, Luo Fei-lu, Texture defect detection of wire rope surface with support vector data description, 2009 Chinese Conference on Pattern Recognition. IEEE, 2009.
[12] Yaman Orhan, Karakose Mehmet, Auto correlation based elevator rope monitoring and fault detection approach with image processing, 2017 International Artificial Intelligence and Data Processing Symposium (IDAP). IEEE, 2017.
[13] Esther-Sabrina Wacker, Joachim Denzler, Enhanced anomaly detection in wire ropes by combining structure and appearance, Pattern Recognition Letters 34 (8) (2013) 942–953.
[14] Young-Jin Cha, Wooram Choi, Oral Büyüköztürk, Deep learning-based crack damage detection using convolutional neural networks, Computer-Aided Civil and Infrastructure Engineering 32 (5) (2017) 361–378.
[15] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, Imagenet classification with deep convolutional neural networks, Advances in Neural Information Processing Systems 25 (2) (2012), https://doi.org/10.1145/3065386.
[16] Daniel Soukup, Reinhold Huber Mörk, Convolutional neural networks for steel surface defect detection from photometric stereo images, Advances in Visual Computing, Springer, Berlin, 2014, pp. 668–677.
[17] Ning Liu, Bo Fan, Xianyong Xiao, et al., Cable incipient fault identification with a sparse autoencoder and a deep belief network, Energies 12 (18) (2019) 3424.
[18] Jiaying Deng, Wenhai Zhang, Xiaomei Yang, Recognition and classification of incipient cable failures based on variational mode decomposition and a convolutional neural network, Energies 12 (10) (2019) 2005.
[19] Ping Zhou, Gongbo Zhou, Zhencai Zhu, et al., Health monitoring for balancing tail ropes of a hoisting system using a convolutional neural network, Applied Sciences 8 (8) (2018) 1346.
[20] Ping Zhou, Gongbo Zhou, Zhencai Zhu, et al., A review of non-destructive damage detection methods for steel wire ropes, Applied Sciences 9 (13) (2019) 2771.
[21] Scherer Dominik, M ü ller Andreas, Behnke Sven, Evaluation of pooling operations in convolutional architectures for object recognition , Artificial Neural Networks – ICANN 2010 Springer , Berlin, Heidelberg , 2010 , pp. 92–101.
[22] Vinod Nair, Geoffrey E. Hinton, Rectified linear units improve restricted boltzmann machines, Proceedings of the 27th International Conference on Machine Learning (ICML-10) (2010) 807–814.
[23] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, et al., Dropout: a simple way to prevent neural networks from overfitting, Journal of Machine Learning Research 15 (1) (2014) 1929–1958.
[24] Sergey Ioffe, Christian Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift 37 (2015) 448–456.
[25] Karen Simonyan, Zisserman Andrew, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556 (2014).

[26] Hinton Geoffrey E., Srivastava Nitish, Krizhevsky Alex, et al., Improving neural networks by preventing co-adaptation of feature detectors, arXiv preprint arXiv:1207.0580 (2012).

[27] John Duchi, Elad Hazan, Yoram Singer, Adaptive subgradient methods for online learning and stochastic optimization, Journal of Machine Learning Research 12 (2011) 2121–2159.

[28] Tijmen Tieleman, Geoffrey Hinton, Lecture 6.5-Rmsprop, Coursera: Neural Networks for Machine Learning, University of Toronto, Technical Report, Toronto, 2012.

[29] Kingma Diederik P., Ba Jimmy, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980(2014).

[30] Loshchilov Ilya, Frank Hutter, Decoupled weight decay regularization, (2018).

[31] Thomas M. Cover, Peter Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory 13 (1) (1967) 21–27.

[32] Corinna Cortes, Vladimir Vapnik, Support-vector networks, Machine Learning 20 (3) (1995) 273–297.

[33] Adam L. Berger, Vincent J. Pietra, Stephen A. Pietra, A maximum entropy approach to natural language processing, Computational Linguistics 22 (1) (1996) 39–71.

[34] Leo Breiman, Random forests, Machine Learning 45 (1) (2001) 5–32.

[35] Fabian Pedregosa, Varoquaux Gael, Alexandre Gramfort, et al., Scikit-learn: machine learning in python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[36] Maaten Laurens van der, Hinton Geoffrey, Visualizing data using t-SNE, Journal of Machine Learning Research 9 (2008) 2579–2605.