Wear 460-461 (2020) 203477

Contents lists available at ScienceDirect

Wear

journal homepage: http://www.elsevier.com/locate/wear

Optimized CNN model for identifying similar 3D wear particles in few samples

Shuo Wang [a], Tonghai Wu [a,*], Peng Zheng [a], Ngaiming Kwok [b]

[a] Key Laboratory of Education Ministry for Modern Design and Rotor-Bearing System, Xi'an Jiaotong University, Xi'an, 710049, China
[b] School of Mechanical and Manufacturing Engineering, The University of New South Wales, Sydney, 2052, Australia

ARTICLE INFO

Keywords:
Wear debris analysis
Particle surface generation
CNN
3D particle identification

ABSTRACT

Typical wear particles can be considered as distinctive indicators of on-going wear faults in machines. However, the small number of samples has limited the identification accuracy for similar fault particles. Besides, the three-dimensional (3D) characterization of wear particles may face huge challenges due to excessive surface parameters. Focusing on the problems of high similarity particles and few samples, a CNN-based particle classification method is developed with an example set of fatigue and severe sliding particles, which are the product of severe wear of machines. For data reduction, imaged 3D particle surfaces are firstly converted into 2D depth maps without losing surface information. Virtual fault particle images are then synthesized using a Conditional Generative Adversarial Networks (CGAN), according to the particle generation mechanism and distinctive particle features. Furthermore, a non-parametric particle identification model is established with the optimization of the CNN structures and training method, and the network is further optimized with the particle image standardization and the network visualization. Validation experiments reveal that the proposed method can accurately identify all tested fatigue and severe sliding particles with their typical characteristics.

1. Introduction

Fault wear particles contained in the lubricant system carries significant information for the determination of wear severity and wear mechanisms of machines [1–4]. Therefore, wear debris analysis (WDA) can be considered as an effective non-interventional analysis method for the wear-induced fault monitoring and diagnosis. Accordingly, traditional ferrography technique has been applied for over 100 years, and its 2D images have been processed with advanced image processing methods. Still, the success rate in industrial applications is lower than 50%, especially for fault particles with similar shapes [5], such as fatigue and severe sliding particle, which are produced in abnormal wear conditions and often treated as the precursor of on-going failures. This may be attributed to the limited information of the 2D images [6]. Based on increasing demands for a reliable WDA method, there has been a flourishing emergence by shifting from 2D to 3D analysis for acquiring more comprehensive and accurate information on particle morphologies. However, the massive 3D information presents great challenges to the characterization and classification of similar fault particles, which only have ultra-few samples.

Parameter characterization is the foundation of automated particle identification in traditional artificial-intelligent algorithms, and its validity directly affects the accuracy of the type identification. Referring to reported researches, different descriptors have been established from the particle shape and surface morphology [5,7]. The shape-based descriptors have been available to identify well-recognized particles, such as rubbing, spherical and cutting particles, with a nearly 100% accuracy. However, these descriptors may not work for similar particles whose key characteristics exist on the surface, such as fatigue and severe sliding particles. As the effect is concerned, the characterization of similar particles seems to be a big task through 2D images. Early, synthesis texture parameters are extracted with the gray level co-occurrence matrix (GLCM) and principal component analysis (PCA) for the two kinds of particles in ferrograph images [8,9]. However, these parameters character the relationship of surface colors rather than surface morphologies, and their value will be affected by mechanical equipment and oxidation of particles. In view of this, Laser Scanning Confocal Microscopy (LSCM) [10] and Atomic Force Microscopy (AFM) [11] are used to extract 3D features from wear particle surfaces. The introduction of 3D features has overturned the 2D-based characterization system and improved the accuracy of similar particle identification. Nonetheless, there exist over 200 artificially-designed features that contribute to the

* Corresponding author.
E-mail address: wt-h@163.com (T. Wu).

https://doi.org/10.1016/j.wear.2020.203477
Received 14 May 2020; Received in revised form 29 July 2020; Accepted 2 September 2020
Available online 13 September 2020

description of particles from different perspectives [8]. Such excessive parameters will inevitably bring redundant information to the characterization of similar particles.

Great demands for automation are issued for this promising particle analysis technique, especially driven by artificial intelligence algorithms. The commonly-used identification methods are constructed with neural networks [9], fuzzy mathematics [12], or gray theory [8]. These methods use the 2D or 3D parameters as input vectors to automatically recognize particle types, but their accuracy relies heavily on the parameter characterization models. Inspired by the Convolutional Neural Networks (CNN) [13,14], the non-parametric recognition of similar particles is developed and achieve a high recognition rate in ferrograph images. However, this method is still limited due to the surface characterization defect of 2D images. Meanwhile, there are no sufficient real fault particles to train the CNN model.

To address these issues, a non-parametric identification model of 3D similar particles is developed by taking fatigue and severe sliding particles as examples. For the data reduction, the 3D surfaces of particle surfaces are first converted into 2D images without losing the faint surface features. Furthermore, a particle generation model is established based on CGAN to increase fault particle samples for the training of the CNN model. Aiming at the non-parametric identification of similar particles, a CNN-based model is constructed and trained with the optimization algorithm. This model is further improved with the results of the network visualization. The entire framework ensures an effective identification for similar particles in a few real samples. The performance of this methodology is verified using severe sliding and fatigue particles that are produced from a four-ball machine.

The rest of this paper is organized as follows: Section 2 contains the description of the procedures involving the 2D mapping of 3D particle surfaces, fault wear particle sample generation, CNN-based particle identification and improvements on CNN-based identification model. The verification of the proposed method is given in Section 3, followed by discussions in Section 4. The conclusions are presented in Section 5.

2. Materials and methods

2.1. Framework of the proposed method

A new method is proposed for recognizing of similar particles by taking fatigue and severe sliding particles as an example, and the framework is shown in Fig. 1. The idea is to use 3D surface information of particles as the system input and then convert to 2D images for subsequent CGAN and CNN processing. In step 2, an improved CGAN is employed to synthesize fault particles to assist the later identification. Based on the CNN structure, a non-parametric particle identification method is developed, in Step 3, according to the characteristic of fatigue and severe sliding particles. In Step 4, the working principle and improvements of the established model are explored by visualizing the convolution kernel. Further details of the procedure are presented in the sections below.

2.2. 2D mapping of 3D particle surfaces

Compared with 2D images, 3D particle surfaces contain abundant morphology information, but they need to be handled with 3D-based CGAN and CNN models [15,16], which includes a large number of network parameters that are beyond 2D-based models. In addition, excessive training samples are demanded to optimize their structures. Given this, 2D characterizations of 3D surfaces are needed here. 2D mapping methods mainly include the contour map and the depth map. For the contour map, topographic fluctuations and altitudes are represented with loops, in which the points in the same topographic height are projected into a horizontal curve. The depth map adopts the gray value of each image pixel to represent the distance between the object and the imaging plane of the camera. These two mapping methods are applied to characterize 3D surfaces of particles, as shown in Fig. 2.

As can be observed from Fig. 2, the depth map can reflect the particle surface details by changing the gray level of the image, while the contour map can only sparsely represent the particle surface. Therefore, the depth map is chosen as the 2D mapping method of the wear particle morphology for the subsequent particle analysis.

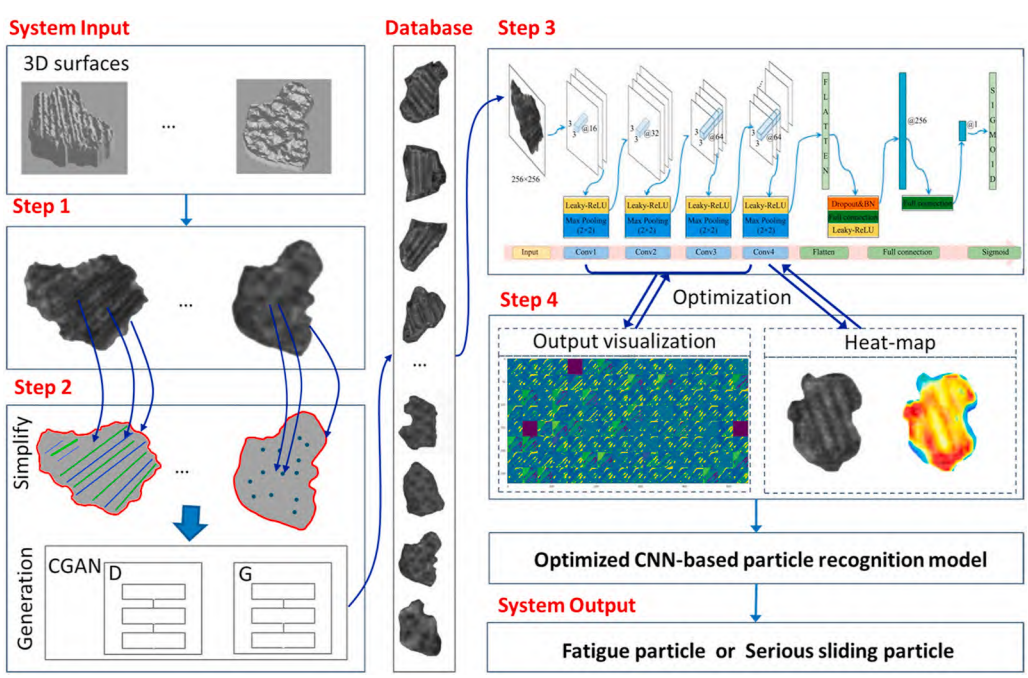

Fig. 1. The framework of the CNN-based particle identification model with few 3D surface samples.

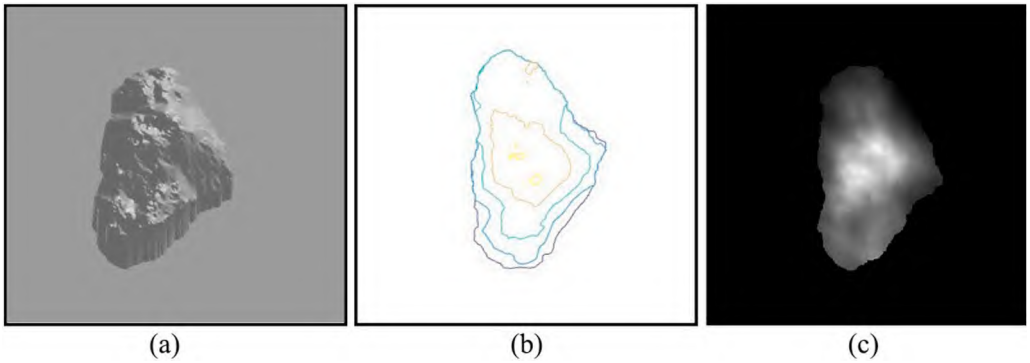

Fig. 2. 2D mapping of 3D wear particle surface: (a) 3D surface, (b) contour map, and (c) depth map.

2.3. Sample generation of fault wear particles

Fault wear particles are usually considered as the failure precursor of mechanical equipment [5]. However, well-designed equipment rarely fails in the actual working condition, thus it is a challenging and time-consuming task to collect typical fault particles. Few particle samples will weaken the generalization ability of the CNN model, which is embodied in the fact that the model works well on the training set, but not for the tested set. Thus, it is necessary to expand the training samples before applying powerful CNN technology to the similar particle identification.

Reported researches [5,7–9,13] show that the surfaces of fault particles have typical characteristics. For example, the surface of fatigue particles owns many pits, while parallel scratches exist on severe sliding particles. Because of this, the fault particles may be generated with these typical surface characteristics. To accomplish this task, the GAN, as a non-parametric generation model, is employed. Among various GAN-based models, such as LSGAN [17] and WGAN [18], CGAN can impose constrains to generate high-quality images with desired features [19]. Thus, CGAN is more suitable to construct the fault particle generation model. This section involves fault wear particle simplification, improvements on CGAN, and performance evaluation of particle generation model.

2.3.1. Fault wear particle simplification

The CGAN takes labeled samples and real images as the input, in which the labeled samples can guide the image generation. Therefore, the labels of particle image need to be first created. As mentioned before, the distinctive features of fatigue and severe sliding particles are pits and parallel scratches respectively [5]. On this account, it may be an effective means to create labeled particle images based on their typical characteristics. The particle simplification process is shown in Fig. 3. Concretely, fatigue particles are simplified with circular colored areas that represent the pits; severe sliding particles are simplified with colored parallel lines for the scratches (blue lines representing low scratches and green lines representing high scratches). Besides, the edges of wear particles are represented by red lines, which are extracted by edge detection. The inner particle areas are marked with gray colors.

2.3.2. CGAN improvements for wear particle images

CGAN is an excellent model for the image generation, which can learn the mapping relationship between the real image r, label data z, and output vector v, i.e. $G:\{r,z\}\rightarrow v$ [20]. The CGAN is composed of the generator and the discriminator, which are alternately trained to improve the similarity between the generated image and the real image. The objective function is shown in Eq. (1).

$$V_{CGAN} = \arg\min_{G}\max_{D} V_{GAN}(G,D) + \lambda\, V_{L_1}(G) \tag{1}$$

where $V_{L_1}(G)$ is L1 loss function, $V_{GAN}(G,D)$ is the loss function of GAN, its expression is:

$$\begin{aligned} V_{GAN}(G,D) = {} & E_{x,y\sim(x,y)}[logD(x,y)] \\ & + E_{x\sim p(x),z\sim p(z)}[log(1-D(x,G(x,z)))] \end{aligned} \tag{2}$$

The specific structure of the model includes:

Generator CGAN's generator adopts the U-Net [21], which contains a symmetric structure: encoder and decoder, to generate the targeted image according to the labeled sample. Notably, the skip concatenation

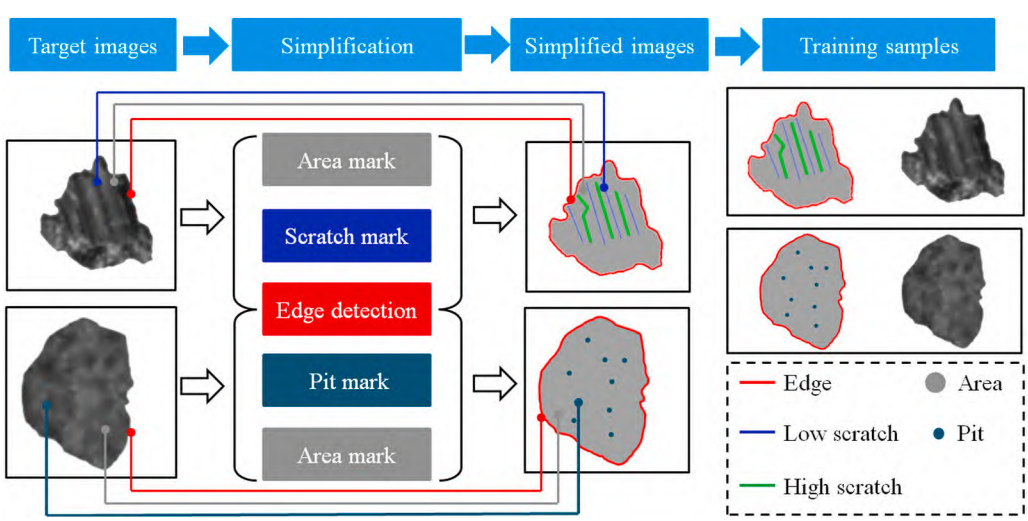

Fig. 3. Diagram of the fault particle simplification.

between the encoder and the decoder can enhance the generated image resolution and reduce the number of training samples.

Discriminator The discriminator is established with Patch-GAN to distinguish the generated and real images [20]. The test image is divided into small patches, and they are identified with the CNN model. The average of all patch results is taken as the discriminator output. This would impose more constraints to highlight the sharp (high-frequency) details of images. With Patch-GAN, the efficiency of the discriminator has been dramatically improved due to the small number of network parameters.

Different from traditional generated images [22], 2D mapped particle images have abundant surface details, but they are gray images without color information. Therefore, further improvements are made on the objective function and the discriminator structure of the CGAN for the particle image generation.

(1) Objective function construction

Objective function describes the errors between the generated image and the actual particle image. A suitable objective function can drive the predicted value gradually approaching the real value. Due to fine surface details in particle images, there is a high requirement for the objective function in the particle image generation. Reference [20] shows that adding loss functions, such as L1 and L2, to the objective function will improve the effectiveness of the CGAN. However, the biggest problem of the L1 loss function is that the gradient is not smooth at zero, which takes a long time to find the extreme point for minute details on particle surfaces. Addressing this, Smooth_L1 is introduced to replace the original L1 loss function. Smooth_L1 is a combination of L1 and L2 loss [23], which takes advantages of the two losses in different intervals, as defined in Eq. (3).

$$S\_L_1 = \begin{cases} 0.5n^2 & if\,|n| < 1 \\ |n| - 0.5 & otherwise \end{cases} \tag{3}$$

For the particle image generation, a new objective function of CGAN is defined as:

$$V_{CGAN} = arg \min_G \max_D V_{GAN}(G, D) + \lambda V_{S\_L_1}(G) \tag{4}$$

(2) Discriminator establishment

CGAN can generate high similarity samples by means of the generator and discriminator against each other, in which the discriminator strives hard to distinguish the generated image from the real one [20]. Nonetheless, the generator network may face gradient disappearance when the discriminator network is too vast. As gray images, 2D mapped particle images contain less color information than colored images, thus the discriminator structure is reduced from the original five layers to four layers, and the size of convolution kernel adopted in each layer is 3×3.The constructed discriminator is shown in Fig. 4.

To enhance the generalization ability, Batch Normalization (BN) is introduced into the convolution layer of the second and third convolution layers of the discriminator [24]. In this way, each mini-batch data will be normalized to the normal distribution of $N(0, 1)$. In addition, Leaky-ReLU [25] is adopted to replace the original ReLU function, which assigns a 0.2 slope to allow all neurons to continue to be updated in the negative interval.

With the above optimization, a new CGAN model is constructed for generating fault particle images: U-Net is selected as the generator; the discriminator is developed with four convolution layers, Batch Normalization and Leaky-ReLU activation function; Smooth_L1 loss function and the original CGAN loss function are combined to construct the objective function of the particle image generation.

2.3.3. Evaluation of particle generation model

The constructed CGAN is trained with the standard approach [20]: we alternate between one gradient descent step on the discriminator and one step on the generator. For the generator training, its weights are updated according to the generated image deviation, the discriminator output, and Smooth_L1. The discriminator weights are adjusted based on the discriminant deviation. The Adam algorithm is selected as the trainer with a learning rate of 0.0002, and the batch size is set to 1 [20]. Twelve groups of severe sliding particles and fatigue particles are collected from the lubrication oils of a four-ball tester, which can simulate the wear process from the start to the end of the machine life [26]. The test ball is manufactured with carbon chromium bearing steel (GCr15). With image rotating, original particle samples are expanded to 48 groups. The CGAN model is trained using these particles and their simplified images, and the training process is shown in Fig. 5. As can be observed, the discriminator and the generator are confronted with each other in the early training. Both of their training losses tend to stabilize after the 300th iteration.

To evaluate the performance of the particle generation model, 100

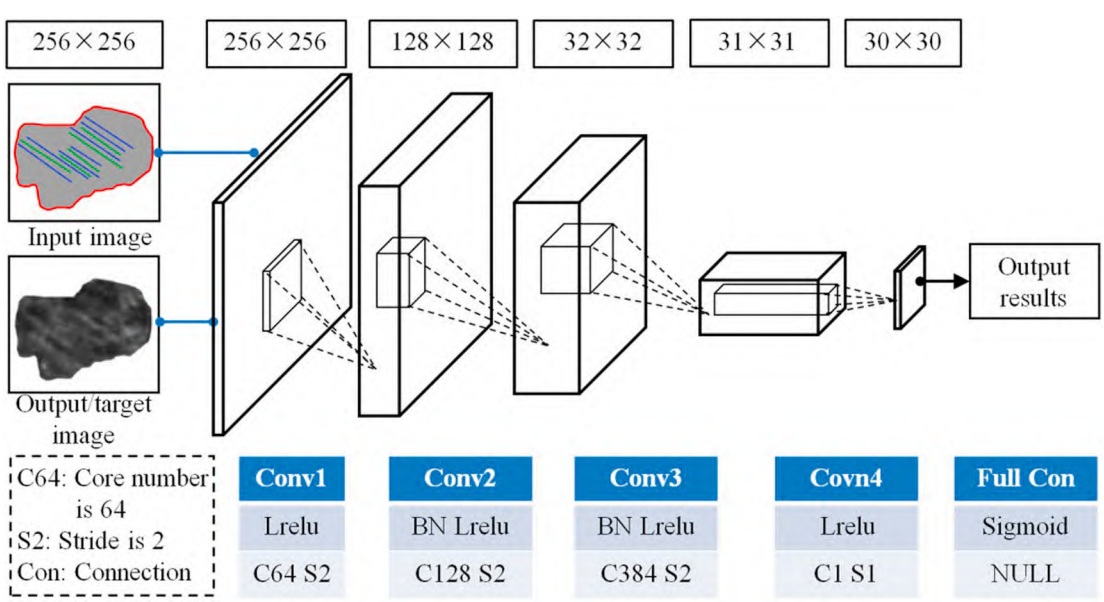

Fig. 4. The parameters of each layer of the discriminator.

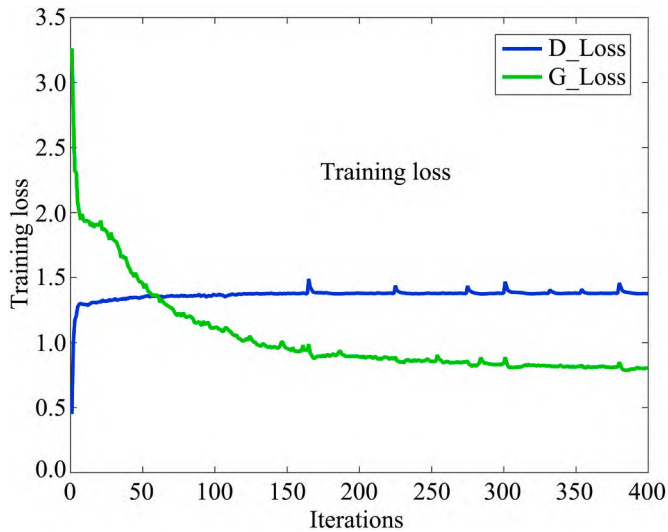

Fig. 5. Adam-based training process of the constructed CGAN model (D_Loss represents the discriminator loss, and G_Loss is the generator loss).

simplified images of each type of particles are first produced by simplifying the typical characteristics of 2D fault particle images. This approach will ensure that the obtained images have a similar characteristic distribution to the real particles. These simplified images are dealt with the trained CGAN model to generate virtual particle surfaces. The average time of creating a particle surface is about 0.8 s. Example results are shown in Fig. 6. As can be observed, the particle features, defined by the simplified images, are produced in the corresponding position at generated surfaces. In contrast, the undefined area is automatically generated by the trained CGAN model. Overall, the generated surfaces remain relatively reasonable.

To further evaluate the generated surfaces, the 3D surface parameters are introduced to characterize the surface morphology quantitatively. Surface arithmetic mean height (Sa) and central liquid retention index (Sci) have been reported for the identification of typical particles [27], thus these two 3D surface parameters (Sa and Sci) are selected to evaluate the generated images. The parameters are extracted from the real and generated surfaces, as shown in Fig. 7. It can be observed that the distribution of parameters of generated particle surfaces is similar to the real ones, but each of them owns different values. This can reveal that the constructed CGAN model can generate virtual particle surfaces according to the simplified images. Given this, the constructed CGAN model can be considered an effective particle image generation method. These generated particle surfaces can be adopted to improve the training performance of the CNN model.

2.4. CNN-based particle identification

The feature-based identification methods have resulted in lower recognition accuracy of fatigue and severe sliding particles than that of other particles, which may be blamed on the characterization incompleteness of artificial-designed features for similar particles. For example, Sa and Sci cannot be adopted to distinguish the fatigue and

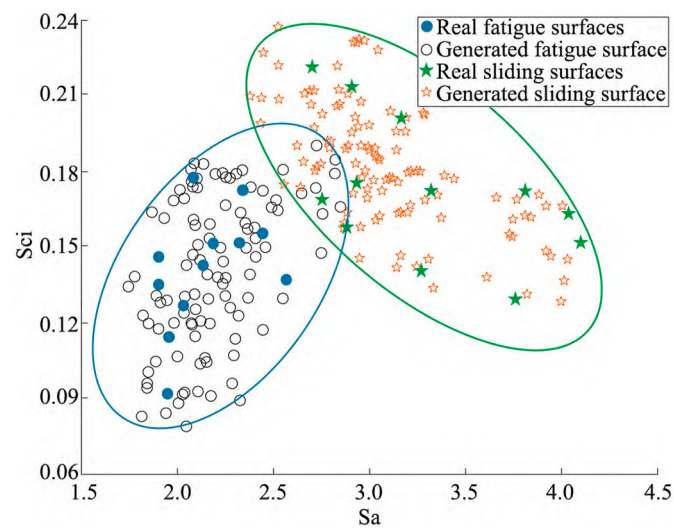

Fig. 7. Surface parameters of real particles and generated particles.

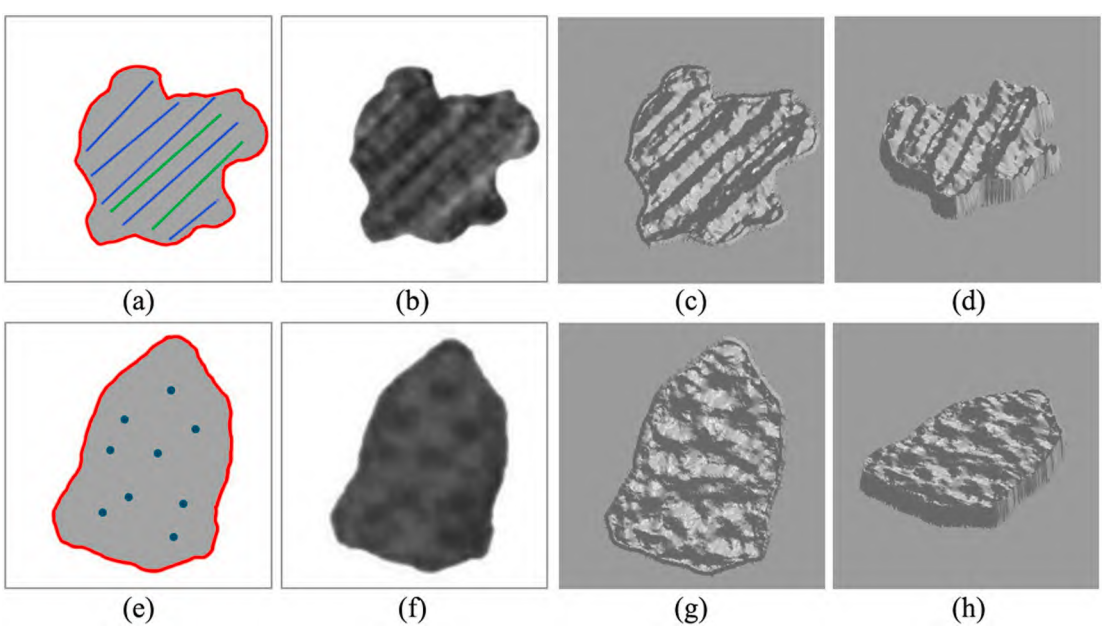

Fig. 6. Particle of generated particle samples: (a) simplified sliding particle, (b) generated sliding particle, (c) front view of the 3D surfaces converted from (b), (d) oblique view of the 3D surfaces converted from (b), (e) simplified fatigue particle, (f) generated fatigue particle, (g) front view of the 3D surfaces converted from (f), and (h) oblique view of the 3D surfaces converted from (f).

severe sliding particles in the interface area in Fig. 7. To address this issue, a non-parametric recognition method is constructed with CNN, including 1) CNN construction for similar particle recognition, 2) loss function selection, and 3) CNN training with the parameter optimization method.

2.4.1. CNN construction for similar particle recognition

Referenced by typical CNN models [28,29], the CNN model mainly consists of three basic layers: convolution layer, pooling layer, and full connection layer. The function of the convolution layer is to extract image features by local receptive fields and weight sharing. As the depth of convolution layers increases, convolution kernels gradually extract high-level features. The pooling operation can reduce the number of free variables and the size of feature maps. With the full connection layer, the local features in previous layers are transferred to the classifier to recognize similar particles. The structure of the constructed particle classifier is shown in Fig. 8.

In view of that the main difference between severe sliding and fatigue particles remains on the surface texture, a four-convolution-layer CNN can be considered as a moderate structure for extracting sufficient surface textures. The particle height images are the model input. Similar to the image feature extraction approach reported in Ref. [30], convolution layers use a filter of 3×3 to convolve the input tensor to obtain the output tensor. Then these tensors are inputted to the Max-pooling layer, which can retain more texture information than the mean pooling. After convolution and pooling operation, a flatten layer and two full connection layers are applied to turn the image information to a feature vector of 1×1. The sigmoid classifier is introduced to output the possibility of particle types.

In addition, the dropout layer [31] and BN layer are introduced to the constructed CNN model to improve the performance of CNN. In this work, the dropout ratio is selected as 0.5, which means that the activation value of a neuron will stop working with a probability of 0.5, so that the model will not depend on local characteristics.

2.4.2. Loss function selection

Loss function can be considered as a measurement of training errors. Choosing an appropriate loss function can improve the recognition accuracy. Mean square error function and cross-entropy function are two kinds of common-used loss functions [32]. Among them, the cross-entropy function can accelerate the CNN training when cooperated with the sigmoid function. Considering that the identification of fatigue and severe sliding particles belongs to a two-class problem, binary cross-entropy is chosen as the loss function. Its principle can be described as in the following.

For the sample (x,y), x is the corresponding label of the sample y. In the binary classification, the probability of belonging to a certain class is in $\{0,1\}$. Assume that the real label of the sample is y_t and the probability of $y_t = 1$ of the sample is y_p, the loss function can be defined as:

$$log(y_t|y_p) = -\left(y_t \times log(y_p) + (1-y_t) \times log(1-y_p)\right) \tag{5}$$

2.4.3. CNN training with parameter optimization method

CNN training is the process of updating and adjusting the initial values of the parameters according to the error between the actual output value and the expected value. To solve this unconstrained problem, many iterative optimization algorithms have been proposed based on gradient descent (GD), such as SGD, RMSRrop, Adam, and so on [33–36]. To determine a suitable training method for the particle classification, these GD-based methods are adopted to train the constructed CNN model, respectively. The sample database consists of real and generated particles, involving 110 severe sliding particles and 110 fatigue particles. One hundred groups of severe sliding and fatigue particles are randomly selected as training samples, and the rest are test samples. Training samples are further expanded from one to five by image flipping, shifting, and amplification [37] to provide sufficient images for CNN training. The learning rate of the optimizer is 1×10^{-4}, and the training process is shown in Fig. 9.

As can be observed, the training process with AdaDelta, SGD and Adagrad may face a low learning rate, learning rate disappearance and slow convergence, failing to reach high accuracy. Inversely, RMSProp, Adam and NAdam have an effective convergence speed. Moreover, Adam has a faster learning speed in the early stage of the training than the other two algorithms. Therefore, Adam is chosen as the optimization method to train the constructed CNN model. The Adam-trained model is adopted to identify fatigue and severe sliding particles, as shown in Fig. 10. The result indicates that the model can accurately identify all tested particles.

2.5. Visualization and improvements on CNN identification model

The application of CNN is driving the parameter-based particle identification to a non-parametric one. But the inner of the constructed CNN model is like a "black box", its working principle cannot be explored. To address this issue, the visualization of the CNN model is studied to explain the effect of convolution layers. Further improvements are applied to enhance the CNN-based particle identification model.

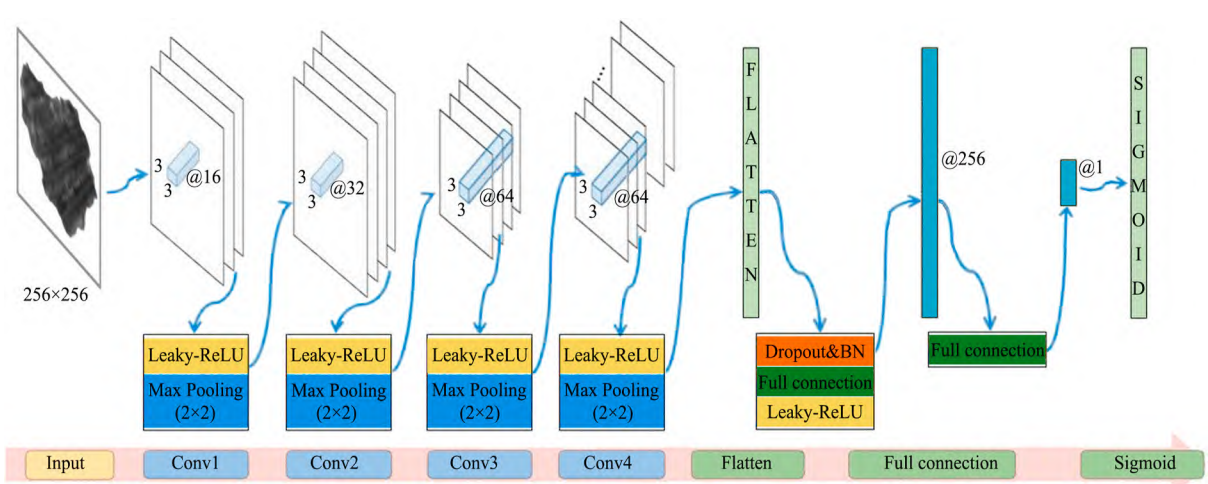

Fig. 8. The structure of the CNN-based wear particle identification model.

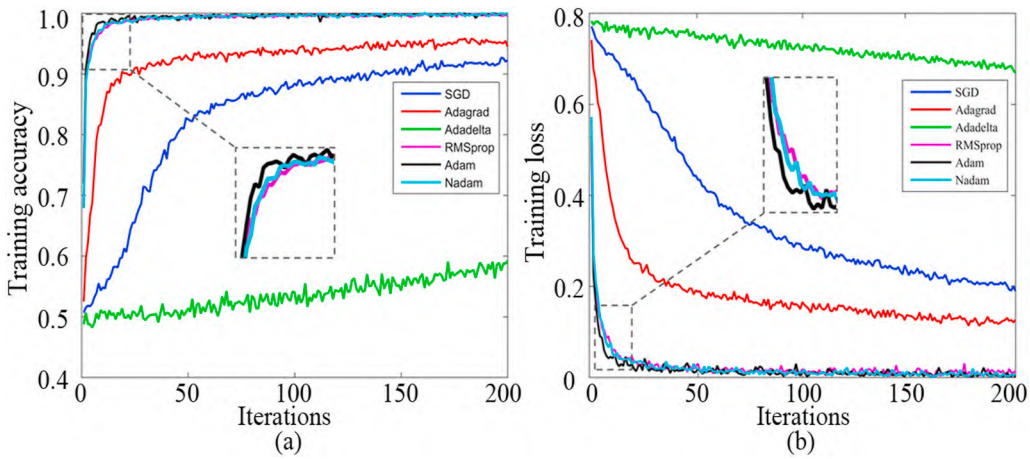

Fig. 9. The CNN training process with different optimization methods: (a) training accuracy, (b) training loss.

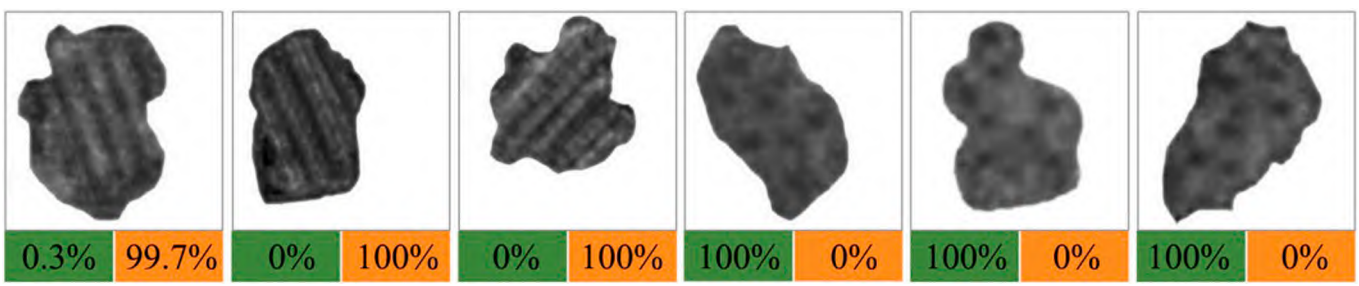

Fig. 10. The identification results of fault particles (Note that the number in the green area indicates the probability that the particle belongs to the fatigue type and the number in the yellow area for the severe sliding type). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

2.5.1. Visualization of CNN convolution kernel

(1) Output visualization of convolution kernel

With the Adam training, each convolution layer has possessed unique feature extraction abilities. The role of each convolution kernel in the CNN identification model can be explored by checking the kernel output with the fatigue or severe sliding particle images. As the convolution layer closely connected with the classifier, Conv4 provides significant information for particle identification. Thus, Conv4 is visualized through the sixth image (a fatigue particle) in Fig. 10. The result is shown in Fig. 11.

As can be observed from Fig. 11, there are several texture feature maps embedded in the fatigue particle pits, which proves that the four-convolution-layer CNN model can extract fine texture features for the characterization of similar particles. However, Conv4 extracts various edge feature maps, even exceeding the number of texture feature maps. This phenomenon reveals that the constructed CNN identification model pays more attention to the particle edge than textures, and it may not apply the essential features to distinguish severe sliding particles and fatigue particles. In addition, many similar feature images exist on the output of Conv4, which means their function may be similar. It can be concluded that the ability of the constructed CNN is far beyond identifying two similar particles.

(2) Heat-maps of class activation

The kernel output visualization proves that the constructed network can extract a small number of critical feature maps for similar particles, but cannot explain which parts of the tested particle image play a

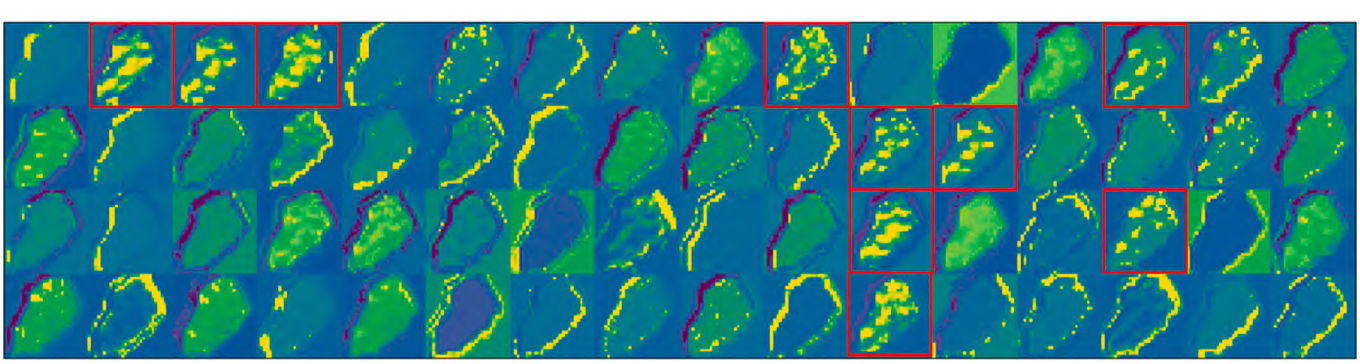

Fig. 11. Output visualization of convolution kernel of Conv4 for a fatigue particle image (Note: texture feature maps are marked in red box). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

dominant role in its classification. To further understand the constructed CNN-based particle identification, Grad-CAM is introduced to show the recognition results [38]. With Grad-CAM, the discriminative regions of the classification are calculated by weighting the importance of each channel of the output feature maps to the category, as described in Eq. (6). As mentioned before, Conv4 is closely connected to the classifier, thus Conv4 is processed to highlight the discriminative regions in the particle classification.

$$Y^c = \sum_k \omega_k^c \frac{1}{Z} \sum_i \sum_j A_{ij}^k \tag{6}$$

where, Z is the number of pixels in the feature map, Y^c is the importance of given category c, and A_{ij}^k refers to the activation at location (i,j) of the feature map A^k, ω_k^c is the weight connecting the kth feature map with the cth class.

Fig. 12 is a set of heat-maps of class activation to explain the classification results of the constructed CNN model in Section 2.4. As can be observed, the heat-maps for severe sliding and fatigue particle images mainly concentrate on the edges rather than the scratch area or pits. This phenomenon reveals that the discriminative regions are not adopted to identify similar particles in the trained CNN model. On account of this, the constructed CNN model cannot be considered an effective particle identification method, even though it has accurately distinguished severe sliding and fatigue particles.

2.5.2. Optimization on CNN-based particle recognition model

(1) Training sample standardization

CNN can distinguish the main features of different types of images, but it is sensitive to data features. With further checking of Fig. 10, there are apparent differences in the average gray between severe sliding and fatigue particle images. As a result, the constructed CNN model uses the gray feature to distinguish these two kinds of fault particles instead of discriminative features (scratches or pits). To address this issue, the training sample images are standardized with the average gray and variance-based method, which can be expressed as:

$$D_0[i,j] = \frac{\sigma_0}{\sigma}(D[i,j] - u) + u_0 \tag{7}$$

where σ_0 is the target variance, σ is the image variance, u_0 is the target mean, u is the image mean, $D[i,j]$ represents the gray value of the pixel at $[i,j]$.

Variables u_0 and σ_0 are acquired by calculating the average gray and variance of all images in the fault particle database. With the standardization process, all particle images possess the same mean and variance, i.e., $u_0 = 68.598$ and $\sigma_0 = 38.943$. Part of the standardized particle image is shown in Fig. 13. As can be observed, the key features of severe sliding and fatigue particles are enhanced, while the non-detected feature in the image has been weakened or eliminated, such as the average gray difference of the two types of particle images.

(2) Structure optimization of CNN model

The output visualization of convolution kernels has shown that the structure of the constructed CNN model is too redundant for identifying two kinds of particles, which will lower the efficiency of the CNN-based particle identification model. In the trained CNN model, the unnecessary kernels cannot be directly removed; otherwise the network connection will be destroyed. One efficient method is to reduce the convolution kernel and re-train the network. Referring to the number of redundant kernels in the CNN model in Section 2.4, the number of convolution kernels is reduced to construct Model 2. Model 3 is deduced by further decreasing the convolution kernel in Model 2. The number in the first layer remains unchanged for the diversity of feature images extracted in the following three layers. The number of convolution kernels per layer is shown in Table 1.

The optimized CNN models are trained by using the Adam algorithm, and the training process is shown in Fig. 14. Similar to the Adam training process in Fig. 9, the identification accuracy of optimized CNN models gradually converges to 1 as the iteration number increases. Importantly, the difference is that these models spend more iteration to achieve the maximum classification accuracy. This can be contributed to the standardization of fault particle images, which diversifies the features of fault particle images.

Fig. 14 also reveals that the learning rate of the CNN model decreases with the reduction of the number of convolution kernels. Although the learning rate of Model 2 is slow in the early stage, it starts to possess the same accuracy with Model 1 at the 40th iteration, and the accuracy of the two models converges to 1 at about 130th iteration. With fewer convolution kernels, the training time of Model 2 is reduced from 703 min of Model 1 to 584 min. In Model 3, due to the over reduction of the convolution kernel, its learning rate is low and the iterative accuracy is only 0.99 when iterating 200 times.

As shown in Fig. 15, the three CNN models are visualized with inputting the same fatigue image of Fig. 11. As can be observed in the visualization of Model 1, owing to the standardized particle images, more attention is placed on texture features in the feature maps of Conv4, which can be deduced from the number of texture feature maps in Figs. 11 and 15. Unfortunately, Model 1 still owns many similar convolution kernels. With the simplification of the CNN model, similar convolution kernels in Conv4 in Models 2 and 3 are greatly reduced, and the critical feature images (including pits) can be extracted for fatigue particles with Conv4. Therefore, the image standardization enhances the classification features for severe sliding and fatigue particles, and the CNN model is simplified by removing redundant convolution kernels without losing key features for the characterization of fault particles.

Fig. 16 is the heat-maps of class activation of the trained CNN model with fatigue and severe sliding particles. As can be observed, all trained CNN model can adopt the discriminative surface regions to identify these two types of particles, except for the fatigue particle in Fig. 16 (h).

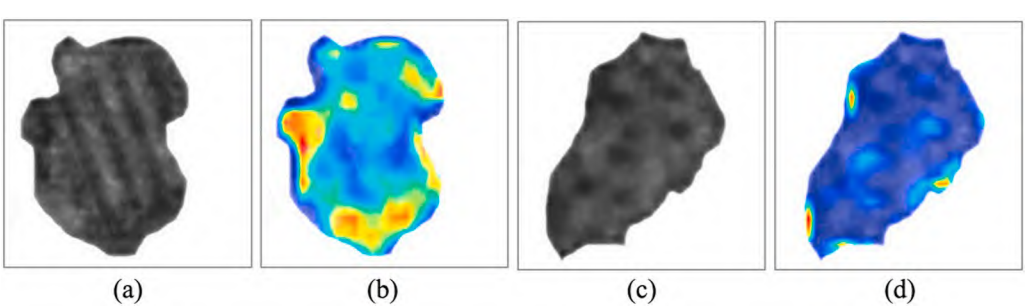

Fig. 12. Heat-maps of class activation: (a) severe sliding particle image, (b) the heat-maps of (a) relative to the sliding type, (c) fatigue particle image, (d) the heat-maps of (c) relative to the fatigue type.

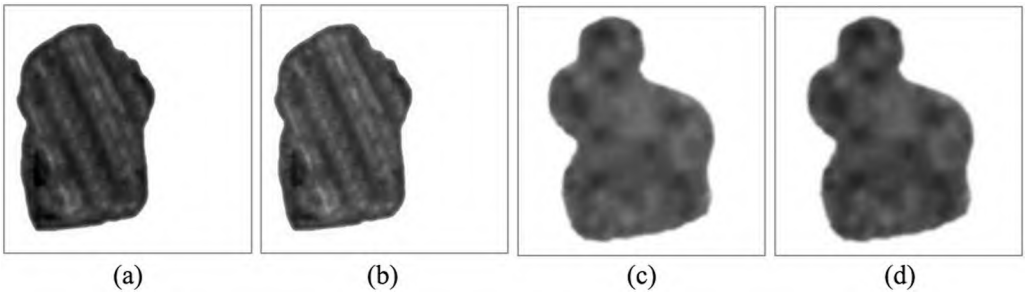

Fig. 13. Fault particle image standardization: (a) severe sliding particle image, (b) standardized image of (a), (c) fatigue particle image, (d) standardized image of (c).

Table 1
Convolution kernel number of each layer of the constructed CNN model.

Convolution layer	Model 1	Model 2	Model 3
Conv1	16	16	16
Conv2	32	16	16
Conv3	64	32	16
Conv4	64	32	16

There are deviations of the discriminative regions for the fatigue particle in Model 3, and they focus on the edge features rather than the pit area. The reason may be concluded that the convolution kernel of Model 3 is reduced too much, which results in a low learning rate and insufficient training. The results of heat-maps of class activation undoubtedly reveal that Models 1 and 2 can classify severe sliding and fatigue particles with their discriminative features.

Considering the algorithm structure and efficiency, Model 2 can be recognized as an effective particle identification model, and this model can be used for classifying severe sliding and fatigue particles.

3. Verification

Since Model 2 has been verified in detail using kernel visualization, it is only verified by recognizing typical particles. The test sample consists of two parts, i.e., ten groups of severe sliding and fatigue particles selected from the established particle database, and eight groups of new fault particles collected from the four-ball tester. With the 2D mapping method and image standardization method, these particles are identified with the constructed network (Model 2). The network can identify fatigue and severe sliding particles with a recognition accuracy of 100%. Parts of the recognition results are shown in Fig. 17. Combined with the visualization results of Model 2 and the recognition results, a highly effective particle identification model is established with optimizing the CNN model.

4. Discussions

A non-parametric automatic analysis method is constructed for identifying 3D surface of similar wear particles in a few real samples, involving the 2D mapping of 3D particle surfaces, wear particle sample generation, CNN-based identification, and improvements on CNN-based identification. The structure of the CNN-based identification model is optimized and verified with the results of network visualization. The constructed particle identification model is applied to identify severe sliding and fatigue particles and achieves high accuracy. It may be mentioned here that the model needs more iterations to be fully trained due to network simplification; otherwise, the identification accuracy may be limited. The comparison between the proposed method and other methods is shown below.

The combination of 2D images and parameter-based intelligent method (support vector machine and BP neural network) or non-parametric intelligent method (CNN) has promoted the automation of wear particle identification [5,8]. However, the 2D images of particles only provide the color information rather than the surface morphology, which can reflect the essential difference between these two particles. Compared with 3D particle analysis methods [12], the advantage of the proposed identification method is that the CNN model directly adopts the surface as the input, which effectively avoids the loss of surface information caused by the incompleteness of artificial-designed features. In addition, the 3D morphology of typical particles is difficult to be collected from actual machines. This will result in the over-fitting of the network in the training process. To address this issue, this paper introduces an improved CGAN to generate sufficient fault particle surfaces for the CNN-based particle identification model. Therefore, the developed system can be regarded as a practical particle identification approach for wear debris analysis.

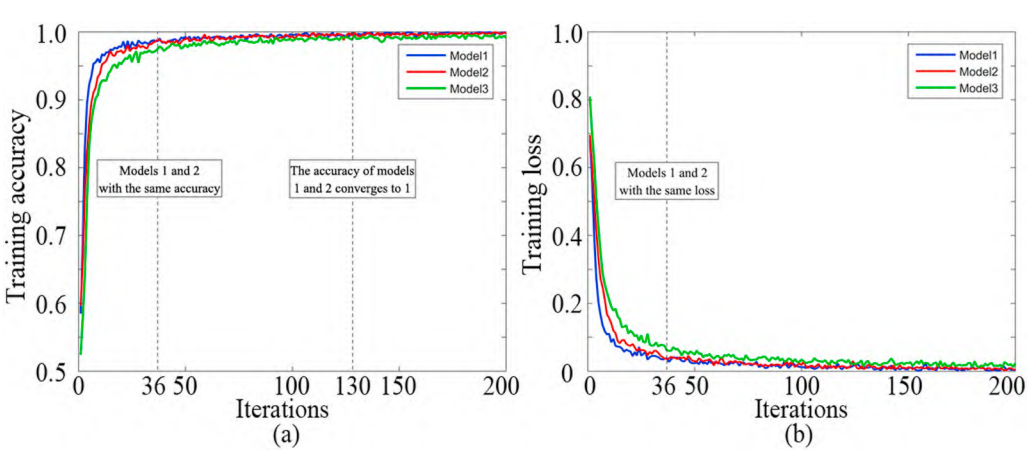

Fig. 14. The CNN training process: (a) identification accuracy, (b) identification loss.

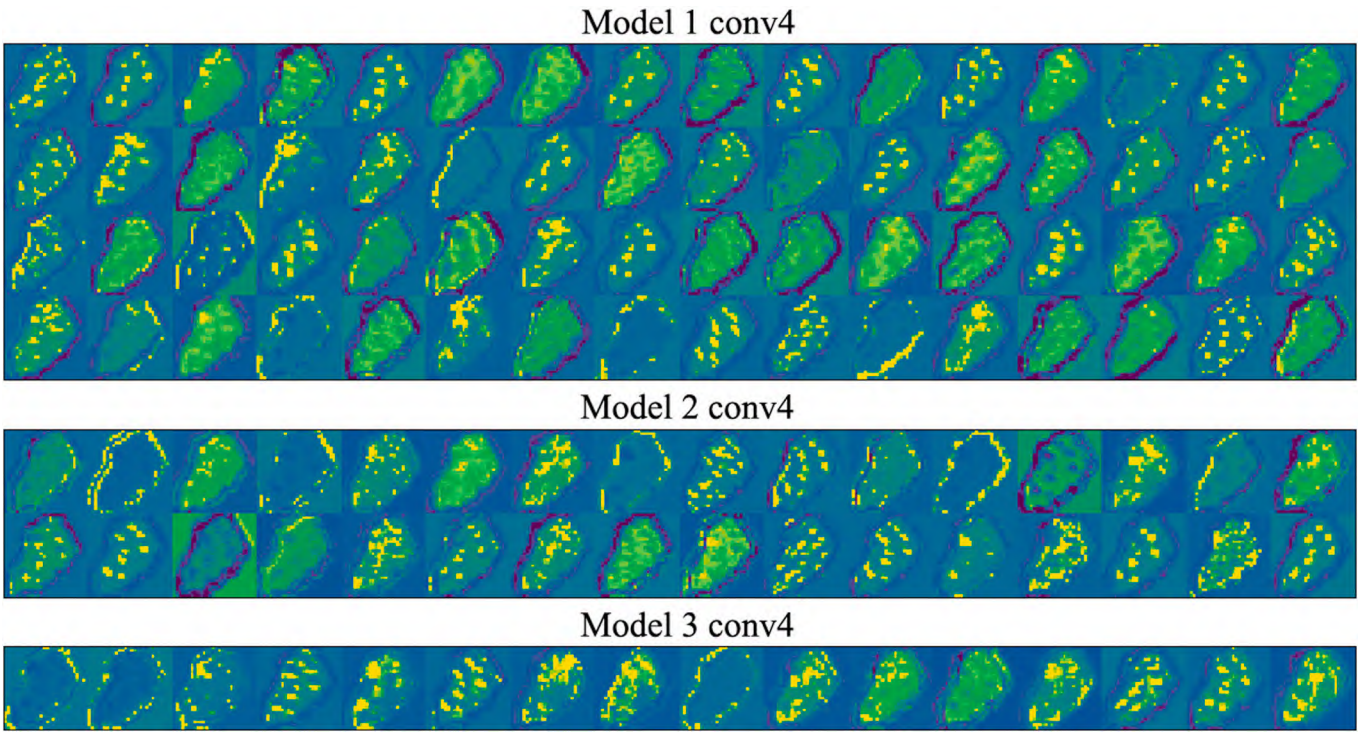

Fig. 15. The visualization of convolution kernel output with CNN models for the sixth particle image in Fig. 10.

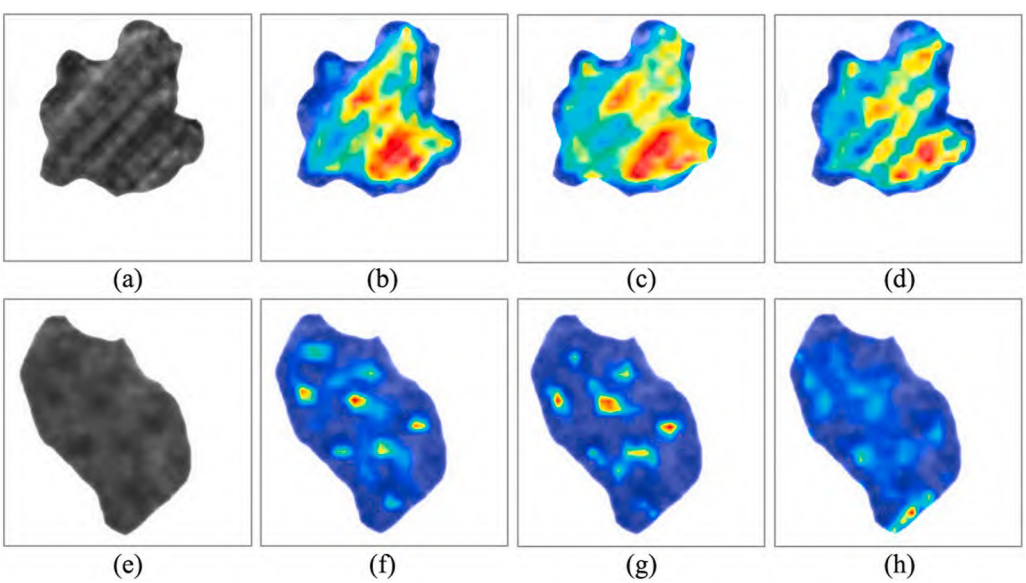

Fig. 16. The heat-maps of class activation for CNN model: (a) severe sliding particle, (b) the heat-maps of (a) with Model 1, (c) the heat-maps of (a) with Model 2, (d) the heat-maps of (a) with Model 3, (e) fatigue particle, (f) the heat-maps of (e) with Model 1, (g) the heat-maps of (e) with Model 2, (h) the heat-maps of (e) with Model 3.

Although the constructed similar particle identification model accurately identified all test particles, more particle samples are required to verify its reliability. This model will be combined with clustering algorithms [9] for typical wear particle recognition. The clustering algorithm is aimed at particles with distinct shape features, such as rubbing, cutting, and spherical particles. The other particles will be handled with the model described in this paper. With the application of this system, new wear particles will be collected and identified to expand the sample dataset, thus the efficiency of this system will be further enhanced.

5. Conclusions

A new similar particle 3D surface identification model is proposed and applied to the identification of fatigue and severe sliding debris. The main features are: 1) The 3D particle surfaces are converted to the form of 2D images without losing surface details. 2) A CGAN-based sample generation model is constructed to expand the 3D sample database of fault particles by optimizing the loss function and discriminant network. 3) By selecting CNN network parameters and optimization algorithm, a non-parametric identification model is constructed, and further optimized based on the results of the network visualization. The proposed

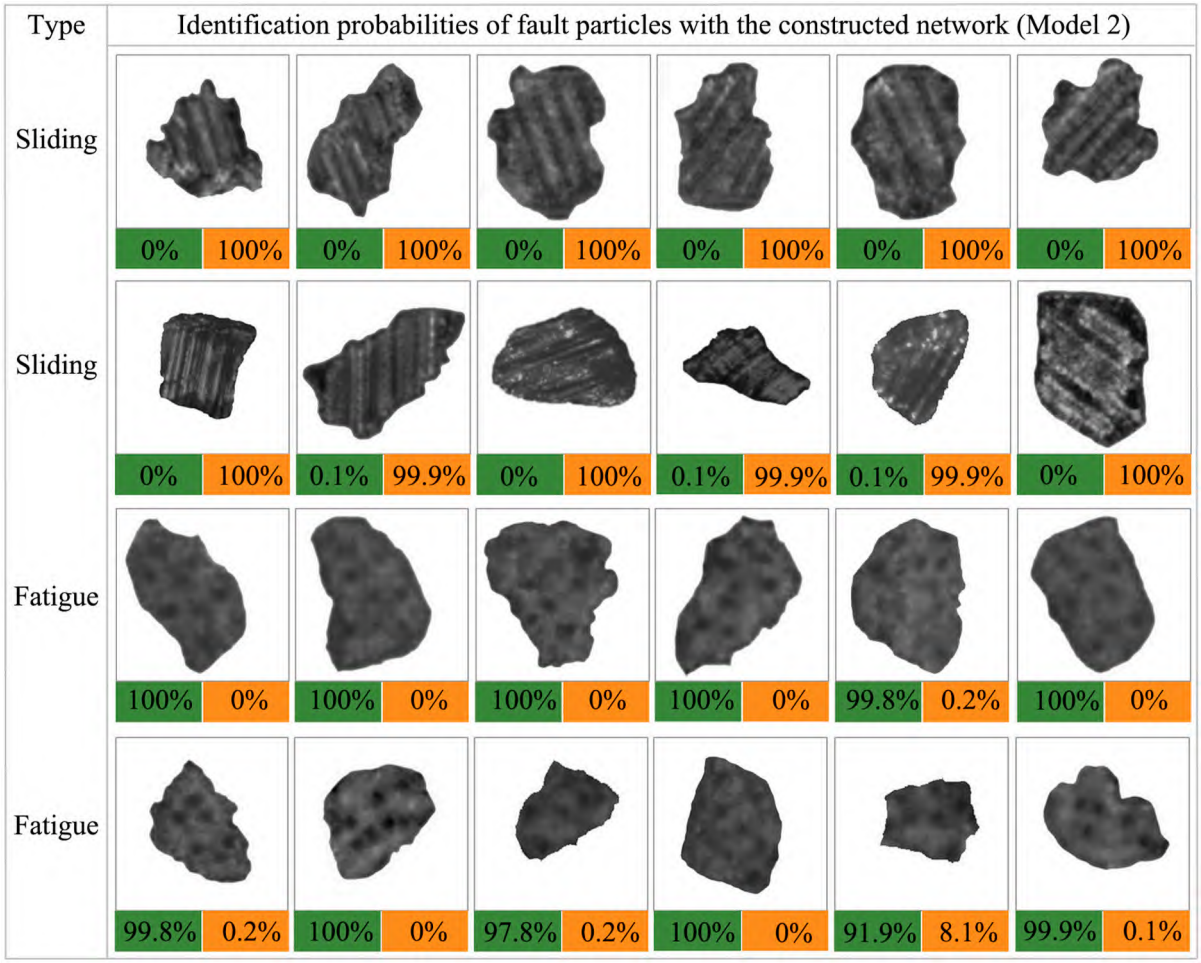

Fig. 17. Part of the identification results of fault wear particles (Note that the number in the green area indicates the probability that the particle belongs to the fatigue type, and the number in the yellow area for the severe sliding type). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

method can accurately identify all tested fault particles using key particle features. This investigation offers an insight into improving the efficiency of similar particle classification. This work will contribute toward enhancing the analysis of machine wear state.

CRediT authorship contribution statement

Shuo Wang: Methodology, Software, Validation,Writing- Original draft preparation. **Tonghai Wu:** Supervision. **Peng Zheng:** Investigation. **Ngaiming Kwok:** Writing- Reviewing and Editing.

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.

Acknowledgement

This work is supported by the National Natural Science Foundation of China (No. 51975455 and No. 51675403). The authors gratefully acknowledge the support of K.C. Wang Education Foundation.

References

[1] C. Haiden, T. Wopelka, M. Jech, F. Keplinger, M.J. Vellekoop, A microfluidic chip and dark-field imaging system for size measurement of metal wear particles in oil, IEEE Sensor. J. 16 (5) (2015) 1182–1189.
[2] S. Wang, T. Wu, L. Yang, N. Kwok, T. Sarkodie-Gyan, Three-dimensional reconstruction of wear particle surface based on photometric stereo, Measurement 133 (2019) 350–360.
[3] T. Wu, Y. Peng, H. Wu, X. Zhang, J. Wang, Full-life dynamic identification of wear state based on on-line wear debris image features, Mech. Syst. Signal Process. 42 (1–2) (2014) 404–414.
[4] Y. Du, S. Zhou, X. Jing, Y. Peng, H. Wu, N. Kwok, Damage Detection Techniques for Wind Turbine Blades: A Review, Mechanical Systems and Signal Processing, 2019, 106445.
[5] S. Wang, T. Wu, T. Shao, Z. Peng, Integrated model of bp neural network and cnn algorithm for automatic wear debris classification, Wear 426 (2019) 1761–1770.
[6] T. Wu, H. Wu, Y. Du, N. Kwok, Z. Peng, Imaged wear debris separation for on-line monitoring using gray level and integrated morphological features, Wear 316 (1–2) (2014) 19–29.
[7] W. Yuan, K. Chin, M. Hua, G. Dong, C. Wang, Shape classification of wear particles by image boundary analysis using machine learning algorithms, Mech. Syst. Signal Process. 72 (2016) 346–358.
[8] J. Wang, X. Wang, A wear particle identification method by combining principal component analysis and grey relational analysis, Wear 304 (1–2) (2013) 96–102.
[9] S. Wang, T. Wu, K. Wang, T. Sarkodiegyan, Ferrograph analysis with improved particle segmentation and classification methods, J. Comput. Inf. Sci. Eng. 20 (2).
[10] Y. Tian, J. Wang, Z. Peng, X. Jiang, A new approach to numerical characterisation of wear particle surfaces in three-dimensions for wear study, Wear 282 (2012) 59–68.
[11] G.P. Stachowiak, G.W. Stachowiak, P. Podsiadlo, Automated classification of wear particles based on their surface texture and shape features, Tribol. Int. 41 (1) (2008) 34–43.
[12] Z. Peng, T. Kirk, Wear particle classification in a fuzzy grey system, Wear 225 (1999) 1238–1247.

[13] P. Peng, J. Wang, Wear particle classification considering particle overlapping, Wear 422 (2019) 119–127.
[14] Y. Peng, J. Cai, T. Wu, G. Cao, N. Kwok, S. Zhou, Z. Peng, A hybrid convolutional neural network for intelligent wear particle classification, Tribol. Int. 138 (2019) 166–173.
[15] S. Moschoglou, S. Ploumpis, M. A. Nicolaou, A. Papaioannou, S. Zafeiriou, 3dfacegan: Adversarial Nets for 3d Face Representation, Generation, and Translation, arXiv: Computer Vision and Pattern Recognition.
[16] K. Nagasubramanian, S. Jones, A. K. Singh, A. Singh, B. Ganapathysubramanian, S. Sarkar, Explaining Hyperspectral Imaging Based Plant Disease Identification: 3d Cnn and Saliency Maps, arXiv: Computer Vision and Pattern Recognition.
[17] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, Least Squares Generative Adversarial Networks, arXiv: Computer Vision and Pattern Recognition.
[18] M. Arjovsky, S. Chintala, L. Bottou, Wasserstein Generative Adversarial Networks, Medical Image Computing and Computer Assisted Intervention, 2017, pp. 214–223.
[19] G. Antipov, M. Baccouche, J.-L. Dugelay, Face aging with conditional generative adversarial networks, in: IEEE International Conference on Image Processing, ICIP), 2017, pp. 2089–2093, 2017.
[20] P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks, Proceedings of the IEEE conference on computer vision and pattern recognition (2017) 1125–1134.
[21] O. Ronneberger, P. Fischer, T. Brox, U-net, Convolutional Networks for Biomedical Image Segmentation, Medical Image Computing and Computer Assisted Intervention, 2015, pp. 234–241.
[22] C. Liu, Z. Yang, F. Xu, J.-H. Yong, Image generation from bounding box-represented semantic labels, Comput. Graph. 81 (2019) 32–40.
[23] S. Ren, K. He, R. Girshick, J. Sun, Faster r-cnn: towards real-time object detection with region proposal networks, Neural information processing systems 2015 (2015) 91–99.
[24] S. Ioffe, C. Szegedy, Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, arXiv: Learning.
[25] B. Xu, N. Wang, T. Chen, M. Li, Empirical Evaluation of Rectified Activations in Convolutional Network., arXiv: (Learning).
[26] S. Wang, T. Wu, H. Wu, N.M. Kwok, Modeling wear state evolution using real-time wear debris features, Tribol. Trans. 60 (6) (2017) 1022–1032.
[27] Z. Peng, M. Wang, Three dimensional surface characterization of human cartilages at a micron and nanometre scale, Wear 301 (1–2) (2013) 210–217.
[28] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Neural information processing systems (2012) 1097–1105.
[29] G. Ciocca, P. Napoletano, R. Schettini, Cnn-based features for retrieval and classification of food images, Comput. Vis. Image Understand. 176 (2018) 70–77.
[30] P. Peng, J. Wang, Fecnn: a promising model for wear particle recognition, Wear 432–433 (2019) 202968, https://doi.org/10.1016/j.wear.2019.202968.
[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (1) (2014) 1929–1958.
[32] K. Wang, A. Kumar, Cross-spectral iris recognition using cnn and supervised discrete hashing, Pattern Recogn. 86 (2019) 85–98.
[33] G.B. Orr, K.-R. Müller, Neural Networks: Tricks of the Trade, Springer, 2003.
[34] J. Duchi, E. Hazan, Y. Singer, Adaptive subgradient methods for online learning and stochastic optimization, J. Mach. Learn. Res. 12 (2011) 2121–2159.
[35] M. D. Zeiler, Adadelta: an Adaptive Learning Rate Method, (arXiv: Learning).
[36] D. P. Kingma, J. Ba, Adam: A Method for Stochastic Optimization, (arXiv: Learning).
[37] B. Leng, K. Yu, J. Qin, Data augmentation for unbalanced face recognition training sets, Neurocomputing 235 (2017) 10–14.
[38] R.R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra, Grad-cam: visual explanations from deep networks via gradient-based localization, Proceedings of the IEEE International Conference on Computer Vision (2017) 618–626.