Advances in Reliability, Safety and Security, Part 9 – Kołowrocki, Dąbrowska (eds.)
 ISBN 978-83-68136-21-0 (printed), ISBN 978-83-68136-08-1 (electronic)

Advances in Reliability, Safety and Security

ESREL 2024
Monograph Book Series

Metal Surface Damage Recognition Method For Augmented Reality Assisted Maintenance Systems

Hongduo Wu, Dong Zhou, Ziyue Guo, Qidi Zhou

School of Reliability and System Engineering, Beihang University, Beijing, PR China
State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, PR China
State Key Defense Science and Technology Laboratory on Reliability and Environmental Engineering, Beihang University, Beijing, PR China

Abstract

The small damages such as cracks and scratches on the surface of aerospace products pose a serious threat to the safety of life and property, and manual visual inspection is prone to omissions, leaving great safety hazards. Using augmented reality (AR) assisted maintenance systems to assist visual inspection is one of the effective solutions. However, the limitations of computing power in augmented reality devices and the real-time requirements of augmented reality pose significant challenges to small-scale object detection algorithms. Therefore, this paper proposed a metal surface damage recognition method for augmented reality assisted maintenance system. Firstly, based on the appearance characteristics of surface damage in the steel image database NEU-CLS, the dataset was expanded and the images enhanced to expand the sample size and improve image quality. Afterwards, a SURF (speed up robust features) + K-means + Bag-of-Features (BOF) feature extraction and dimensionality reduction method was proposed to improve recognition efficiency while ensuring the robustness of the method. On this basis, the number of feature points in each image was introduced to reconstruct the feature vector and further improve the accuracy of defect recognition. Finally, the Support Vector Machine (SVM) was trained in Adaptive Boosting (AdaBoost) learning framework to construct a surface damage recognition model. After testing, this method has good accuracy and efficiency for common metal surface damages, and is suitable for augmented reality assisted maintenance system.

Keywords: augmented reality to assist maintenance, feature extraction, image recognition, ensemble learning

1. Introduction

For aerospace products, vehicles, and bridge facilities, metal surface damages caused by collision, vibration, impact, corrosion, and other effects are quite common. In the traditional visual inspection process, personnel fatigue and human negligence may occur, and some small surface damages (such as scratches and cracks) may cause product shutdown, production stoppage, and even endanger life and property safety. Therefore, there is an urgent need for real-time, efficient, and precise detection, localization, and maintenance of surface damage.

Augmented reality to assisted maintenance (ARTAM), it has become the main research and application direction of AR technology in the industrial field (Bottani et al., 2019). The augmented reality assisted maintenance system is the main form of achievement and application of ARTAM, relying on its advantages of virtual and real integration, lightweight wearing, and convenient and natural interaction to provide assistance and support for assembly, maintenance, and inspection operations. The system mainly relies on computer vision technology to recognize and locate maintenance targets.

At present, most surface damage detection methods based on computer vision are almost offline detection, and offline defect detection has lag. However, augmented reality requires the real-time overlay of virtual information in the field of view, so it is necessary to ensure a certain degree of real-time performance. However, improving the real-time performance of the method will inevitably affect accuracy, so it is necessary to

comprehensively consider the surface damage detection requirements and computational power limitations of the AR maintenance system, and balance the real-time and accuracy of recognition.

Currently, object detection technology can be divided into machine learning based methods and deep learning based methods based on network depth. Among them, deep learning based methods are currently the most popular, but due to the high complexity of deep learning models, such as Region CNN (Agrawal et al., 2014), Faster R-CNN (Ren et al., 2017), their computational efficiency simply cannot match the real-time requirements of augmented reality. However, more efficient one stage deep learning methods, such as the YOLO (Redmon et al., 2017) series of algorithms and the SSD (He et al., 2015) algorithm, cannot meet the visual inspection requirements in terms of balancing efficiency and accuracy either. In fact, the current object detection technology still needs to be improved in recognizing small surface damages in products. Various defect appearances, large intra-class variation, ambiguous inter-class distance, and unstable gray values will bring huge challenges. Machine learning based object recognition methods generally have the characteristics of high efficiency and easy optimization. (Luo et al., 2020) proposed a generalized completed local binary patterns (GCLBP) framework, simple nearest neighbor classifier (NNC) was utilized to accomplish the online detection of defects in hot-rolled strip steel. (Wang et al. 2019) used a sparse entity tracking strategy for defect detection based on the principle that surface image textures often form special structures. In order to solve the problem of insufficient features in defect images, they also proposed to perform intrinsic prior mining on defect images to obtain more information.

Therefore, this paper proposed a metal surface damage recognition framework based on machine learning, and studied image processing, feature extraction (Heinly et al., 2012), classifier construction, and other aspects in the framework.

2. Method

The overall method framework is illustrated in Figure 1. Initially, we utilized six common surface damage images from the Northeastern University steel surface defect image database (NEU-CLS) as the database. The dataset was expanded based on the distinctive appearance features of various damages, addressing the issue of limited data samples.

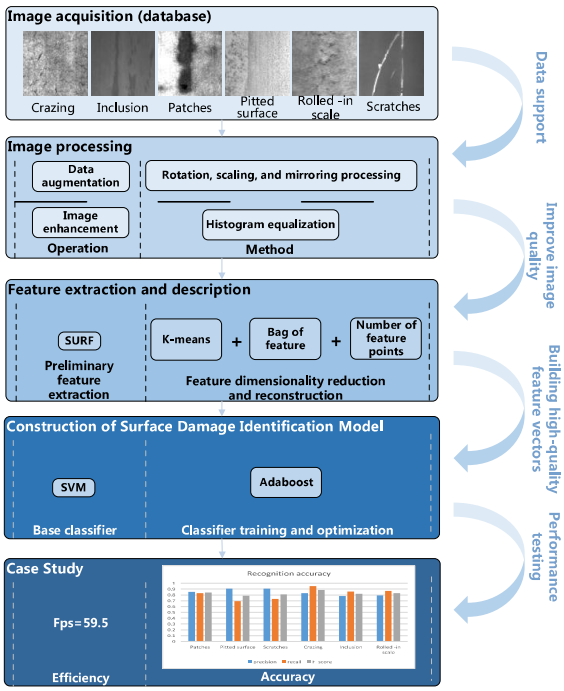

Fig. 1 Overall method framework.

Subsequently, histogram equalization was employed to enhance the visual features of surface damage, aiming to improve the image quality. Afterwards, feature detection and description were performed. Building upon the initial feature extraction using SURF (Bay et al., 2008), we employed the "K-means + Bag-of-Features" method (Cao et al., 2013) for data dimensionality reduction. This method enables the transformation of high-dimensional feature vectors into low-dimensional ones with specified dimensions, enhancing recognition efficiency while maintaining the robustness of the approach. On this basis of feature vectors, the non-dimensional information of feature point quantity for each image was introduced. This information was utilized for the reconstruction of feature vectors, further enhancing defect recognition accuracy. Finally, a SVM (Lin et al., 2011) was employed for training within the adaptive boosting (Ratsch et al., 2001) learning framework. This facilitated the construction of a surface damage recognition model, followed by case validation.

2.1. Data and image processing

In various industrial sectors such as aerospace, shipbuilding, automotive, electromechanical, and chemical industries, hot-rolled strip steel is predominantly used as the primary raw material for manufacturing various metal-clad products. Therefore, this paper selected the steel surface defect image database (NEU-CLS) from Northeastern University in the United States as the dataset for research. This database encompasses six common types of surface damage observed in the manufacturing, transportation, and usage processes of hot-rolled strip steel: cracking, inclusions, patches, surface pitting, rolled-in scale, and scratches. Each type of damage comprises 300 images, as illustrated in Figure 2.

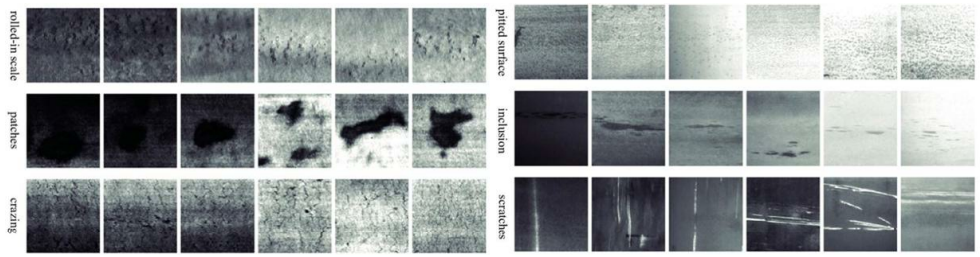

Fig. 2. 6 types of surface damage in NEU-CLS.

To address the issue of potential sample scarcity during model training, data augmentation was applied to the dataset. This serves the dual purpose of increasing the dataset size to prevent overfitting and enriching the variety of sample forms in the training set to enhance the robustness of the method. In this approach, each image underwent rotation (random clockwise rotations of 90°, 180°, or 270°), mirroring, and scaling (downsizing to half the original size, or upsizing to twice the original size). This augmentation process expanded the original dataset by a factor of five.

The images in the database are relatively dark. To enhance the visibility of surface damage in the images, histogram equalization was employed for image enhancement. This method involves stretching and adjusting the contrast of the images, ensuring a uniform distribution of grayscale across the entire grayscale range. The effect of histogram equalization is illustrated in Figure 3.

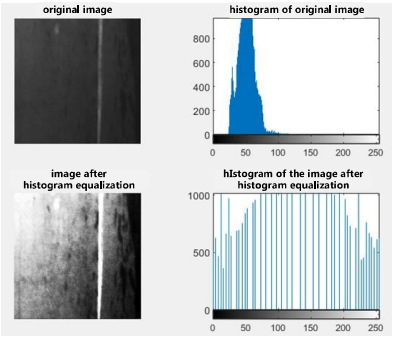

Fig. 3. The effect of histogram equalization and corresponding grayscale histograms.

2.2. Feature detection and description

After image processing, it is necessary to perform feature point extraction and description. Among mainstream feature extraction algorithms such as SIFT (Scale Invariant Feature Transform) (Wu et al., 2021) and FAST (Features from Accelerated Segment Test) (Rosten et al., 2010), the feature vectors extracted by the SURF (Speeded-Up Robust Features) algorithm possess both scale and rotation invariance. Additionally, they exhibit extremely high efficiency in feature extraction, meeting the requirements for real-time applications.

(1) *The initial extraction of features*

This paper employs the SURF algorithm for the initial extraction of features from the images enhanced through image processing. In the feature detection stage, SURF first filters the image and subsequently constructs the Hessian matrix of the image. The purpose is to utilize the discriminant of the Hessian matrix to detect more robust feature points. The Hessian matrix of an image is a square matrix composed of the second-order partial derivatives of pixel points and their neighbors. SURF calculates the second-order derivatives through convolution with a specific kernel to obtain the Hessian matrix, denoted as H, as shown in (1). Each pixel point can thus yield a Hessian matrix.

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{1}$$

The SURF algorithm constructs a scale space by building an image pyramid for each image, ensuring that the detected feature points exhibit scale invariance. The process of constructing the scale image pyramid is as follows: Firstly, the scale space is divided into multiple octaves, and various-sized box filters are applied within each octave. Through filtering the image with these filters, several images with different scales are obtained.

Subsequently, within the scale space, feature points are determined using the feature point discriminant (also known as feature point responsiveness). The discriminant for feature points is expressed in (2). A pixel is considered a feature point if its discriminant achieves a local maximum compared to its neighboring points. Following this, by setting a threshold for feature point responsiveness, weaker and mislocated feature points are filtered out, resulting in a set of stable feature points for each image.

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 \tag{2}$$

In the feature description stage of the SURF algorithm, the first step is to assign a primary orientation to each feature point, aiming to impart rotational invariance to the feature vectors. Subsequently, for each feature point, a square region with a side length of 20 pixels is selected based on the primary orientation. This region is further divided into a 4×4 grid, and within each grid, the total sum of Haar wavelet responses in the horizontal and vertical directions is computed for the pixels included. The sum of Haar wavelet responses includes the sum of Haar wavelet values in the horizontal direction, vertical direction, absolute values in the horizontal direction, and absolute values in the vertical direction, as illustrated in Figure 4. These four values are extracted as part of the feature vector for each feature point, resulting in a 64-dimensional feature vector (4×4×4=64 dimensions) for each feature point. SURF features are then extracted from the images in the processed database, forming a collection of feature vectors.

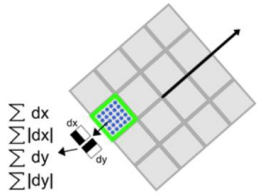

Fig. 4. The feature vector description process of SURF.

(2) *Feature dimensionality reduction and reconstruction*

Although the SURF algorithm excels in real-time performance, each feature point still corresponds to a 64-dimensional feature vector. Considering that each image typically contains hundreds or even thousands of feature points, the data volume is significant, posing computational challenges for subsequent processing. Directly inputting these high-dimensional feature vectors into a classifier for defect recognition often fails to meet real-time requirements. Therefore, dimensionality reduction and reconstruction are applied to the SURF features.

This paper initially employs a dimensionality reduction method that allows for specifying the data dimensions. It utilizes the "K-means + Bag-of-Features" approach to reduce the dimensionality of feature vectors. Subsequently, non-dimensional information regarding the quantity of feature points for each image is introduced into the dimensionality-reduced feature vectors. This process results in the re-description of the feature vectors. The flow of feature dimensionality reduction and reconstruction is illustrated in Figure 5.

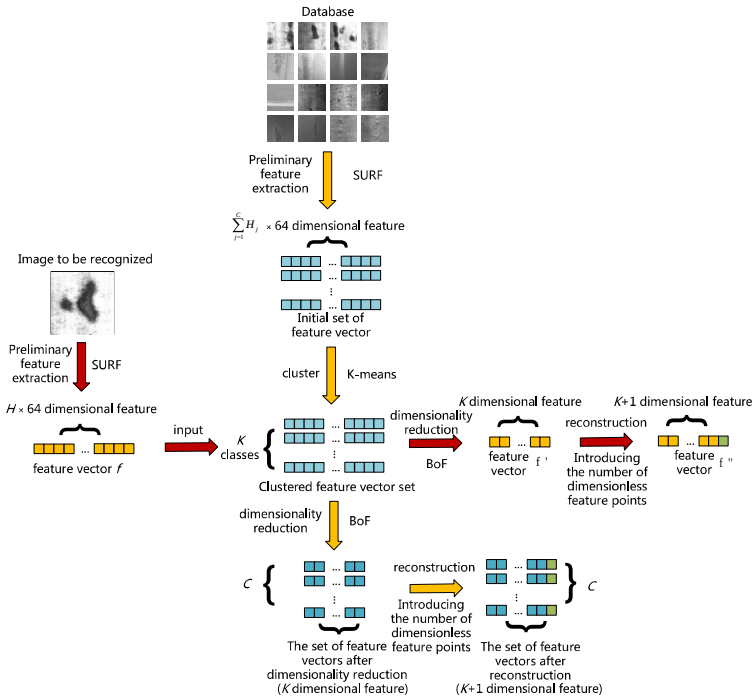

Fig. 5. Process of feature dimensionality reduction and reconstruction. (*K* is the number of clusters, *C* is the number of images in the database, *H* represents the number of feature points in the image to be recognized, and H_j represents the number of feature points in the *j*-th image).

Firstly, the SURF feature vectors extracted from the database images are subjected to clustering using the K-means method. This process yields *K* cluster centroids, which serve as the feature dictionary. The number of clusters *K* is determined through multiple pre-training iterations. The average F-score, a recognition accuracy parameter, is computed for each of the six types of metal surface damage. The relationship between the average F-score and K is analyzed, as depicted in Figure 6. It is observed that the average F-score reaches its maximum when *K* equals 7. Therefore, *K* is set to 7.

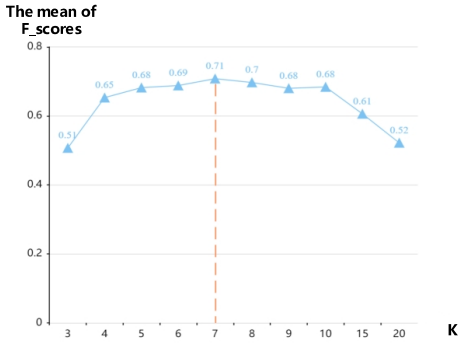

Fig. 6. The relationship between the number of clustering categories *K* and the accuracy of defect recognition.

For each image, the Bag-of-Features is employed to statistically analyze the distribution of all feature points in that image among the K clusters in the feature vector set. This involves calculating the proportion of feature points in each cluster relative to the total number of feature points in the entire image. The calculated proportions for the K clusters are arranged in order to construct a new feature vector $f'=(f_1', f_2', \dots, f_k')$ for the image, thereby obtaining the dimensionality-reduced feature vector set. The calculation for each element in the feature vector is represented by the following equation, where N_j is the number of feature points in the j-th image, and N_{ji} is the number of feature points in the j-th image that belong to the i-th cluster.

$$f_i' = \frac{N_{ji}}{N_j}, i = 1,2, \dots, K; j = 1,2, \dots, M. \tag{3}$$

Since the quantity of feature points for each type of defect contributes positively to improving defect recognition accuracy, and the feature vector f' does not utilize this information, for each dimensionality-reduced feature vector f' in the set, the quantity of feature points for the corresponding image is introduced into f'. This is achieved by appending the non-dimensionalized feature point quantity for that image after the K-th dimension, resulting in a reconstructed feature vector $f''=(f_1'', f_2'', \dots, f_{k+1}'')$ with K+1 dimensions, where $f_1'' = f_1', i = 1,2, \dots, K$.

The purpose of non-dimensionalization is to eliminate the dimensional discrepancy between the (K + 1)-th dimension and the preceding K dimensions in the feature vector. Otherwise, the last dimension of the feature vector might exert a decisive influence. The non-dimensionalization method for the (K + 1)-th dimension of the feature vector is represented by the following equation.

$$f_{k+1}'' = \frac{N_j - N_{min}}{N_{max} - N_{min}} \tag{4}$$

In the equation, f_{k+1}'' represents the K+1th dimension of the reconstructed feature vector, N_j is the number of feature points in the j-th image, N_{max} is the maximum number of feature points among all images in the image database, and N_{min} is the minimum number of feature points among all images in the image database. For the dimensionality-reduced feature vector of the image to be recognized, the value of the (K + 1)-th dimension is computed according to the above equation and is appended to the original feature vector to reconstruct the feature vector f'.

After the aforementioned feature dimensionality reduction and reconstruction, the feature vector for each image, originally of size N_j×64, has been reduced to K+1 dimensions.

(3) *Test for the effectiveness and superiority of reconstructed features*

To validate the effectiveness and superiority of the feature vector reconstruction method, the reconstructed feature vectors and the original feature vectors described by the SURF algorithm are separately input into the base classifier (Support Vector Machine) for a comparative analysis of recognition performance and efficiency. The images representing the six types of metal surface defects in NEU-CLS, with 300 images for each defect type, are evenly divided into five folds for a five-fold cross-validation.

During the reconstruction process, the K-means clustering employs the Euclidean distance metric. Prior to the commencement of clustering, K centroids (where K = 7) are uniformly and randomly generated based on the distribution range of the dataset. In the SVM training process, the kernel function utilized is the Gaussian kernel. The training is conducted using the one-versus-one (OVO) strategy. Precision, recall, and F-score are compared separately for SURF features and reconstructed features, as illustrated in Figure 7.

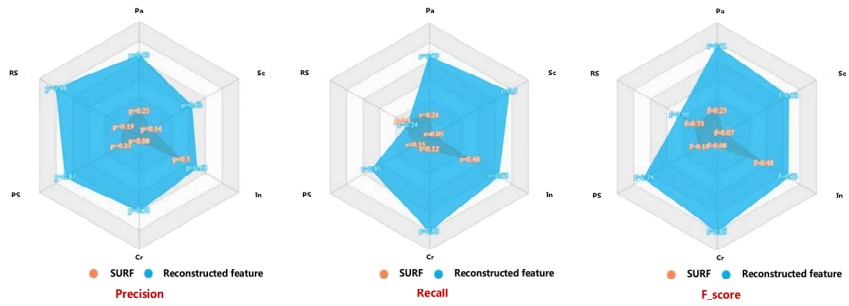

Fig. 7. Comparison of recognition accuracy between SURF and reconstructed features.

It is evident that the advantages of reconstructed features are prominent. The precision, recall, and F-score for the six types of surface damage using SURF features are remarkably low. This may be attributed to the

excessively high dimensionality of SURF feature vectors, resulting in the curse of dimensionality and leading to overfitting. Reconstructed features demonstrate outstanding precision in recognizing all six types of surface damage, with a commendable recall for all surface damages except pitted surface.

Pitted surface, characterized by small individual pits, is challenging to identify, as these pits often have a light color and may be difficult to discern even with the naked eye. This difficulty in recognition could be one of the reasons for the comparatively lower recall rate. Consequently, the F-score for pitted surface is also relatively low, at 0.39. To enhance recognition accuracy, optimization of the model during the classifier training process is recommended.

2.3. Construction of surface damage identification model

In response to the challenge of low recall rates in the recognition of pitted surface using the combination of reconstructed features and SVM, this study addresses the issue through the utilization of the AdaBoost learning strategy within the framework of ensemble learning. The SVM is employed as the base classifier for training the surface damage identification model.

The AdaBoost algorithm is a learning strategy that, through iterative learning, prediction, and sample updates, generates a series of weak classifiers with different parameters. In this process, altering the weight distribution of training data is based on the principle of "focusing on misclassified samples." Specifically, for samples that are misjudged, their weights are increased when constructing the next training set, while for samples that are correctly classified, their weights are reduced. The reweighted dataset is then used to train the next weak classifier, allowing the classification problem to be addressed through a sequence of weak classifiers employing a "divide and conquer" approach. Subsequently, based on the misclassification rates of each classifier, a strong classifier specific to the problem is generated through weighting.

The combination of weak classifiers adopts a weighted majority voting approach, where the weights are determined by their misclassification rates. Weak classifiers with lower misclassification rates are assigned larger weights, allowing them to have a more significant impact in the voting process. Conversely, weak classifiers with higher misclassification rates are assigned smaller weights, reducing their influence. This ultimately aims to achieve a reduction in the overall misclassification rate by giving more weight to the more accurate classifiers and diminishing the impact of less accurate ones.

The enhancement steps for the SVM base classifier based on the AdaBoost algorithm are as follows:

- *Input*: Training dataset *S*, number of training rounds *M*, where $S = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$ and x_i represents the feature vector describing the *i*-th sample, while y_i denotes the category identifier for the *i*-th object;
- *Step1*: Initialization of Weight Data $D_1 = (w_{11}, w_{12}, \dots, w_{1i}, \dots, w_{1N})$, $w_{1i} = \frac{1}{N}$, $i = 1,2, \dots, N$;
- *Step2*: Initialize m, m=1;
- *Step3*: Training SVM with training data containing feature weights denoted by D_m to obtain the corresponding weak classifier;

  $$D_m(x): X \to Y \tag{5}$$

- *Step4*: If $\sum_{i=1}^{N} I(G_m(x_i) \neq y_i)\, D_1 = 0$ (it means no misclassification points), or *M*=*m*, proceed to Step 9, otherwise, enter Step 5;
- *Step5*: Calculate the classification error rate of $G_m(x)$;

  $$e_m = P(G_m(x_i) \neq y_i) = \sum_{i=1}^{N} w_{mi} I(G_m(x_i) \neq y_i) \tag{6}$$

- *Step6*: Compute the combined weight α_m corresponding to the weak classifier;

  $$\alpha_m = \frac{1}{2} log \frac{1-e_m}{e_m} \tag{7}$$

- *Step7*: Update sample weights D_m;

  $$D_{m+1} = (w_{m+1,1}, w_{m+1,2}, \dots, w_{m+1,i}, \dots, w_{m+1,N}) \tag{8}$$

  where $w_{m+1,i}$ represents the weight of the *i*-th sample in the next round of training.

  $$w_{m+1,i} = \frac{w_{m,i}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)), \tag{9}$$

  Z_m is the normalization factor, calculated according to the following formula:

  $$Z_m = \sum_{i=1}^{N} w_{mi} \exp(-\alpha_m y_i G_m(x_i)) \tag{10}$$

- *Step8*: $m = m+1$， return to Step 4;
- *Step9*: Combine the weak classifiers linearly to obtain the strong classifier;

$$f(x) = \sum_{m=1}^{M} \alpha_m G_m(x) \tag{11}$$

if $\sum_{i=1}^{N} I\left(G_m\left(x_i\right) \neq y_i\right)=0$, then $\alpha_m = 1$, and obtain the final surface damage recognition model;

$$G(x) = sign(f(x)) = sign(\sum_{m=1}^{M} \alpha_m G_m(x)) \tag{12}$$

- *End*: Output the surface damage recognition model $G(x)$.

3. Case study

This paper employs the NEU-CLS database as a case study dataset for validation purposes. The proposed framework, including processes such as image processing, feature extraction, and damage recognition model, is validated using this dataset. The dataset consists of 300 images for each of the six types of metal surface defects in NEU-CLS, and they are evenly divided into five folds for a five-fold cross-validation.

3.1. Image processing, feature extraction and model training

For the 240 images in the training set, perform the rotation, mirroring, and scaling operations as described in Section 2.1. Apply histogram equalization to both the training and testing sets of images, and the results are depicted in Figure 8.

During the feature extraction and reconstruction process, K-means is employed with Euclidean distance, where *K* is set to 7. Utilizing the AdaBoost framework, train the SVM to generate the surface damage recognition model. In the training process, the kernel function used is the Gaussian kernel.

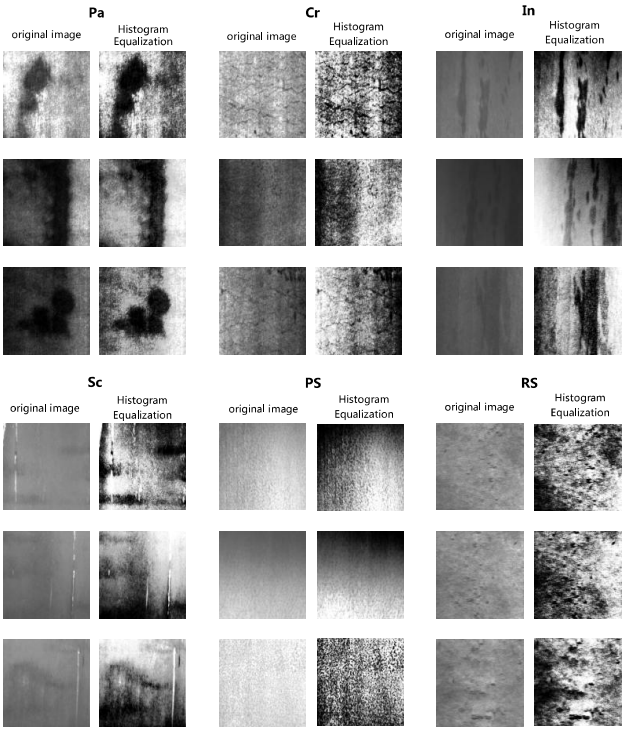

Fig. 8. Surface damage image after histogram equalization.

3.2. Results and discussion

To intuitively verify the effectiveness of the AdaBoost training strategy in this framework, a comparison is made between the SVM recognition model trained through AdaBoost and the SVM model trained directly. The results are illustrated in Figure 9.

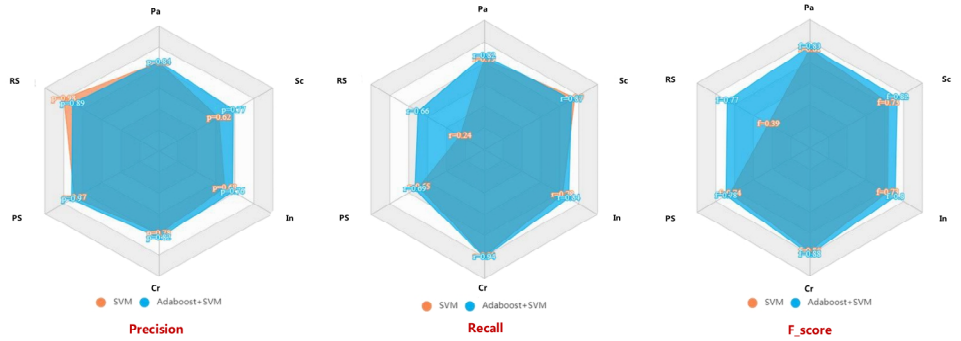

Fig. 9. The comparison of the recognition accuracy between AdaBoost+SVM and SVM.

It can be observed that the surface damage recognition model trained through the AdaBoost algorithm shows a 15% improvement in precision for rolled-in scale and an 8% improvement for inclusion compared to the model trained directly. There is also a slight improvement in precision for other types of damage. Previously, the recall rate of the basic SVM model for pitted surface was only 24%. After enhancement through the AdaBoost algorithm, the recall increased to 0.66, representing a 42% improvement and effectively addressing the low recall issue for pitted surface in the basic classification model. Correspondingly, the comprehensive evaluation parameter F-score also demonstrates an overall improvement, with a significant increase in the F-score for pitted surface, reaching 0.77.

The precision histogram of the recognition accuracy is depicted in Figure 10. The inclusion has the lowest precision but still reach 0.78. The recall rates for pitted surface and scratches are slightly lower, at 0.69 and 0.73, respectively. However, compared to the basic classification model's recall rates of 0.24 and 65% for these defects, there is a significant improvement, and these recall rates are considered acceptable. The recall rates for other types of surface defects are all above 0.80. The F-scores are 0.84 (patches), 0.78 (pitted surface), 0.81 (scratches), 0.89 (crack), 0.82 (inclusion), and 0.83 (rolled-in scale).

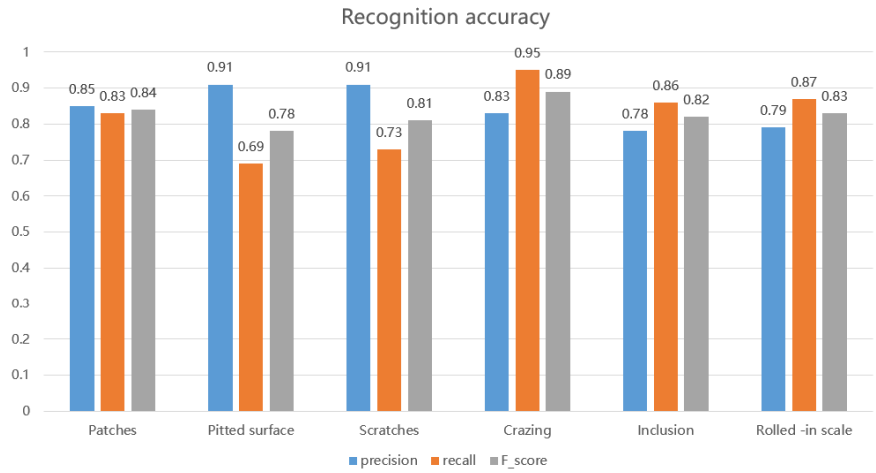

Fig. 10. Recognition accuracy histograms for the 6 types of surface damage.

In terms of recognition efficiency, the average time taken by the model to process the five rounds of test sets (each round consisting of 360 images) was calculated during five-fold cross-validation. In the operating environment of Windows 10, Intel(R) Xeon(R) CPU E3-1220 v3 (the overall performance is slightly higher than that of the Qualcomm Snapdragon 850), the average time per round was 6.0480 seconds, the average recognition time per image was 0.0168 seconds, achieving a frame rate of 59.5 and demonstrating real-time capabilities.

4. Summary

In addressing the challenge of lacking an algorithm that balances accuracy and real-time performance for surface damage recognition in augmented reality assisted maintenance system, this paper proposes a comprehensive method framework tailored for typical metal surface damage. The framework includes methods for surface damage data augmentation and image enhancement specifically designed for augmented maintenance systems, as well as techniques for feature extraction, reconstruction, and target recognition. Through testing, it has been demonstrated that this framework can achieve high surface damage recognition accuracy while ensuring real-time performance, addressing the specific needs of augmented reality assisted maintenance system.

Acknowledgements

I would like to give my sincere gratitude to Mr. Zicheng Song who collected a lot of literature about the research. I also pleased to acknowledge Mr. Songliang Shuai for searching for suitable databases.

References

Agrawal, P., Girshick, R., Malik, J. 2014. Analyzing the performance of multilayer neural networks for object recognition, European Conference on Computer Vision. Switzerland: ECCV, 329-344.

Bay, H., Ess, A, Tuytelaars, Y. et al. 2008. Speeded-up robust features (SURF). Computer Vision and Image Understanding 110(3), 346-359.

Bottani E., Vignali G. 2019. Augmented reality technology in the manufacturing industry: a review of the last decade. IISE Transactions 51(3), 284-310.

Cao, J., Wu, Z., Wu, J. et al. 2013. Towards information-theoretic K-means clustering for image indexing.Signal Processing 93(7), 2026–2037.

He, K., Zhang, X., Ren, S. et al. 2015. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis & Machine Intelligence 37(9), 1904-1916.

Heinly, J., Dunn, E., Frahm, J.M. 2012. Comparative evaluation of binary features, Proceedings of European Conference on Computer Vision. Berlin, Germany: Springer, 759 -773.

Lin, Y., Lv, F., Zhu, S., et al. 2011. Large-scale image classification: Fast feature extraction and SVM training, CVPR 2011.

Luo, Q., Sun, Y., Li, P., Simpson, O., Tian, L., He, Y. 2019. Generalized completed local binary patterns for time-efficient steel surface defect classification. IEEE Transactions on Instrumentation & Measurement 68(3), 667-679.

Ratsch, G. 2001. Soft Margins for AdaBoost. Machine Learning 42(3), 287-320.

Redmon, J., Divvala, S., Girshick, R. et al. 2016. You Only Look Once: unified, real-time object detection, IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: CVPR, 779-788.

Ren, S., He, K., Girshick, R. et al. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis & Machine Intelligence 39(6), 1137-1149.

Rosten, E., Porter, R., Drummond, T. 2010. Fast and better: a machine learning approach to corner detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 32(1), 105-119.

Wang, J., Li, Q., Gan, J., Yu, H., Yang, X. 2020. Surface defect detection via entity sparsity pursuit with intrinsic priors. IEEE Transactions on Industrial Informatics 16(1), 141-150.

Wu, H., Zhou, J. 2021. Privacy Leakage of SIFT Features via Deep Generative Model Based Image Reconstruction.Institute of Electrical and Electronics Engineers (IEEE).