Steel Surface Defect Detection and Classification Using Bag of Visual Words with BRISK

Conference Paper · July 2023

Alaa Aldein M. S. Ibrahim
Univrsity of Al Fashir

Steel Surface Defect Detection and Classification Using Bag of Visual Words with BRISK

Chapter · July 2023

DOI: 10.1007/978-981-99-2468-4_18

Alaa Aldein Mohamed Suliman Ibrahim
University of KwaZulu-Natal

Chapter 18
Steel Surface Defect Detection and Classification Using Bag of Visual Words with BRISK

Alaa Aldein M. S. Ibrahim and Jules-Raymond Tapamo

Abstract Nowadays, in many companies, defect classification plays a vital role in surface quality measuring instruments. However, there is a conflict between accuracy, efficiency, and high computational complexity for traditional defect classification methods. This paper focuses on the accuracy and efficiency of classification for steel surface defects. We propose a bag of visual words technique with low computational complexity using BRISK detector and Support Vector Machines. Experiments conducted show that the proposed method outperforms many state-of-the-art approaches.

18.1 Introduction

Surface defect classification assigns a surface defect to one category automatically or manually and plays a significant role in steel strip quality control. In steel companies, manual inspection methods are unreliable in mass production and high-speed automated systems. It consumes a lot of time and effort, besides their high cost, and is highly dependent on inspector expertise and accuracy. To reduce the cost, increase profitability, improve the quality of products at a lower price, and remain competitive, steel companies have significantly increased their effort to use machine vision to automatize defect detection processes. Many works in the classification of steel surfaces recently proposed have varying degrees of success. Also, surveys on several methods for steel product inspection have been introduced by Mordia et al. [1], Jin et al. [2], Zheng et al. [3], Czimmermann et al. [4], Luo et al. [5, 6], are worthy articles for scholars in the field. This paper proposes a method that uses Support Vector Machines and a bag of visual words (BoVW) to perform steel surface defects classification. The use of BRISK for the extraction and detection of surface descriptors and a BoVW-based model makes this method innovative and efficient.

A. A. M. S. Ibrahim (✉) · J.-R. Tapamo
School of Engineering, University of Kwazulu Natal Durban, Durban 4041, South Africa
e-mail: Alaaaldein@fashir.edu.sd

J.-R. Tapamo
e-mail: Tapamoj@ukzn.ac.za

J. C. Bansal et al. (eds.), *Congress on Smart Computing Technologies*, Smart Innovation, Systems and Technologies 351, https://doi.org/10.1007/978-981-99-2468-4_18

The rest of the paper is organized as follows. Section 18.2 reviews related work. Section 18.3 presents the technical background and method. Experiments and discussions on results are out lined in Sect. 18.4. Section 18.5 concludes there and outlines future work.

18.2 Related Work

Recent developments in image recognition methods have shown that the BoVW feature method has a fabulous performance for many image recognition and classification problems. In literature, a lot of approaches have been developed using BoVW. For example, a method based on BoVW was proposed to distinguish between car and non-car images. Zhi-jie et al. [7] improved a method based on a bag of visual words approach to classifying images. In this work, the visual saliency of the image has been computed, and the histogram of visual words of the image ware computed according to the image visual saliency. Then, the histogram of visual words was used to represent the image. Huzaifa et al. [8] proposed a system based on the manufacture of the intelligent transportation system to identify car models from their frontal image using the BRISK method. Hao et al. [9] improved a traditional bag of words by combining Hamming embedding technology and contrast words. In this work, a highly discriminative word bag model has been applied and combined into the SVM. Lima et al. [10] proposed a method named BoVW-CN that combines bag of visual words and complex networks to describe keypoints detected in a given image. Liu et al. [11] proposed an intelligent system to detect fake coins automatically by using the enhanced bag of visual words model, called the SEBOVW model. Nabizadeh et al. [12] proposed a method consisting of a bag of visual words and a neural network classifier to classify X-ray chest images into COVID-19 and non-COVID-19. Suvdaa et al. [13] utilized the SIFT descriptor to perform defectuous regions detection and extract features of steel surfaces; SVM was then used for defects classification. The high descriptive power, viewpoint changes, and robustness to illumination have made the SIFT descriptor to be rated best in the survey in [14]. However, the high dimensionality makes SIFT very slow. To overcome this problem, they reduced the feature's dimensional from 128 to 20, but this process increased the computational time for descriptor formation. Zaghdoudi et al. [15] presented a classification system of steel defects based on machine vision technology that has two-step: feature extraction and defects classification. They used two sets of features (Histogram of Gradients (HOG) and Gray Level Co-occurrence (GLCM)) for feature extraction and SVM as a classifier. However, combining HOG and GLCM features incurs a higher computational cost. Boudiaf et al. [16] developed an inspection system for surface defects detection of hot-rolled steel. They used a histogram of oriented gradients (HOG) for feature extraction, principal component analysis (PCA) to reduce the dimensionality of the feature, and a K-nearest neighbor classifier (KNN) to classify defects. However, they obtained unsatisfactory accuracy. Due to a growing demand for high-quality, high-speed features, and low computational cost, we have proposed an efficient method for the classification of steel surface defects to control the quality of steel strips.

Many Feature detectors, that differ in the type of keypoints detected, time complexity, repeatability, and space complexity, have been proposed in the literature. Some examples, are Binary Robust Invariant Scalable Keypoints (BRISK) [17], SIFT, Features from Accelerated Segment Test (FAST) [18], Speeded Up Robust Features (SURF) [19], Binary Robust Independent Elementary Features (BRIEF) [20], ORB [21]. Two thorough reviews [22, 23] on keypoint detectors have been produced, and they discuss worthy articles that are very useful for researchers interested in this area. Also, there are a set of feature extractors like BRIEF, BRISK, ORB, and FREAK [24] that can be used for object detection purposes. BRISK has a comprehensive performance, with less computational complexity and requires less storage space [25]. Nowadays, most of the existing works that are based on the bag of visual words technique use the common SURF and SIFT algorithms to describe detected keypoints in images. However, the problem with these algorithms is that they have high computational complexity due to the construction of high-dimensional feature vectors. To overcome this problem, we have proposed a feature extraction model based on bag of visual words (BoVW) and BRISK extractor, that takes advantage of the low computational complexity and less storage space of the BRISK extractor to help speed-up keypoints detection process, for steel surface defects detection.

18.3 Technical Background and Methodology

The proposed method is based on three steps. First, extracting local features using the BRISK descriptor. Then, the extracted features are passed on to the BoVW model, and each image in the dataset is then represented by a feature vector, which is the standardized. Using Support Vector Machines, defects of steel surfaces these feature vectors represent can be classify as: *patches, crazing, inclusion, rolled-in scale, pitted surface, and scratches*.

18.3.1 Feature Extraction

Feature extraction is done in three main stages: Feature detection, feature selection, and feature normalization.

Detection Feature detector used in this work is BRISK [17, 26] algorithm, which is organized in 4 stages:

(1) **Scale space Keypoint Detection**
 The keypoint detection methodology of BRISK is inspired by the Adaptive and Generic Accelerated Segment Test (AGAST) by Mair et al. [27] which is used to

detect regions of interest in an image. Keypoints detection is perform through a repeated down-sampling the input image into n octaves c_i and n intra-octaves d_i to build a pyramid scale-space. FAST detector [18] is the used to select keypoint candidates from the image pyramid.

(2) **Keypoint filtering and sub-pixel localization**
Keypoint candidates are then filtered by performing a 3D non- maxima suppression within the scale-space pyramid. The remaining keypoints are interpolated to a sub-pixel position; refer to [17] for details.

(3) **Orientation assignment**
Orientation assignment of sampling pattern adopted here is similar to the one used in DAISY descriptor [28]. Points in a pattern are paired with each other, represented by p_i and p_j and the gradients of these pairs are calculated. For the set of all long distance pairs, L, meaning pairs with a distance bigger than 13.67t (t is the scale of the keypoint), the gradient $\mathbf{g}(\mathbf{p}_i, \mathbf{p}_j)$ between the point pair $\mathbf{p}_i, \mathbf{p}_j$ is calculated using the formula

$$\mathbf{g}(\mathbf{p}_i, \mathbf{p}_j) = (\mathbf{p}_j - \mathbf{p}_i)\frac{I(\mathbf{p}_j, \sigma_j) - I(\mathbf{p}_i, \sigma_i)}{\|\mathbf{p}_j - \mathbf{p}_i\|} \tag{18.1}$$

where $I(\mathbf{p}_x, \sigma_x)$ is the smoothed value around $\mathbf{p}_x$, computed using a Gaussian filter with standard deviation σ_x .
The average direction $\mathbf{g}_{avg}$ of all possible long distance pair is computed as

$$g_{avg} = \begin{pmatrix} \mathbf{g}_{avg_x} \\ \mathbf{g}_{avg_y} \end{pmatrix} = \frac{1}{L} \cdot \sum_{(\mathbf{p}_i, \mathbf{p}_j) \in L} \mathbf{g}(\mathbf{p}_i, \mathbf{p}_j) \tag{18.2}$$

with L being the number of long distance pairs. The orientation of the keypoint can then be calculated by $\alpha = \arctan(\mathbf{g}_{avg_x}, \mathbf{g}_{avg_y})$, with $\mathbf{g}_{avg_x}$ and $\mathbf{g}_{avg_y}$ being the average gradient in horizontal and vertical direction, respectively.

(4) **Descriptor generation** of descriptor is performed by rotating the sampling pattern by α and comparing with each other all pair combinations with distance smaller than 9.75 t. A vector descriptor of length 512 is then generated and each element is defined as

$$b = \begin{cases} 1, & I(p_j, \sigma_j) > I(p_i, \sigma_i) \\ 0 & \text{otherwise.} \end{cases} \tag{18.3}$$

As this binary descriptor does not describe the absolute difference between two points, it is invariant to monotonic gray value changes between the scenes.

A comprehensive evaluation of benchmark datasets shows that BRISK's has better performance as reported in state-of-the-art algorithms and has less computational complexity, and requires less storage space [17]. Figure 18.1 shows some keypoints detected on the defects area of steel surfaces using a BRISK detector.

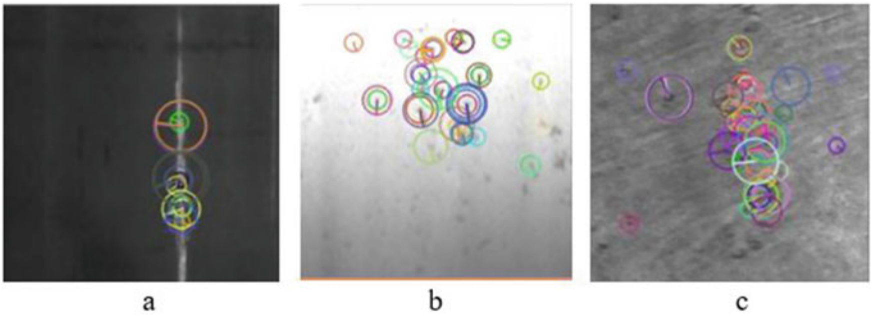

Fig. 18.1 keypoints detected using BRISK detector: **a** are the keypoints detected on the scratches defect class, **b** are the keypoints detected on the Pitted surface defect class, **c** are keypoints detected on the Rolled-in scale defect class

18.3.2 Feature Coding with Bag of Visual Word (BoVW)

The main idea of a bag of visual words is to represent an image as a single feature vector, where each feature consists of keypoints and descriptors in the image [7, 29]. BoVW is computed in two main steps: Dictionary Learning, in which features are extracted using techniques such as SIFT, SURF, and then visual dictionaries are learned using clustering (e.g., K-means clustering), the second step is coding that builds visual words by associating each feature to a visual word (nearest cluster center), and pooling that builds the histogram by counting the number of visual word occurrences for each image.

Dictionary Learning entails clustering keypoint descriptor vectors, where each cluster represents a codeword. Given $F = f_j \mid j = 1, 2, \ldots, N$ the set of unordered keypoint descriptors, with $f_j \in \mathbb{R}^D$ is a keypoint descriptor vector, N is the total number of keypoint descriptor vectors, and D is the number of component in a keypoint descriptor vector. Codebook is constructed using K-means, by clustering the N keypoint descriptor vectors into a codebook defined as: $C = \{c_k | k = 1, 2, ..., K\}$ where $c_k \in \mathbb{R}^D$.

Coding will represent each image of the codeword (elements of codebook). The coding step can be modeled using the function g defined as[30]:

$$g : \mathbb{R}^D \longrightarrow \mathbb{R}^K$$

$$f_j \longrightarrow \beta_j \tag{18.4}$$

where $\beta_j = (\beta_{k,j}) | k = 1..., K$ maps a descriptor vector f_j into the closest codeword c_k in the codebook according to the following hard coding equation.

$$\beta_{k,j} = \begin{cases} 1, & \text{if } k = \arg\min_{k \in \{1 \ldots, K\}} ||f_j - c_k||_2^2 \\ 0, & \text{otherwise} \end{cases} \tag{18.5}$$

Where $\beta_{k,j}$ is the k^{th} component of the encoded vector β_j.

Pooling Based on codewords, pooling performs the construction of the vector h, that gives the frequency of each codeword in the given image. This will mean that with an image with n descriptors, the k^{th} component of vector h is calculated as[30]:

$$h_k = \sum_{j=1}^{n} \beta_{k,j} \tag{18.6}$$

Algorithm 1 summarizes the steps used to construct the codebook and to form a global feature vector for every image in the dataset using the BRISK and BoVW.

Algorithm 1: Feature Extraction

FeatuteExtraction(D,K);
Input : $D = (d_i)_{i=1,2,\ldots,N}$(Dataset of steel surface of N images)
K (Number of codewords)
Output: $H = (h_i)_{i=1,2,\ldots,N}$(Extracted features of N imputed images)
// Use BRISK to generate feature descriptor;
for $i = 1$ ***to*** N **do**
  Generate the feature descriptor f_i of the image d_i;
end
// Generation of the Dictionary;
Generate the Dictionary C from $(f_j)_{i=1,2,\ldots,N}$ using K-means;
// Generation of feature using the dictionary C made with K codewords.;
for $i = 1$ ***to*** N **do**
  for *every descriptor vector* dv *in* d_i **do**
    Assign dv to the nearest codeword in C using Eq. 18.4.
  end
  Construct feature vector h_i using Eq. 18.6;
end
return $H = (h_1, h_2, \ldots, h_N)$;

18.3.3 Normalization of Feature Vectors

An Enhancement technique is essential to achieve good performance for the proposed classification model. The method named standardization has been used. Feature standardization is a pre-processing technique used to scale the dataset into one that has a 0 mean and a unit variance. The method has rescaled all the images to have a standard deviation one and mean as zero. The technique used for both the training and validation sets is such that, given a set n feature vectors $(X_i)_{1,2,\ldots,n}$ where each feature vector

X_i defined as $X_i = (x_i^1, x_i^2, \ldots, x_i^p)$ is transformed into $\tilde{X_i} = (\tilde{x_i^1}, \tilde{x_i^2}, \ldots, \tilde{x_i^p})$ and each x_i^j for $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, p$ is defined by [31]:

$$\tilde{x_i^j} = \frac{x_i^j - \mu_j}{\sigma_j} \tag{18.7}$$

where μ_j and σ_j are the mean and standard deviation of the j^{th} component of the n feature vectors, respectively. These mean and standard deviation are defined as

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} x_i^j \tag{18.8}$$

and

$$\sigma_j^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i^j - \mu_j)^2 \tag{18.9}$$

Both the standard deviation and mean have been calculated from the corresponding training and test sets.

18.3.4 Feature Classification

Features extracted by the bag of visual words approach, in the previous sections, are used as inputs to train and validate the SVM classifier. The SVM is a supervised learning method [32]. It has outstanding performance for object classification problems by achieving a minimal structural risk and minimizing the Vapnik-Chervonenkis (VC) dimension. It is used in many linear and non-linear classification problems. SVM was originally designed for binary classification, then developed to solve multi-class problems. For multi-classes problems, there are two strategies that can be used by combining several binary classifiers (One-Against-Rest also referred to as One-vs-All) or (one-against-one) [33]. I this work, SVM will then be used to classify steel image surface the features into one of the following classes Crazing, Inclusion, Patches, Pitted surface, Rolled-in scale, and Scratches. The implementation has been done using the LinearSVC package that supports multi-class using one-vs-the-rest scheme.

The proposed method of steel surface defects detection and classification using a bag of visual words and SVM classifier is illustrated in Fig. 18.2

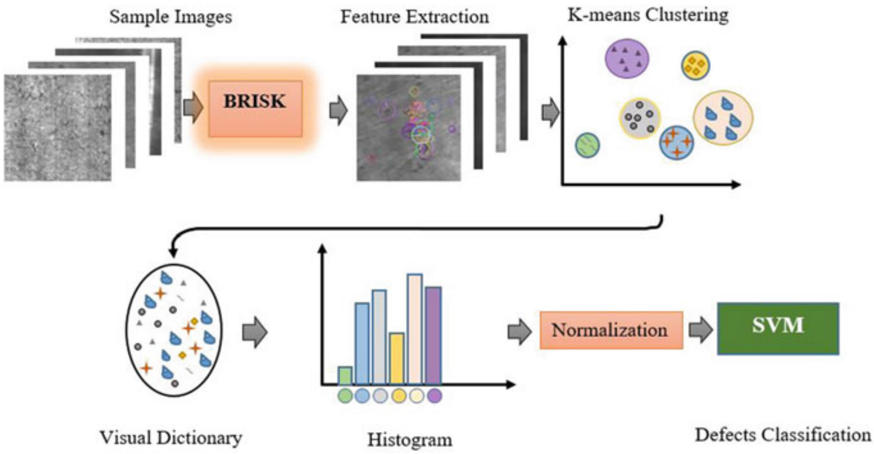

Fig. 18.2 The process of creating a bag of visual words for surface defects of steel strips

18.4 Experimental Setup and Results

18.4.1 Dataset

The NEU surface defect benchmark [34] dataset has been used in this paper. It is a publicly available dataset for steel surface defect classification. It has 1800 samples divided into 6 categories of surface defects of the hot-rolled steel strip, i.e., Crazing (Cr), Rolled-in-Scales (RS), Pitted Surface (PS), Patches (Pa), Inclusion (In), and Scratches (Sc). Each class has 300 samples. Figure 18.3 shows some examples of steel surface defects from the NEU dataset. The dataset is split into two sets, namely, training and validation. The training set includes 1440 samples, while the validation set includes 360; in other words, for each class there are 240 training samples and 60 validation samples.

18.4.2 Results and Discussion

This section evaluates the performance of the proposed steel surface defects detection and classification procedure. More precisely, the accuracy of detection will calculate and analyzed according to the number of defect categories and the estimated defect categories. The proposed method consists of the following phases: detection and description of keypoints from the images achieved using the BRISK descriptor, application of bag of visual words on the keypoint descriptors to represent each image in the dataset as a feature vector, which is then normalized, and SVM is thus trained and used to classify steel surfaces.

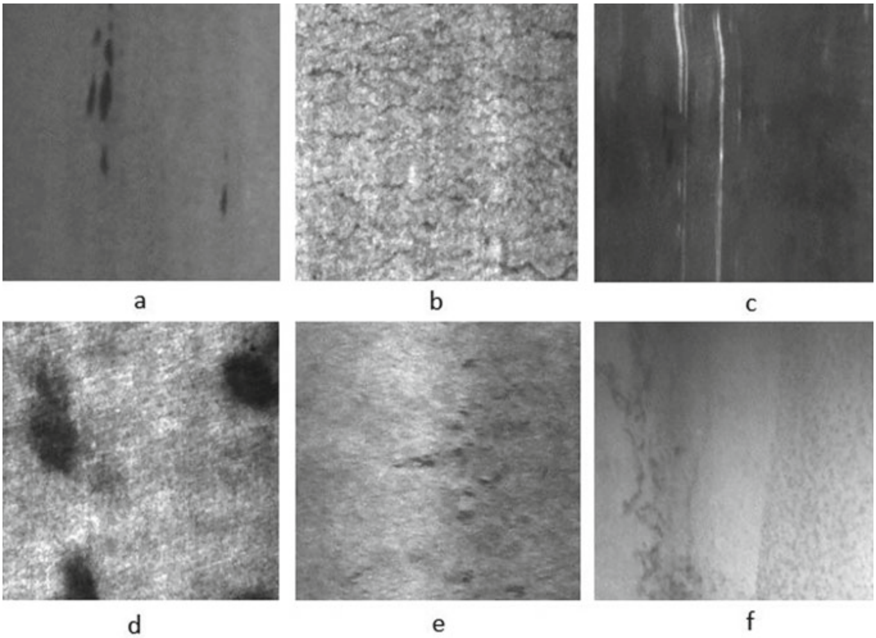

Fig. 18.3 Samples images of six typical surface defects in the NEU surface defect database including **a** Crazing, **b** Inclusion, **c** Patches, **d** Pitted surface **e** Rolled-in scale, **f** Scratches

Experiments were performed using the programming language Python (version 3.7.2) installed on a computer with computer with Intel Core(TM) i5-7200 CPU with 8GB of memory and equipped with Nvidia GeForce 940MX graphic card, a Processor Speed of 2.60 GHZ, and equipped with Windows 10 operating system. Furthermore, Opencv [35] Library, which provides a comprehensive suite of tools and algorithms for object detection and recognition, has been used. To evaluate the classification accuracy of 6 different defect classes, a total number of samples used is 360 (60 for each defect category). To measure the accuracy of detection, the following Detection Success Rate is used [33]:

$$\text{Detection Success Rate} = \frac{\text{Number of Samples Correctly Detected}}{\text{Total Number of Samples}} \tag{18.10}$$

Due to the large “intra group" diversity and “inter group" similarity in some classes of defects, some defects have been misclassified. For example, two defects of crazing are misclassified as Rolled-in scale, as shown in the confusion matrix, the category of patches has been classified totally correctly, and the error rate of inclusion is bigger than other defects.

Table 18.1 Comparison of classification accuracy of different methods

Methods	Accuracy (%)
HOG+GLCM-SVM [15]	90.16
HOG-KNN [16]	91.12
CDSDL [36]	94.25
SIFT+SVM [13]	94.74
Proposed method	95.00

Table 18.2 Comparison of accuracy of SVM and random forest

Classifier	Accuracy (%)
Random forest	91.95
SVM	95.00

18.4.3 Comparisons with State-of-the-Art

We have compared the accuracy obtained from our model with those achieved with existing methods found in [13, 15, 16, 36]. As shown in Table 18.1, our model achieves 95.00% classification accuracy compared to 94.74% in [13], 90.16% in [15], 91.12% in [16], 94.25% in [36]. Which shows that our model outperforms the above methods. Also, to evaluate our model, we have tested our model using the Random Forest classifier. Table 18.2 shows that our model achieved quite a good accuracy when we used the SVM classifier.

18.5 Conclusion

A classification model, based on a bag of visual word technique with a BRISK descriptor extractor and Support Vector Machine, has been proposed for defects classification in the steel strip surface images. The effectiveness of the proposed method has been evaluated using 360 images from the NEU surface defect benchmark dataset. Experiments conducted gave a recognition rate of 95:00%, which is superior to many existing methods. In future work, a larger dataset with an additional number of defects will be done.

References

1. Mordia, R., Verma, A.-K.:Visual techniques for defects detection in steel products: a comparative study. Eng. Fail. Anal. 106047 (2022)

2. Jin, Q., Chen, L.:A survey of surface defect detection of industrial products based on a small number of labeled data. arXiv preprint arXiv, 2203-05733 (2022)
3. Zheng, X., Zheng, S., Kong, Y., Chen, J.: Recent advances in surface defect inspection of industrial products using deep learning techniques. Int. J. Adv. Manuf. Technol. **113**(1), 35–58 (2021)
4. Czimmermann, T., Ciuti, G., Milazzo, M., Chiurazzi, M., Roccella, S., Oddo, C.-M., Dario, P.: Visual-based defect detection and classification approaches for industrial applications-a survey. Sensors **20**(5), 1459 (2020)
5. Luo, Q., Fang, X., Su, J., Zhou, J., Zhou, B., Yang, C., Liu, L., Gui, W., Tian, L.: Automated visual defect classification for flat steel surface: a survey. IEEE Trans. Instrum. Meas. **69**(12), 9329–9349 (2020)
6. Luo, Q., Fang, X., Liu, L., Yang, C., Sun, Y.: Automated visual defect detection for flat steel surface: a survey. IEEE Trans. Instrum. Meas. **69**(3), 626–644 (2020)
7. ZhiJie, L.: Image classification method based on visual saliency and bag of words model. In: 2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), pp. 466–469. IEEE (2015)
8. Huzaifa, M., Suwardi, I.-S.: August. Car model recognition from frontal image using BRISK. In: 2017 International Conference on Electrical Engineering and Computer Science (ICECOS), pp. 104–108. IEEE (2017)
9. Hao, Z., Kaidong, L., Feng, Q.: Improvement of word bag model based on image classification. In: 2019 IEEE 1st International Conference on Civil Aviation Safety and Information Technology (ICCASIT), pp. 272–276. IEEE (2019)
10. Lima, G.-V., Saito, P.-T., Lopes, F.-M., Bugatti, P.H.: Classification of texture based on bag-of-visual-words through complex networks. Expert Syst. Appl. **133**, 215–224 (2019)
11. Liu, L., Qiu, T., Lu, Y., Chen, Q., Suen, C.-Y.: Combination of spatially enhanced bag-of-visual-words model and genuine difference subspace for fake coin detection. Expert Syst. Appl. **159**, 113551 (2020)
12. Nabizadeh-Shahre-Babak, Z., Karimi, N., Khadivi, P., Roshandel, R., Emami, A., Samavi, S.: Detection of COVID-19 in X-ray images by classification of bag of visual words using neural networks. Biomed. Signal Process. Control **68**, 102750 (2021)
13. Suvdaa, B., Ahn, J., Ko, J.: Steel surface defects detection and classification using SIFT and voting strategy. Int. J. Softw. Eng. Its Appl. **6**(2), 161–166 (2012)
14. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. IEEE Trans. Pattern Anal. Mach. Intell. **27**(10), 1615–1630 (2005)
15. Zaghdoudi, R., Seridi, H., BOUDIAF, A.: Detection and classification of steel defects using machine vision and SVM classifier. In: 2nd International Conference on Automatic control, Telecommunication and Signals (ICATS'17) (2017)
16. Boudiaf, A., Moussaoui, A., Zaghdoudi, R., Mentouri, Z., Saadoune, A.: An automatic system to surface defect classification of hot rolled steel (2018)
17. Leutenegger, S., Chli, M., Siegwart, R.-Y.: November. BRISK: Binary robust invariant scalable keypoints. In: 2011 International conference on computer vision, pp. 2548–2555. IEEE (2011)
18. Rosten, E., Drummond, T.: May. Machine learning for high-speed corner detection. In: European Conference on Computer Vision, pp. 430–443. Springer, Berlin (2006)
19. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). Comput. Vis. Image Underst. **110**(3), 346–359 (2008)
20. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: European conference on computer vision, pp. 778–792. Springer, Berlin (2010)
21. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: November. ORB: An efficient alternative to SIFT or SURF. In: 2011 International Conference on Computer Vision, pp. 2564–2571. IEEE (2011)
22. Mak, M.-W., Guo, J., Kung, S.-Y.: PairProSVM: protein subcellular localization based on local pairwise profile alignment and SVM. IEEE/ACM Trans. Comput. Biol. Bioinf. **5**(3), 416–422 (2008)

23. Joachims, T.: Text categorization with support vector machines: Learning with many relevant features. In: European Conference on Machine Learning, pp. 137–142. Springer, Berlin (1998)
24. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: Fast retina keypoint. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–517. IEEE (2012)
25. Liu, Y., Zhang, H., Guo, H., Xiong, N.-N.: A fast-brisk feature detector with depth information. Sensors **18**(11), 3908 (2018)
26. Schwind, P., d'Angelo, P.: Evaluating the applicability of BRISK for the geometric registration of remote sensing images. Remote Sens. Lett. **6**(9), 677–686 (2015)
27. Mair, E., Hager, G.-D., Burschka, D., Suppa, M., Hirzinger, G.: September. Adaptive and generic corner detection based on the accelerated segment test. In: European Conference on Computer Vision, pp. 183–196. Springer, Berlin (2010)
28. Tola, E., Lepetit, V., Fua, P.: Daisy: an efficient dense descriptor applied to wide-baseline stereo. IEEE Trans. Pattern Anal. Mach. Intell. **32**(5), 815–830 (2009)
29. Kesorn, K., Poslad, S.: An enhanced bag-of-visual word vector space model to represent visual content in athletics images. IEEE Trans. Multimedia **14**(1), 211–222 (2011)
30. Molefe, M., Tapamo, J.R. : Classification of rail welding defects based on the bag of visual words approach. In: Mexican Conference on Pattern Recognition, pp. 207–218, (2021). Liu, Y., Zhang, H., Guo, H., Xiong, N.-N.: A fast-brisk feature detector with depth information. Sensors, **18**(11), 3908 (2018)
31. Rahman, A., Zunair, H., Rahman, M.-S., Yuki, J.-Q., Biswas, S., Alam, M.-A., Alam, N.B., Mahdy, M.-R.-C.: Improving malaria parasite detection from red blood cell using deep convolutional neural networks. arXiv preprint arXiv:1907.10418 (2019)
32. Vapnik, V.-N.: An overview of statistical learning theory. IEEE Trans. Neural Netw. **10**(5), 988–999 (1999)
33. Ozseven, T., Yasar, E.: Theoretical Investigations and Applied Studies in Engineering (2019)
34. He, Y., Song, K., Meng, Q., Yan, Y.: An end-to-end steel surface defect detection approach via fusing multiple hierarchical features. IEEE Trans. Instrum. Meas. **69**(4), 1493–1504 (2019)
35. Howse, J., Joshi, P., Beyeler, M.: Opencv: Computer Vision Projects with Python. Packt Publishing Ltd. (2016)
36. Zhou, S., Chen, Y., Zhang, D., Xie, J., Zhou, Y.: Learning a class-specific and shared dictionary for classifying surface defects of steel sheet. ISIJ Int. **57**(1), 123–130 (2017)