Rust Detection of Steel Structure via One-Class Classification and L2 Sparse Representation with Decision Fusion

Guizhong ZHANG†, Nonmember, Baoxian WANG††††, Member, Zhaobo YAN†††, Yiqiang LI††, and Huaizhi YANG††††, Nonmembers

SUMMARY In this work, we present one novel rust detection method based upon one-class classification and L2 sparse representation (SR) with decision fusion. Firstly, a new color contrast descriptor is proposed for extracting the rust features of steel structure images. Considering that the patterns of rust features are more simplified than those of non-rust ones, one-class support vector machine (SVM) classifier and L2 SR classifier are designed with these rust image features, respectively. After that, a multiplicative fusion rule is advocated for combining the one-class SVM and L2 SR modules, thereby achieving more accurate rust detecting results. In the experiments, we conduct numerous experiments, and when compared with other developed rust detectors, the presented method can offer better rust detecting performances.

key words: rust detection, color contrast descriptor, one-class classification, L2 sparse representation, decision fusion

1. Introduction

Rust is one of the most common surface damages of steel structures, and the rust detection in a timely manner is an effective method for decreasing possibility of accident and extending service life of steel structure. Recently, computer vision has been successfully applied in many industrial applications, and has been also used for rust detection. The reader can refer to one review of automated inspection methods for steel surfaces using computer vision techniques [1].

Owing to the color feature of rust is obviously different from that of non-rust, the color-based rust detection methods have been presented in some previous works. For instance, Lee et al. computed nine statistic variables of RGB color channels from one image, and three variables were selected to construct a multivariate discriminant function for recognizing the rusts [2]. This rust detector can achieve good detecting results in some specific conditions, but the presented discriminant function with fixed parameters cannot deal with other complicated disturbances. In [3], the rust color spectrum in the RGB space is used, and one rust color range is defined for recognizing the rusts. Although this color spectrum based rust detector performs efficiently, it is powerless to address some environmental troubles. With the separated image regions, rusts can be detected via one thresholding operation in the YCbCr space [4]. Due to that the color features are generally high sensitive to the illumination, the simple utilization (i.e., empirical formulas [2], threshold setting [3], [4]) of color information is not suitable for the practical application. To alleviate this issue, some pattern classifiers are applied for distinguishing the rust from backgrounds. In [5], K-means method is used for rust discrimination, and this model only works on the condition that there are rust defects in the images. For improving the recognition accuracy, artificial neural network is exploited for separating the rusts from non-rusts [6]. Similarly, SVM is also applied for finding the optimal separating hyper-plane between rusts and non-rusts [7]. In general, these rust detectors based on binary classification can achieve good rust detecting results. However, they cannot correctly identify rusts with several particular conditions (such as complex and changeable non-rust contents, varied background clutters, etc.). One possible reason may be that: the patterns of non-rusts are more diversified than those of rusts, and binary classifier is not effective with an imbalanced class distribution of training samples. Recently, convolutional neural network is used for rust detection [8]. However, this method adopts one strategy of dividing the images into sub-blocks, which cannot locate rusts at a pixel level.

To recognize the rusts under complex conditions, we attempt to present one new rust detector in this letter. As shown in Fig. 1, the developed rust detection model consists of four steps: color-contrast feature extraction, one-class SVM classification, L2-SR classification and multiplicative decision fusion. The main contributions are as follows: 1) One novel rust feature is implemented by using the specific value of different color spaces, which is efficient and less affected by illumination. 2) In view of the more simplified rust features, two potential classifiers (i.e., one-class SVM and L2-SR) are applied, which can address the imbalanced training issue, and the resultant rust detector becomes more discriminative. 3) One multiplicative fusion strategy is used to combine the two recognition results, thereby leading to more robust detecting performances.

Manuscript publicized November 11, 2019.
1The author is with Shandong University, Shandong, 250100, China.
2The authors are with Shijiazhuang Tiedao University, Hebei, 050043, China.
3The author is with Key Laboratory for Health Monitoring and Control of Large Structures of Hebei Province, China.
4The author is with Beijing-Shanghai High-Speed Railway Co., Ltd., Beijing, 100038, China.
a) E-mail: wbx1025@163.com (Corresponding author)
DOI: 10.1587/transinf.2019EDL8178
2. Proposed Rust Detection Method

2.1 Rust Feature Extraction

As for the non-uniform illuminations, images are often converted into other color space (e.g., YCbCr) from RGB space [9]. In this letter, one simple but effective rust feature extraction is developed. Specifically, the specific values of features are computed as follows.

\[
x = [R/G, \ B/G, \ R/B]
\]

Here \( R, G, B \) denotes the pixel intensity value of RGB space; \( x \) represents the calculated color-contrast feature for one image pixel.

For validating the effectiveness of presented feature descriptor, two images are collected from the same scene with different illuminations. Then, the original RGB, \( R/G, \ B/G, \ R/B \) image spaces are shown in Fig. 2. From the comparisons, one can see that the rust damage regions using the developed feature spaces are not affected by the non-uniform illuminations.

2.2 One-Class SVM Classification

As shown in Fig. 1, the coating colors (including silver, blue, white, etc.) of steel structure are various. In comparison, the rusts come from the chemical reaction between the broken surfaces and external environments, and the corresponding color features are more similar. Based on this inference above, we exploit one-class SVM for recognizing the rusts.

In this work, there is only one positive class (i.e., rust) for training the rust detector. Mathematically, one-class SVM is used to learn the restricted domains around the positive rust samples. Here, given \( L \) rust samples \( x_i \in \mathbb{R}^3 \) \( (i = 1, \ldots, L) \), one-class SVM aims to identify as many rust training samples as possible, while it minimizes the possibility of recognizing the outlier samples. The training task of one-class SVM is to optimize the following problem.

\[
\min_{R \in \mathbb{R}, \xi_i} \ \left\{ R^2 + \lambda \sum_{i=1}^{L} \xi_i \right\}
\]

\[
\text{s.t. } \quad ||\phi(x_i) - c||^2 \leq R^2 + \xi_i \]
\[
\xi_i \geq 0, \quad i = 1, \ldots, L
\]

Here, \( R \) is the radius of hypersphere around the positive rusts; \( c \) is the corresponding center of hypersphere; \( \xi_i \) is the slack variable of \( i \)-th training sample; \( \lambda \) is one regularization parameter; \( \phi(\bullet) \) is the non-linear function. Via the dual problem optimization, these parameters mentioned above can be calculated in the training phase. Finally, the label of one pixel \( l \) can be determined using the following equation.

\[
i_l = \text{sgn}\left( \left[ R^2 - ||\phi(x_l) - c||^2 \right]/R^2 \right)
\]

Here \( \text{sgn}(\bullet) \) is the symbolic function, and the input \( y \) is recognized as the rust sample when \( i_l = 1 \).

2.3 L2-SR Classification

As discussed in Sect. 2.2, these color features of rust pixels are more similar than those of non-rust ones. Because of this, K-means is used to cluster rust images, and has achieved good detecting results [5]. Then, motivated by this framework, we also apply one L2 sparse representation model, which takes full advantage of the clustering characteristic of rust features.

Specifically, suppose that there is one feature set \( D = [D_r, D_{nr}] = [d_1, \ldots, d_p, d_{p+1}, \ldots, d_{p+q}] \). Here, \( D_r \) denotes the rust feature set, and \( D_{nr} \) is the non-rust feature set. In this letter, the number \( q \) of non-rust features is much larger than the number \( p \) of rusts. Then, one new feature \( y \) can approximately lie in the linear span of \( D \), which is given in the following equation.

\[
y \approx Da = a_1d_1 + \ldots + a_pd_p + a_{p+1}d_{p+1} + \ldots + a_{p+q}d_{p+q}
\]

With the inference above, the coefficient vector \( a \) of sample \( y \) should be sparse, and we need to solve the following problem.

\[
a^* = \arg\min_{a} \left\{ ||y - Da||_2^2 + \beta||a||_2^2 \right\}
\]

Here \( \beta \) is one small regularization parameter, and the optimal solution can be obtained via the ridge regression method as follows.

\[
a^* = (D^TD + \beta\mathbf{I})^{-1}D^Ty
\]

Mathematically, the coefficient \( a^* \) has two subsets, i.e., \( a^* = [a_r^*, a_{nr}^*] = [a_1, \ldots, a_p, a_{p+1}, \ldots, a_{p+q}] \). In addition, \( a_r^* \) represents the relationship between the input \( y \) and rust features,
and \( a^* \) denotes the correlation between the input \( y \) and non-rust ones. Based on this, one mean-pooling strategy is used for determining the label of input \( y \).

\[
i^* = \text{sgn}\left(\frac{1}{p} \sum_{i=1}^{p} a_i - \frac{1}{q} \sum_{j=1}^{q} a_{p+j}\right)
\]

(7)

where \( \text{sgn}(\bullet) \) is the symbolic function, and the input \( y \) is determined as the rust sample when \( i^* = 1 \).

2.4 Decision Fusion

For better rust detecting performance, we attempt to combine the two recognition results via one multiplicative fusion rule. For the one-class SVM, one detecting probability is computed as follows.

\[
p^i_{\text{svm}} = \exp(J_{\text{svm}}) = \exp\left([R^2 - \|\phi(x) - c\|^2]/R^2\right)
\]

(8)

For L2-SR model, we need to calculate another one detecting probability

\[
p^i_{\text{sr}} = \exp(J_{\text{sr}}) = \exp\left(\frac{1}{p} \sum_{i=1}^{p} a_i - \frac{1}{q} \sum_{j=1}^{q} a_{p+j}\right)
\]

(9)

Then, two corresponding adapted weights are computed

\[
w_{\text{svm}} = \frac{p^i_{\text{svm}}}{p^i_{\text{svm}} + p^i_{\text{sr}}} (10)
\]

\[
w_{\text{sr}} = \frac{p^i_{\text{sr}}}{p^i_{\text{svm}} + p^i_{\text{sr}}} (11)
\]

here \( w_{\text{svm}} \) is the weight of one-class SVM model, and \( w_{\text{sr}} \) is the weight of L2-SR model. With the two weights, one multiplicative formula is exploited for fusing these two modules, and the label of input \( y \) can be obtained

\[
i^* = \text{sgn}(w_{\text{svm}} \times J_{\text{svm}} + w_{\text{sr}} \times J_{\text{sr}})
\]

(12)

It is noted that the presented multiplicative formula is more effective and convenient than other alternative additive formula based on the empirical observation.

3. Experimental Results and Discussions

To evaluate the presented method, we compared it with some representative rust detection models. They are referred to as the statistical color-rust detection (SCR) [2], SVM based rust detection (SVMR) [7] and deep learning based rust detection (DLR) [8]. To verify the effectiveness of presented rust feature, RGB/Lab/YCbCr pixel features are applied with the following rust detector, respectively. Besides, one-class SVM (OC-SVM) and L2-SR models are also utilized using the proposed rust feature, which can validate the performance of multiplicative fusion rule. These comparison results of self-compared rust detectors are shown in the Table 1. For fairness, our proposed rust detector and all the compared ones are programmed using the same computer (i.e., Matlab2017b, Win7 x64 system, Intel 2.40GHz CPU, 64GB RAM.). In this work, 100 steel structure images with a resolution of 4608 × 3456 px2 are collected from several steel structures outside in the experiments (e.g., holder, steel beam, canopy, etc.) at Shijiazhuang Tiedao University, China. For these images, these typical challenges of rust detection in real-world environments are considered, such as illumination, multicolor, background noise, blurring, etc. Besides, 60 images are randomly chosen for building the rust detectors mentioned above, and the remaining ones are used for validating these rust detection models.

3.1 Parameter Setting

For OC-SVM, one thousand rust features (i.e., \( L = 1000 \)) is applied; the non-linear function \( \phi(\bullet) \) is set to be the Gaussian function \( \phi(a,b,x) = \exp(-b|x-a|) \); and the LIBSVM package [10] is used to implement the one-class SVM training. For L2-SR, we collect 100 rust features and 1000 non-rust ones from the existed rust damage images. In this letter, there are two regularization parameters (i.e., \( \lambda \) and \( \beta \)), they are to be determined via one trial-and-error procedure. Specifically, \( \lambda \) is set to be 0.1 and \( \beta \) is set to be 0.05.

In this letter, we try our best to implement the compared rust detection methods. For SCR, as described in the original work, the same rust features and discriminant function is used. With the same amount of rust and non-rust training samples, the SVMR model also adopts LIBSVM package [10] for rust recognition. The implementation of DLR is based on the VGG network, which utilizes the 32 × 32 sliding window size.

3.2 Performance Comparison

In this letter, some representative rust detecting results are illustrated in Fig. 3. Specifically, the first row exhibits the original images. For better comparison, the manually labeled ground-truth is shown in the second row. Then, the rust detecting results of SCR [2], SVMR [7], DLR [8] and our method are shown in the subsequent four rows of Fig. 3, respectively. In addition, two metrics (i.e., precision rate (PR) and recall rate (RE)) are used, and their definitions are as follows.

\[
PR = \frac{T_{\text{est}}}{T_d}, \quad RE = \frac{T_{\text{est}}}{T_g}
\]

(13)

where \( T_d \) is the total number of detected rust pixels, \( T_g \) is the number of manually labeled rust pixels, and \( T_{\text{est}} \) is the number of correctly detected rust pixels. For clear comparison, the average PRs and average REs for all the testing images

| Methods | Average precision rate (%) | Average recall rate (%) |
|---------|---------------------------|------------------------|
| SCR     | 41.6                      | 51.7                   |
| SVMR    | 70.6                      | 65.3                   |
| DLR     | 73.5                      | 66.9                   |
| RGB     | 72.2                      | 74.8                   |
| Lab     | 78.7                      | 75.1                   |
| YCbCr   | 77.6                      | 78.4                   |
| OC-SVM  | 88.5                      | 84.8                   |
| L2-SR   | 89.6                      | 83.6                   |
| Proposed| 91.8                      | 86.6                   |
are shown in Table 1.

From the compared results of Fig. 3 and Table 1, one can be see that SCR is the least effective method, owing to the poor adaptability of its adopted discriminant function with fixed parameters. Due to that the background contents is more diversified that the used non-rust training samples, the SVMR cannot achieve favorable results for dealing with the new testing images (i.e., the ellipse dashed boxes in Fig. 3 (d)). With the help of multi-layer feature learning, DLR has obtained one higher PR value than SVMR. However, from the qualitative evaluation (see Fig. 3 (e)), DLR cannot detect some rust regions. A possible reason is that DLR is easy to over-fit when rust training samples are not enough. In contrast, as shown in Table 1, OC-SVM and L2-SR models have better evaluation metrics (i.e., PR and RE), which can be attributed to the appropriate application of clustering characteristic of rust features.

Furthermore, compared with the self-compared rust detectors, our presented model has achieved the best performance, and there are two reasons: 1) For different backgrounds, the proposed color-contrast feature extraction is more stable than other rust features; 2) Two effective rust detection modules are combined via one multiplicative fusion rule, which can lead to better rust detecting performances.

4. Conclusion

In this work, one novel rust detecting model has been proposed. First, we present one color-contrast feature extraction, which is efficient and less affected by different illuminations. Then, by using the clustering characteristic of rust features, one-class SVM and L2-SR modules are exploited for recognizing the rusts, respectively. Finally, to obtain more stable performance, a multiplicative fusion strategy is used to combine these two recognition results. Experiments on steel structure images verify that the effectiveness of proposed rust damage detecting model.

Acknowledgments

This paper is supported by National Key Research and Development Program (2016YFC0802207), National Natural Science Foundation of China (51808358, 51978423).

References

[1] N. Neogi, D.K. Mohanta, and P.K. Dutta, “Review of vision-based steel surface inspection systems,” EURASIP Journal on Image and Video Processing, 2014(1):50, 2014.
[2] S. Lee, L.-M. Chang, and S. Miroslaw, “Automated recognition of surface defects using digital color image processing,” Automation in Construction, vol.15, no.4, pp.540–549, 2006.
[3] H.-K. Shen, P.-H. Chen, and L.-M. Chang, “Automated steel bridge coating rust defect recognition method based on color and texture feature,” Automation in Construction, vol.31, pp.338–356, 2013.
[4] Y. Li, A. Kontsos, and I. Bartoli, “Automated rust-defect detection of a steel bridge using aerial multispectral imagery,” Journal of Infrastructure Systems, vol.25, no.2, 04019014, 2019.
[5] S. Lee, “Digital color image processing system for civil infrastructure health assessment and monitoring: Steel bridge coating case,” Ph.D. Dissertation, Purdue University, U.S.A., 2005.
[6] Y.C. Yang, “Smart Color Image Recognition for Steel Bridge Rust Inspection,” Master Thesis, National Taiwan University, Taiwan, 2009.
[7] P.-H. Chen, H.-K. Shen, C.-Y. Lei, and L.-M. Chang, “Support-vector-machine-based method for automated steel bridge rust assessment,” Automation in Construction, vol.23, pp.9–19, 2012.
[8] D.J. Atha and M.R. Jahanshahi, “Evaluation of deep learning approaches based on convolutional neural networks for corrosion detection,” Structural Health Monitoring, vol.17, no.5, pp.1110–1128, 2018.
[9] P. Shih and C. Liu, “Comparative assessment of content-based face image retrieval in different color spaces,” International Journal of Pattern Recognition and Artificial Intelligence, vol.19, no.7, pp.873–893, 2008.
[10] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” ACM Trans. Intell. Syst. Tech, vol.2, no.3, 1–27, 2011.
[11] Y. Yong and Z. Weigang, “Curvelet transform-based identification of void diseases in ballastless track by ground-penetrating radar,” Structural Control and Health Monitoring, vol.26, no.4, e2322, 2019.