Automatic Annotation of Web Images for Domain-Specific Crack Classification

66

Peter Cheng-Yang Liu and Nora El-Gohary

Abstract

Manual visual crack detection and classification for inspection of civil infrastructure is time-consuming and labor-intensive. Many automatic crack detection and classification algorithms have, thus, been developed in the past decade, several of which achieved acceptable performance results for specific applications and using large datasets for training. However, developing training data for automatic crack classification is not an easy task. It requires a large dataset in terms of quantity and variability, as well as well-trained professionals to label the dataset. Hence, there is a need for efficient ways to develop well-labeled datasets that could not only reduce human effort, but also adapt to diverse inspection contexts for improved classification performance. To address this need, this paper proposes a data retrieval and annotation method to automatically retrieve and label crack images from the Web. The dataset can be used as pseudo training data for supervised machine learning-based crack classification algorithms. The proposed method incrementally retrieves and labels crack images. A weak Convolutional Neural Network classifier first learns from a limited set of Web images, and then acts as a machine annotator and further labels a larger size of data. The proposed method was able to retrieve and label a set of images with 95% labeling recall, which shows that the proposed approach is promising.

Keywords

Crack classification • Machine learning •
Convolutional neural networks • Automatic training data generation •
Image retrieval • Image annotation

66.1 Introduction

Cracks are a common type of distress for civil infrastructure. Manual visual inspection for crack detection is time-consuming and costly, and can put inspectors at a safety risk. Automatic crack detection and classification is, thus, necessary to reduce human involvement. Crack detection has been well studied in the past decades [1]. Many automatic crack detection algorithms have been developed, and achieved acceptable performance results. However, automatic crack classification is still an open research area. Different types of cracks lead to different structural deterioration risks. For example, fatigue cracks, also known as alligator cracks, might cause disintegration of the surface, and become potholes [2]. Identifying the types of detected cracks is, thus, critical for deterioration detection/prediction and maintenance decision making.

In the computer vision domain, supervised learning is still the state-of-the-art approach for image classification problems [3]. In this case, a labeled dataset is essential for training. However, two problems exist in this regard. First, on one hand, training data for crack classification tasks are not easily available/accessible. On the other hand, developing training data for automatic

P. C.-Y. Liu · N. El-Gohary (✉)
Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign,
205 North Mathews Ave, Urbana, IL 61801, USA
e-mail: gohary@illinois.edu

P. C.-Y. Liu
e-mail: cyliu4@illinois.edu

I. Mutis and T. Hartmann (eds.), *Advances in Informatics and Computing in Civil and Construction Engineering*,
https://doi.org/10.1007/978-3-030-00220-6_66

Fig. 66.1 Comparison between images from a fixed setting acquisition system and a natural scene. The left image is acquired by a system with video cameras mounted on a car [7]. The right image is an arbitrary image downloaded from the Web [9]

crack classification is not an easy task. It requires a large dataset in terms of quantity and variability, as well as well-trained professionals to label the dataset. Second, current crack detection datasets that could be used for classification training are not suitable for training sufficiently adaptive classification models. Existing inspection approaches are moving towards using more general and dynamic platforms. For example, in bridge inspection, unmanned aerial vehicles (UAVs) are used for inspecting deck surfaces and building 3D models of the entire structures [4–6]. Images collected from such platforms contain varied heights and noisy objects in the scene. Current crack detection datasets, however, include images that are taken from specific mobile platforms with fixed settings (e.g., a vehicle with a fixed camera) [7, 8] (Fig. 66.1 shows a comparison between an image taken in a fixed setting and another in a natural setting). Classification models trained on such datasets are, thus, not adaptive to varying, non-ideal settings (e.g., with high levels of noise). Hence, there is a need for efficient ways to develop well-labeled datasets that could not only reduce human effort, but also adapt to diverse settings for improved classification performance.

To address this need, this paper proposes a data retrieval and annotation method to automatically retrieve and label crack images from the Web. The proposed method incrementally retrieves and labels diverse crack images. A weak Convolutional Neural Network (CNN) classifier first learns from a limited set of Web images, and then acts as a machine annotator and further labels a larger size of data. Most of the images retrieved from the Web are natural scene images—they are from different sources, taken from different angles, contain noisy objects in the scene, etc.

66.2 Related Work

66.2.1 Crack Detection and Classification

Many computer vision-based automatic crack detection algorithms have been developed in the past decades, which can be classified into three approaches: crack image thresholding, patch-based classification, and depth-based methods [8]. For example, Chambon and Moliard [7], Oliveira et al. [10], Zou et al. [11], and Salman et al. [12] took an image thresholding approach; they assumed that a crack is darker locally and tried to find the local intensity minimum. Zhang et al. [1] and Varadharajan et al. [13] took a patch-based classification approach; they cropped images into patches and used machine learning methods to check if each region contained cracks. Mertz [14], Yamada et al. [15], and Yu et al. [16] took a depth-based approach; they extracted cracks from depth information.

In addition to detecting cracks in road surface images, analyzing defects and classifying crack types also received research attention. For example, Byoung and Hosin [17] implemented three neural networks—image-based neural network, histogram-based neural network, and proximity-based neural network—to classify images into crack types: alligator, block, longitudinal, and transverse cracks. More recently, Saar and Talvik [18] improved the performance by extracting potential crack pixels before classifying them. Recent work (e.g., Radopoulou and Brilakis [19]) tried to distinguish more defects such as potholes. Hoang and Nguyen [20] compared different feature sets with multiple machine learning algorithms—support vector machines, artificial neural networks, and random forest—for crack type classification. Despite the importance of existing efforts, the majority of these efforts used experimental datasets that were taken in a fixed setting or from a close look at the surface.

66.2.2 Crack Types

A number of state and federal authorities developed standards and guidelines that describe different crack classes to assess pavement conditions. For example, the Washington State Department of Transportation (DOT) [21] divided cracks into alligator, longitudinal, transverse, block, and rigid cracks, according to width, length, pattern, and appearance frequency features. The American Association of State Highway and Transportation Officials [22] introduced more crack classes, such as joint reflection cracks from PCC slab. Most recently, the U.S. DOT [23] developed their Distress Identification Manual for the Long-Term Pavement Performance Program.

66.2.3 Convolutional Neural Networks

The Convolutional Neural Networks (CNN) algorithm is a neural network machine learning algorithm, which does feature learning and classification in one shot. Instead of conventional hand-crafted features, CNN is proven to be more effective and accurate in learning image features directly from training data [24]. CNN has shown promising results in image classification problems, both for general images [3] and domain-specific images such as medical imaging [25–27]. In the civil and infrastructure domain, Zhang et al. [1] and Grandsaert [5], for example, attempted to apply deep CNN in crack detection. As a supervised classifier, the success of CNN relies on large amount of training data, which are neither available nor easy to develop/label for natural scene crack classification problems.

66.2.4 Web Image Retrieval and Existing Datasets

Training data are critical to the performance of image classification algorithms. Outside of the civil infrastructure domain, researchers tended to retrieve such image data from Web resources [28–30]. For example, Li and Fei-Fei [31] used keyword queries to retrieve images and used the top-ranked ones for training a classifier. Schroff et al. [32] leveraged text information associated with the images to help in ranking. Yao et al. [33] used multiple query expansion to retrieve more relevant results.

In the civil infrastructure domain, researchers tend to collect such image data using data acquisition platforms. For example, there are datasets for crack detection [7, 8, 10, 11, 34, 35]. Some of them contain less than 300 images [7, 10, 11], which are not enough for deep learning. Also, all of them are taken from a close view of the surface, which is not sufficient in terms of natural scene analysis. Furthermore, the datasets are not labeled for the crack types.

66.3 Methodology

An approach to incrementally retrieve and automatically annotate web images for supporting crack classification is proposed. The proposed method aims to incrementally retrieve and label diverse crack images, which could serve as pseudo training data for supervised machine learning-based crack classification algorithms. A weak CNN classifier first learns from a limited set of Web images, and then acts as a machine annotator and further labels a larger size of data. The research methodology, thus, includes four primary steps (as per Fig. 66.2): (1) class image retrieval, (2) weak CNN classification, (3) image retrieval and annotation, and (4) evaluation.

66.3.1 Class Image Retrieval

A set of images was retrieved and cleaned. The images were retrieved from the Web, using the Google Image Search and a set of class keywords, such as “fatigue crack”. The downloaded images were then automatically cleaned based on their associated text. For example, non-real images (e.g., those from educational sources such demonstration figures and lecture slides) were removed. The 100-top-ranked images (i.e., ranked by Google) were then used as training data for the proposed supervised CNN classifier/annotator (Sect. 66.3.2), where the keywords were used as labels.

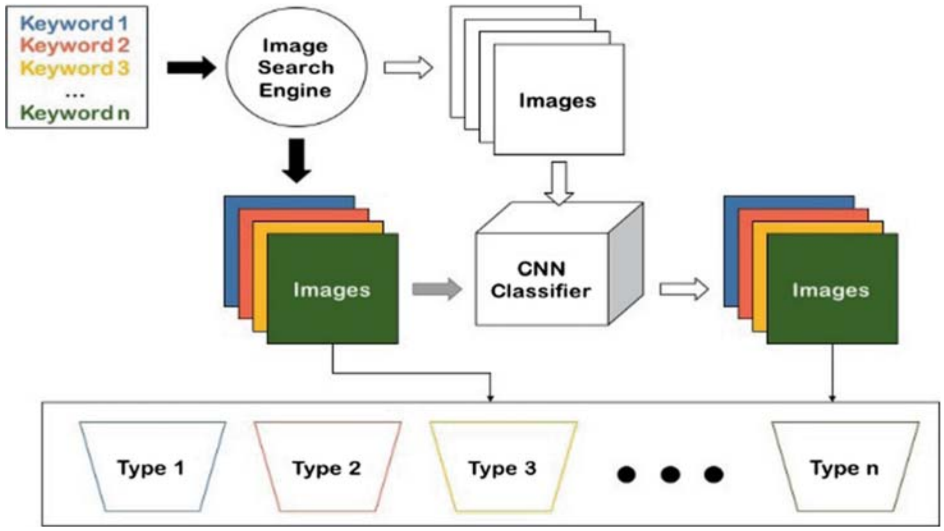

Fig. 66.2 Proposed data retrieval and annotation approach for creating pseudo training data

Table 66.1 Characteristics of the developed weak CNN classifier model

Layer no.	Example	Kernel size	Output size
1.	conv	3 × 3 (32)	128 × 128 × 32
2.	pool	2 × 2 (32)	64 × 64 × 32
3.	conv	3 × 3	64 × 64 × 32
4.	pool	2 × 2	32 × 32 × 32
5.	conv	3 × 3	32 × 32 × 64
6.	pool	2 × 2	16 × 16 × 64
7.			16,384
8.			128
9.			128
10.			2

66.3.2 Weak Convolution Neural Network Classification

A CNN classifier was developed to act as a machine annotator to automatically annotate a larger size of data (Sect. 66.3.3). The classifier was trained on the aforementioned dataset (Sect. 66.3.1). In developing the classifier, a set of random weights were initially used. The weights were then updated by minimizing the cost function with backward propagation. The proposed classifier is considered as a weak CNN, since it is only trained with few hundred images, with noise. As a preliminary experiment, binary classification was conducted: cracks were classified as fatigue cracks or non-fatigue cracks.

The authors assumed that crack patterns are low-level features, and a shallow CNN can better represent the features with few hundred images. There are three convolutional layers (conv), followed by one max-pooling layer (pool) for each convolutional layer (layers 1 to 6). A flatten layer is added later (layer 7). At the end, there are two fully-connected layers (fc) (layers 8 and 9), and one output layer for classification (layer 10), as per Table 66.1. In general, the convolutional layer aims to learn visual features. The pooling layers are used to reduce the spatial dimensions. Finally, after learning 2D features from the convolutional layer, the cubic is flattened to a vector, and connected with the fully-connected layers to do classification learning. The last layer is the soft-max layer to produce a probability distribution for each of the classes.

66.3.3 Image Retrieval and Annotation: Creating the Pseudo Training Data

A larger set of unlabeled images were further retrieved from the Web and classified using the trained CNN classifier (Sect. 66.3.2). The images were retrieved using the Google Image Search and general keywords, such as "cracking", for the queries. The classifier's results are a set of probabilities—for each instance-class pair, the probability that an instance belongs to a class. These probabilities indicate the confidence of the classifier in assigning the labels. A threshold was used to assign the final labels. After retrieval and automatic annotation, this image dataset could serve as the automatically labelled pseudo training data for future research.

66.3.4 Evaluation

The classifier performance was evaluated on natural scene crack images. To develop the ground truth, a set of images were manually labeled into two classes, positive (fatigue crack) and negative (non-fatigue crack). The classification results were evaluated based on recall and precision, with higher importance given to recall because fatigue cracks are critical and should not be missed. Recall and precision are defined in Eqs. (66.1) and (66.2), in which true positive (TP) refers to the number of retrieved images labeled correctly as positive, false positive (FP) refers to the number of retrieved images labeled incorrectly as positive, and false negative (FN) refers to the number of retrieved images labeled incorrectly as negative.

$$Recall = \frac{True\,Positive}{True\,Positive + False\,Negative} \tag{66.1}$$

$$Precision = \frac{True\,Positive}{True\,Positive + False\,Positive} \tag{66.2}$$

66.4 Experimental Results and Discussion

The experiments focused on retrieving and labeling fatigue cracks.

66.4.1 Classifier Training

To train the weak classifier (as per Sect. 66.3.2), the authors downloaded 100 images, including both positive and negative fatigue crack images. The following keywords were used to retrieve the images: “alligator crack” and “fatigue crack” for representing the positive training cases, and “longitudinal crack” and “transverse crack” to represent the negative training cases. After data cleaning, the weak CNN classifier was trained using 153 and 105 positive and negative images, respectively. Examples of the top-two retrieved images are shown in Figs. 66.3 and 66.4.

Fig. 66.3 Top two retrieved images using the “longitudinal crack” and “transverse crack” keywords (further annotated with “non-fatigue crack”)

Fig. 66.4 Top two retrieved images using the “alligator crack” and “fatigue crack” keywords (further annotated with “fatigue crack”)

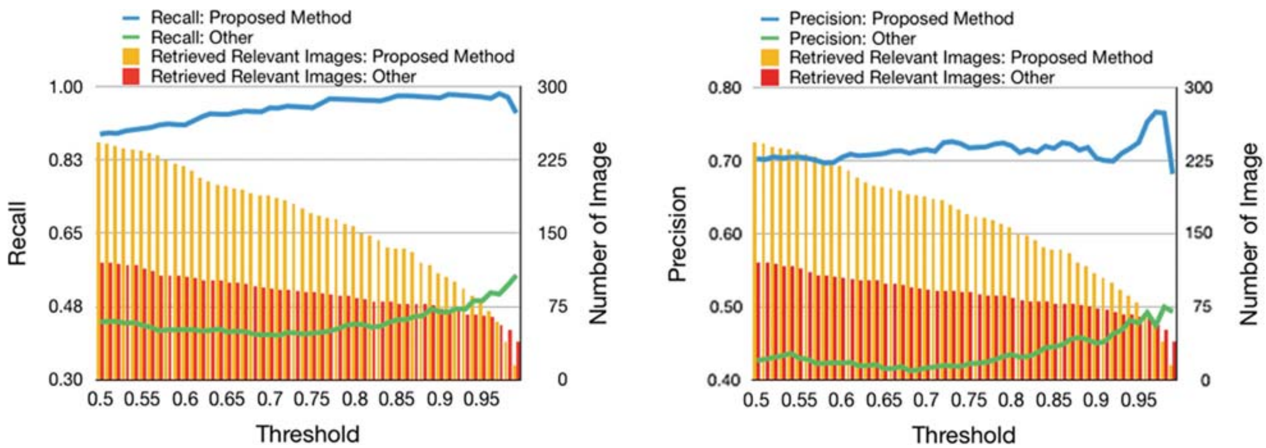

Fig. 66.5 Performance results for the classification

66.4.2 Classifier Evaluation

The proposed data retrieval and automated annotation approach was evaluated in two ways. First, the classifier performance was evaluated using the ground truth. The keyword “pavement crack” was used to retrieve more images from the Web. After data cleaning, 548 images were retrieved and labeled manually for developing the ground truth for testing (as per Sect. 66.3.4). Second, the impact of the dataset on the classification performance was evaluated. An existing dataset was manually labeled and used for retraining the classifier, instead of the authors’ dataset. That dataset contains 132 and 151 positive and negative fatigue crack images, respectively. A set of threshold values were tested.

66.4.3 Performance Results

The proposed method was able to correctly label 188 out of 198 fatigue crack images, with a labeling recall and precision of 95 and 71%, respectively, as shown in Fig. 66.5. On the other hand, the classifier trained on the existing dataset was only able to achieve 41 and 42% recall and precision, respectively, as shown in Fig. 66.5. This indicates that the proposed method can retrieve and automatically annotate natural scene fatigue crack images with much higher recall and precision.

66.4.4 Error Analysis

An error analysis was conducted to further study how to improve the precision results, in future work. Three main sources of errors were identified: (1) some images have other types of cracks (e.g., longitudinal or transverse cracks) in addition to fatigue cracks; (2) the size of the training data may be insufficient; and (3) some images have undefined crack types. Figure 66.6 shows examples of misclassified images (false positives).

Fig. 66.6 Examples of misclassified images (false positives)

66.5 Conclusions and Future Work

In this paper, the authors proposed a data retrieval and annotation method to automatically retrieval and label crack images from the Web. The dataset aims to serve as pseudo training data for supervised machine learning-based crack classification algorithms. A weak CNN classifier first learns from a limited set of Web images, and then acts as a machine annotator and further labels a larger size of data. The proposed method achieved labeling recall and precision of 95 and 71%, respectively.

Several directions are proposed in future work. First, further efforts could be conducted to improve the performance results, especially the precision. The methodology could be improved in terms of data cleaning, feature learning, and model design aspects. Second, compared to datasets for general object classification problems, the developed dataset is relatively small in size. More data can be generated by extending the search queries. Third, and most importantly, automatic civil infrastructure inspection includes multiple types of defect detection and analysis. At this stage, this work only focused on fatigue cracks. In future work, the authors plan to develop a larger dataset, with more defect classes, to better support fully automated inspection of bridges.

References

1. Zhang, L., Yang, F., Daniel Zhang, Y., Zhu, Y.J.: Road crack detection using deep convolutional neural network. In: 2016 IEEE International Conference on Image Processing (ICIP) (2016). https://doi.org/10.1109/icip.2016.7533052
2. Adlinge, S.S., Gupta, A.K.: Pavement deterioration and its causes. Intl. J. Innov. Res. Dev. **2**(4), 437–450 (2013)
3. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Adv. Neural Inf. Process. Syst. (2012). https://doi.org/10.1145/3065386
4. Siebert, S., Teizer, J.: Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system. Autom. Constr. (2014). https://doi.org/10.1016/j.autcon.2014.01.004
5. Grandsaert, P.J.: Integrating pavement crack detection and analysis using autonomous unmanned aerial vehicle imagery. In: Air Force Institute of Technology Wright-Patterson AFB OH Graduate School of Engineering and Management (2015)
6. Zhang, C., Elaksher, A.: An unmanned aerial vehicle based imaging system for 3D measurement of unpaved road surface distresses. Comput Aided Civil Infrastruct. Eng. (2012). https://doi.org/10.1111/j.1467-8667.2011.00727.x
7. Chambon, S., Moliard, J.M.: Automatic road pavement assessment with image processing: review and comparison. Intl. J. of Geophys. (2011). https://doi.org/10.1155/2011/989354
8. Eisenbach, M., Stricker, R., Seichter, D., Amende, K., Debes, K., Sesselmann, M., Ebersbach, D., Stoecert, U., Gross, H.M.: How to get pavement distress detection ready for deep learning? A systematic approach. In: Neural Networks (IJCNN), 2017 International Joint Conference on IEEE (2017). https://doi.org/10.1109/ijcnn.2017.7966101
9. Pavement Interactive: http://www.pavementinteractive.org/
10. Oliveira, H., Correia, P.L.: CrackIT-An image processing toolbox for crack detection and characterization. In: Image Processing (ICIP), 2014 IEEE International Joint Conference on IEEE (2014). https://doi.org/10.1109/icip.2014.7025160
11. Zou, Q., Cao, Y., Li, Q., Mao, Q., Wang, S.: CrackTree: automatic crack detection from pavement images. Pattern Recognit. Lett. (2012). https://doi.org/10.1016/j.patrec.2011.11.004
12. Salman M., Mathavan S., Kamal K., Rahman M.: Pavement crack detection using the Gabor filter. In: Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Annual Conference on IEEE (2013). https://doi.org/10.1109/itsc.2013.6728529
13. Varadharajan, S., Jose, S., Sharma, K., Wander, L., Mertz, C.: Vision for road inspection. In: Applications of Computer Vision (WACV), 2014 IEEE Winter Conference on IEEE (2014). https://doi.org/10.1109/wacv.2014.6836111
14. Mertz, C.: Continuous road damage detection using regular service vehicles. In: Proceedings of the ITS world congress (2011)
15. Yamada, T., Ito, T., Ohya, A.: Detection of road surface damage using mobile robot equipped with 2d laser scanner. In: System Integration (SII), 2013 IEEE/SICE International Symposium on IEEE (2013). https://doi.org/10.1109/sii.2013.6776679

16. Yu, Y., Guan, H., Ji, Z.: Automated detection of urban road manhole covers using mobile laser scanning data. IEEE Trans. Intell. Transp. Syst. (2015). https://doi.org/10.1109/tits.2015.2413812
17. Byoung, J.L., Hosin, D.L.: A robust position invariant artificial neural network for digital pavement crack analysis. In: TRB 2003 Annual Meeting (2002)
18. Saar T., Talvik O.: Automatic asphalt pavement crack detection and classification using neural networks. In: Electronics Conf. (BEC), 2010 12th Biennial on IEEE (2010). https://doi.org/10.1109/bec.2010.5630750
19. Radopoulou, S.C., Brilakis, I.: Automated detection of multiple pavement defects. J. Comput. Civil Eng. (2016). https://doi.org/10.1061/(asce)cp.1943-5487.0000623
20. Hoang, N.D., Nguyen, Q.L.: A novel method for asphalt pavement crack classification based on image processing and machine learning. Eng. Comput. (2018). https://doi.org/10.1007/s00366-018-0611-9
21. Washington State Department of Transportation. Pavement Surface Condition Rating Manual. Olympia, WA (1992)
22. American Association of State Highway and Transportation Officials. AASHTO Guide for Design of Pavement Structures. Washington D.C (1993)
23. Miller, J.S., Bellinger, W.Y.: Distress identification manual for the long-term pavement performance program. United States Federal Highway Administration. Office of Infrastructure Research and Development (2014)
24. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature. **521**(7553), 436 (2015)
25. Roth, H.R., Lu, L., Liu, J., Yao, J., Seff, A., Cherry, K., Kim, L., Summers, R.M.: Improving computer-aided detection using convolutional neural networks and random view aggregation. IEEE Trans. Med. Imaging (2016). https://doi.org/10.1109/tmi.2015.2482920
26. Ciresan, D., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: Advances in Neural Information Processing Systems, pp. 2843–2851 (2012)
27. Cirean, D.C., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Mitosis detection in breast cancer histology images with deep neural networks. In: International Conference on Medical Image Computing and Computer-assisted Intervention, Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40763-5_51
28. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images
29. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes (voc) challenge. Intl. J. Comput Vision (2009). https://doi.org/10.1007/s11263-009-0275-4
30. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on IEEE (2009). https://doi.org/10.1109/cvprw.2009.5206848
31. Li, L.J., Fei-Fei, L.: Optimol: automatic online picture collection via incremental model learning. Intl. Comput. Vision (2010). https://doi.org/10.1109/cvpr.2007.383048
32. Schroff, F., Criminisi, A., Zisserman, A.: Harvesting image databases from the web. IEEE Trans. Pattern Anal. Mach. Intell. (2011). https://doi.org/10.1109/iccv.2007.4409099
33. Yao, Y., Zhang, J., Shen, F., Hua, X., Xu, J., Tang, Z.: Automatic image dataset construction with multiple textual metadata. In: Multimedia and Expo (ICME), 2016 IEEE International Conference on IEEE (2016). https://doi.org/10.1109/tits.2016.2552248
34. Shi, Y., Cui, L., Qi, Z., Meng, F., Chen, Z.: Automatic road crack detection using random structured forests. In: IEEE Transactions on Intelligent Transportation Systems (2016)
35. Amhaz, R., Chambon, S., Idier, J., Baltazart, V.: A new minimal path selection algorithm for automatic crack detection on pavement images. In: 2014 IEEE International Conference on Image Processing (ICIP) (2014). https://doi.org/10.1109/icip.2014.7025158