Crack recognition automation in concrete bridges using Deep Convolutional Neural Networks

Hajar Zoubir^{1}, Mustapha Rguig¹, and Mohammed Elaroussi¹*

¹Centre des Etudes Doctorales de l'Ecole Hassania des Travaux Publics, Casablanca, Morocco

Keywords: Visual inspection, Bridge concrete crack recognition, Dataset, Deep Convolutional Neural Network

Abstract. Using Unmanned Aerial Systems (UASs) for bridge visual inspection automation necessitates the implementation of Deep Convolutional Neural Networks (DCNNs) to process efficiently the large amount of data collected by the UASs sensors. However, these networks require massive training datasets for the defects recognition and detection tasks. In an effort to expand existing concrete defects datasets, particularly concrete cracks in bridges, this paper proposes a public benchmark annotated image dataset containing over 6900 images of cracked and non cracked concrete bridges and culverts. The presented dataset includes some challenging surface conditions and covers concrete cracks with different sizes and patterns. The authors analyzed the proposed dataset using three state of the art DCNNs in Transfer Learning mode. The three models were used to classify the cracked and non cracked images and the best testing accuracy obtained reached 95.89%. The experimental results showcase the potential use of this dataset to train deep networks for concrete crack recognition in bridges. The dataset is publicly available at <https://github.com/MCBDD-ZRE/Concrete-Bridge-Crack-Dataset-> for academic purposes.

1 Introduction

Visual inspection is the prevalent and the common practice for bridge condition assessment [1– 3]. It is conducted, regularly, by trained inspectors who record the extent and the severity level of defects on bridge elements (e.g. cracks, concrete spalling, reinforcing steel corrosion and efflorescence) and update the last inspection data related to the structure [4]. Particularly, cracks are common defects encountered in bridges [5, 6] and their presence in reinforced concrete elements provides easy access to aggressive agents that initiate the

*Corresponding author: zoubirhajar@gmail.com

reinforcement corrosion [7]. Hence, they are largely considered in the concrete bridges condition assessment [4].

To conduct visual examinations, the use of access equipments or vehicles is often requisite to gain close range access and examine hard-to-reach areas, which is incurring additional indirect costs [2]. Currently, UASs (Unmanned Aerial Systems) have gained significant interest in bridge inspection application as an assistive, efficient, and cost-effective tool offering great potential for inspection automation [2, 8]. However, data interpretation for damage assessment might be tedious and time consuming given the large amount of images collected by the UAS sensors (i.e. cameras) [7]. To overcome this challenge, deploying vision based techniques can potentially automate the assessment process.

Vision-based techniques have been widely utilized to automate crack detection in concrete surfaces. Koch et al. [9] provided a synthesis of several research studies related to crack detection as well as crack properties extraction in reinforced concrete bridges using computer vision methods. Particularly, Deep Learning Convolutional Neural Networks (DLCNN) trained on datasets help to address the limitations in crack detection using conventional image processing techniques, related to the complexity of concrete surface conditions [7]. Dorafshan et al. [10] presented a comparison of Deep Convolutional Neural Networks (DCNNs) and edge detectors for image-based crack detection in concrete. Their study showed that the AlexNet Network [11] offered great performance compared to the traditional edge detectors for concrete crack detection. However, these networks require massive training datasets to increase their performance in the defect detection task and available datasets for concrete defects, particularly in bridges, are limited [12].

In this context, we present in this paper an annotated image dataset containing over 6900 cracked and non cracked images of concrete bridges and culverts. The authors conducted some experiments on the proposed dataset to demonstrate its potential use to train algorithms for concrete crack recognition.

This paper is subdivided into five sections including this section. The second section presents state-of-the art datasets released by the research community for bridge concrete defects detection with emphasis on concrete cracks. The third details the experimental program for data acquisition, the properties of the proposed dataset and the detailed approach followed to train three DCNNs (i.e. VGG16, VGG19[13], InceptionV3[14]) on our dataset, while the fourth presents and discusses the results of the experiments. The final section provides conclusions and perspectives regarding future work.

2 State of the art datasets for concrete bridge defects detection

For bridge damage detection and given the diversity of defects, their possible representative features, and access difficulties to all bridge components, the construction of a bridge concrete defects dataset is expensive, time consuming and requires the empirical knowledge of experts.

CSSC database [15] contains images of concrete spalling and crack. It was built based on web search and real data collection. More than 1200 images were manually labeled, rotated and randomly sampled resulting in 15,950 sub-images of concrete spalling and 31,180 sub-images for concrete crack images of different sizes.

SDNET2018 [16] is a public benchmark dataset containing annotated images of cracked and non-cracked concrete bridge decks, walls, and pavements. To build SDNET2018, 230 images (54 bridge decks, 72 walls, 104 pavements) were subdivided into more than 56,000 sub-images of 256 x 256 px.

CODEBRIM dataset [12] consists of bridge images of five overlapping concrete defect classes (i.e. crack, spallation, exposed reinforcement bar, efflorescence, and corrosion) and

non defective background images. 1590 high resolution images with defects of 30 bridges were taken with multiple cameras and a subset of the CODEBRIM dataset was gathered by UAV (Unmanned Aerial Vehicle). 5354 annotated defect bounding boxes (with overlapping defects) and 2506 non overlapping background bounding boxes resulted from the annotation process.

Xu et al.[17] shared a dataset consisting of 6069 cracked and non cracked concrete images of 224 x 224 px. The dataset was built based on an artificial augmentation of the bridge crack dataset [18].

Although considerable efforts have been devoted to build and share concrete bridge defects datasets, more contributions are needed to create a larger and a more comprehensive dataset that covers the diverse representations of concrete defects encountered in bridges. We present in the following section a dataset of bridge concrete crack as a contribution to expand and supplement the existing related datasets.

3 Experimental program

3.1 Data acquisition and preparation

The dataset presented in this paper contains cracked and non-cracked images of concrete bridges and culverts of the Moroccan road network.

572 images of concrete decks and piers with cracks were collected at different surface and lighting conditions (e.g. roughness, color, wetness and strong light) to improve the diversity of the proposed dataset. The original images have a 5152 x 3864 px resolution.

The 572 images were manually cropped using the inbac tool (<https://github.com/weclaw1/inbac>) to define accurately the flaw area. This operation is tedious, time consuming and requires the judgment of an expert. It is noteworthy that the original images have not undergone any modification (e.g. by applying pre-processing or data augmentation techniques) other than the aforementioned cropping operation.

Figure 1 presents the process followed to create the proposed dataset.

Fig.1. Process followed to create the proposed dataset

3.2 Dataset properties

The proposed dataset consists of 1304 cracked images and 5634 non-cracked images. The images are 200 x 200-px RGB image files in .jpg format. The dataset is publicly available at: <https://github.com/MCBDD-ZRE/Concrete-Bridge-Crack-Dataset> for any academic purposes. The cracked images cover cracks with different patterns and sizes. The dataset includes some challenging surface alterations as well (e.g. stains and markings). Particularly, some non-cracked images contain concrete joints, which are likely to be mistaken for cracks while performing the classification task, and some minor defects (e.g. scaling and segregation).

Figure 2 shows some representative sample images of the proposed dataset.

Fig.2. Sample images of the proposed dataset: (a) and (b) Different concrete colors; (c) Background with markings; (d) Concrete joints; (e) Concrete segregation; (f) and (g) Concrete scaling; (h)-(m) Cracks with different widths and orientations; (n) and (o) Map cracks.

3.3 Experimentation on the proposed dataset using DCNNs

The models used in this paper (i.e. VGG16, VGG19 [13] and InceptionV3[14]) were proposed in the ImageNet Large Scale Visual Recognition Challenge in 2014 and 2015. With their deep architectures, these classic networks have shown great performance in the image classification task. Figure 3 presents the example of the VGG16 model architecture. The network contains a total of 16 weight layers including 13 convolutional layers that extract features from input images through the convolution operation and 3 fully connected layers that combine the extracted features to output the class using the Softmax activation .The ReLU activation introduces non linearity in the model and the max pooling layers are used to reduce the number of parameters to learn.

Fig.3. Architecture of the VGG16 model

Our dataset was randomly split into 3 parts: training (70%), validation (10%) and testing (20%) subsets. Concrete crack recognition using the three networks was then performed by classification of our dataset images in Transfer Learning (TL) mode. TL approach was applied in [7, 15, 19, 20] to address the limitations related to the limited size of the available concrete crack datasets and the significant time required for training the used models from scratch.

In this work, pretrained weights on the ImageNet Dataset [21] were used and the convolutional layers of the three abovementioned networks models were kept non-trainable. Only the fully connected layers were trained on the proposed dataset.

Several research studies related to concrete crack image classification (e.g. [10, 15, 22]) have deployed advanced computational resources and trained the used models using Graphics Processing Units (GPUs). All the experiments in this paper were conducted in Google Colaboratory (Colab) with the 12GB NVIDIA Tesla K80 GPU provided by the platform. Colab is a cloud platform based on Jupyter Notebooks that provide several machine learning libraries [23]. The three models were implemented using the TensorFlow and Keras libraries in Python language.

Table 1 displays the number of images in the training, validation and testing sets and the number of epochs used to train each network.

Table 1. Number of training, validation and testing images and number of training epochs.

Model	No. training images	No. validation images	No. testing images	Training epochs	Steps per epoch	Validation Steps
VGG16	4855	693	1390	10	10	5
VGG19	4855	693	1390	10	10	5
InceptionV3	4855	693	1390	10	10	5

To evaluate the performance of the models on the proposed dataset, the accuracy metric was considered in this paper. It is defined as follows:

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN}) \quad (1)$$

where:

- TP (True Positives) : refer to the number of cracked images that are correctly classified as cracked
- TN (True Negatives) : refer to the number of non-cracked images that are correctly classified as non-cracked
- FP (False Positives): refer to the number of non-cracked images that are incorrectly identified as cracked.
- FN (False Negatives) : refer to the number of cracked images that are incorrectly identified as non-cracked

4 Results and discussions

In this section we present the results of training the DCNNs presented in section 3 on the proposed dataset in Transfer Learning mode.

4.1 Training and validation

Figure 4 presents corresponding training and validation accuracies during the training of the three DCNNs considered in this paper.

The learning curves show that the models present a good performance on the training dataset and on the validation dataset as well.

Table 2 shows the highest training and validation accuracies achieved by the three models, their corresponding epochs and the longest training time recorded for one epoch.

It can be seen that the three models trained on our dataset in TL mode achieved high training and validation accuracies (e.g. a 96.75% training accuracy and a 97.50% validation accuracy achieved by the VGG16 model).

Fig. 4. Training and validation accuracies of the 3 DCNNs trained on our Dataset in Transfer Learning mode.

Table 2. The highest training and validation accuracies achieved by the three models, their corresponding epochs and the longest training time recorded for one epoch.

Model	Training accuracy (%)	Epoch	Validation accuracy (%)	Epoch	Time (s)
VGG16	96.75	9	97.50	7	424
VGG19	96.17	10	96.25	10	252
InceptionV3	96.96	10	96.88	5	390

4.2 Testing results

Table 3 presents the classification accuracy of the testing dataset using the 3 DCNNs considered in this paper.

It can be seen that the models achieved a high accuracy in the classification of the testing dataset images (95.89% using InceptionV3 architecture in TL mode).

Experimental results show that the trained models in TL mode (with all the convolutional layers being frozen) achieved high learning, validation and testing accuracies. This means that the two defined classes are separable and recognizable by the models considered in this paper.

It is noteworthy that the recognition accuracy can be increased by applying some pre-processing techniques to reduce some of the challenges in the proposed dataset (e.g. image blurriness and strong light).

Table 3. Testing accuracies

Model	Training epochs	Testing accuracy (%)
VGG16	10	94.89
VGG19	10	95.39
InceptionV3	10	95.89

5 Conclusions and future work

The paper proposed a benchmark image annotated dataset containing over 6900 cracked and non cracked images of concrete bridges and culverts. Three state-of-the art DCNN models were trained on the dataset in Transfer Learning mode and achieved high learning, validation and testing accuracies (95.68% testing accuracy achieved by InceptionV3). The experimental results showed that the two defined classes are separable and recognizable by the trained models.

Leveraging all the available bridge concrete crack datasets would provide a more comprehensive representation of cracks encountered in deteriorating concrete bridges and would, as a result, enhance the robustness of the trained algorithms for the bridge concrete crack recognition task.

Since cracks are not the only defects affecting concrete bridges, datasets with more concrete deficiencies classes are needed to provide a diverse coverage of the concrete defects and perform a multi-target classification with designed models tailored for this purpose. We are currently working on building datasets related to two other common concrete defect classes (i.e. concrete spalling with rebar corrosion and concrete efflorescence).

Implementing trained models on large defects datasets on UASs offers a great potential in automating the concrete bridge inspection process and would present a powerful tool for a more efficient bridge condition assessment.

References

1. T. Omar and M. Nehdi, "Condition Assessment of Reinforced Concrete Bridges: Current Practice and Research Challenges," *Infrastructures*, vol. **3**, no. 3, p. 36 (2018)
2. S. Dorafshan and M. Maguire, "Bridge inspection: human performance, unmanned aerial systems and automation," *J Civil Struct Health Monit*, vol. **8**, no. 3, pp. 443–476 (2018)
3. M. Alsharqawi, T. Zayed, and S. Abu Dabous, "Integrated condition rating and forecasting method for bridge decks using Visual Inspection and Ground Penetrating Radar," *Automation in Construction*, vol. **89**, pp. 135–145 (2018)
4. R. S. Adhikari, O. Moselhi, and A. Bagchi, "Image-based retrieval of concrete crack properties for bridge inspection," *Automation in Construction*, vol. **39**, pp. 180–194 (2014)
5. C. M. Yeum and S. J. Dyke, "Vision-Based Automated Crack Detection for Bridge Inspection: Vision-based automated crack detection for bridge inspection," *Computer-Aided Civil and Infrastructure Engineering*, vol. **30**, no. 10, pp. 759–770 (2015)
6. A. M. Alani, M. Aboutaleb, and G. Kilic, "Integrated health assessment strategy using NDT for reinforced concrete bridges," *NDT & E International*, vol. **61**, pp. 80–94 (2014)

7. W. R. L. da Silva and D. S. de Lucena, "Concrete Cracks Detection Based on Deep Learning Image Classification," *Proceedings*, vol. **2**, no. 8, p. 489, (2018)
8. J. Seo, "Drone-enabled bridge inspection methodology and application," *Automation in Construction*, p. 15 (2018)
9. C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. **29**, no. 2, pp. 196–210 (2015)
10. S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," *Construction and Building Materials*, vol. **186**, pp. 1031–1045 (2018)
11. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, vol. **25** (2012)
12. M. Mundt, S. Majumder, S. Murali, P. Panetsos, and V. Ramesh, "Meta-Learning Convolutional Neural Architectures for Multi-Target Concrete Defect Classification With the CONcrete DEfect BRidge IMage Dataset," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, pp. 11188–11197 (2019)
13. K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *International Conference on Learning Representations (ICLR)* (2015)
14. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826 (2016)
15. L. Yang, B. Li, W. Li, Z. Liu, G. Yang, and J. Xiao, "Deep concrete inspection using unmanned aerial vehicle towards cssc database," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems* pp. 24–28 (2017)
16. S. Dorafshan, R. J. Thomas, and M. Maguire, "SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks," *Data in Brief*, vol. **21**, pp. 1664–1668 (2018)
17. H. Xu, X. Su, Y. Wang, H. Cai, K. Cui, and X. Chen, "Automatic Bridge Crack Detection Using a Convolutional Neural Network," *Applied Sciences*, vol. **9**, no. 14, p. 2867 (2019)
18. L.-F. Li, W.-F. Ma, L. Li, and C. Lu, "Research on detection algorithm for bridge cracks based on deep learning," *Acta Automatica Sinica* (2019).
19. Q. Yang, W. Shi, J. Chen, and W. Lin, "Deep convolution neural network-based transfer learning method for civil infrastructure crack detection," *Automation in Construction*, vol. **116**, p. 103199 (2020)
20. S. Dorafshan, R. J. Thomas, C. Coopmans, and M. Maguire, "Deep Learning Neural Networks for sUAS-Assisted Structural Inspections: Feasibility and Application," in *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, Dallas, TX, pp. 874–882 (2018)
21. J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database" in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255 (2009)
22. S. Drouyer, "An 'All Terrain' Crack Detector Obtained by Deep Learning on Available Databases," *Image Processing On Line*, vol. **10**, pp. 105–123 (2020)
23. T. Carneiro, R. V. Medeiros Da Nóbrega, T. Nepomuceno, G.-B. Bian, V. H. C. De Albuquerque, and P. P. R. Filho, "Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications," *IEEE Access*, vol. **6**, pp. 61677–61685 (2018)