Advances in applicable deep-learning based defect detection

M. Żarski [1,2][1], B. Wójcik [1,2], K. Książek [2], M. Salamak [1], J. A. Miszczak [2]

[1] *Silesian University of Technology, Faculty of Civil Engineering, Akademicka 5, 44-100 Gliwice, Poland,*
[2] *Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Bałtycka 5, 44-100 Gliwice, Poland.*

1. Introduction

Early detection of infrastructure defects like cracks of small width can lead to lowering the overall costs of infrastructure through the incorporation of preventive maintenance [1,2]. However, such defects are often omitted during manual inspections – in fact over 60% of small defects are never included in Infrastructure Inspectors reports [3–5], which causes invalid scores of the facilities, and thus invalid future infrastructure maintenance plans. While there already exists methods for automatic defect detection with the use of deep learning methods [6–8], it can be argued that they are not applicable in the reality of infrastructure maintenance. They require vast computing power for training and inference that excludes the use of edge or mobile devices, but still, even with vast resources are limited to low resolution images [9].

In this work we present you our path that led us to reaching practical applicability of deep learning algorithms for infrastructure defect detection and placed us in the fields' state-of-the-art territory. Throughout our studies we developed KrakN – an end-to-end, open source and scalable framework for transfer learned deep learning model development intended for minimizing the training time and allowing the use of a single backbone CNN for multiple defect classifiers. Then we were among the first who tackled the problem of pruning transfer learned CNN models with our Finicky Transfer Learning (FTL) method, which allowed us for pruning as much as 95% of CNNs' parameters while maintaining its initial accuracy. We also present our further plans for refining our work in order to push the field even further and match the accuracy of the computationally heaviest solutions while maintaining high applicability of our methods.

2. Defect detection with transfer learning

The main goal of our research was to enable the practical use of deep learning methods for infrastructure defect detection. In order to do so, we identified the basic problems that prevent the use of already existing methods in practice – the need of vast computing power for training the model, lack of scalability and versatility, and the need of expert knowledge in the field of deep learning to adapt the solution to specific needs.

1 Corresponding author: mateusz.zarski@polsl.pl

The KrakN [10] framework created by us is the answer to these problems by offering end-to-end methods of deep learning model development suited for users without background in data science. Its principle of operation is shown in Figure 1. KrakN uses a single CNN as a backbone feature extractor for multiple defect classifiers, and offers tools for semi-automatic development of datasets. The tests we have carried out have shown that KrakN is characterized by a greater ability to generalize knowledge than analogous solutions that do not use transfer learning, when target datasets differ from the training one, *e.g.* in the appearance of a concrete surface. It scored above 90% in accuracy of detecting cracks with width under 0,2mm.

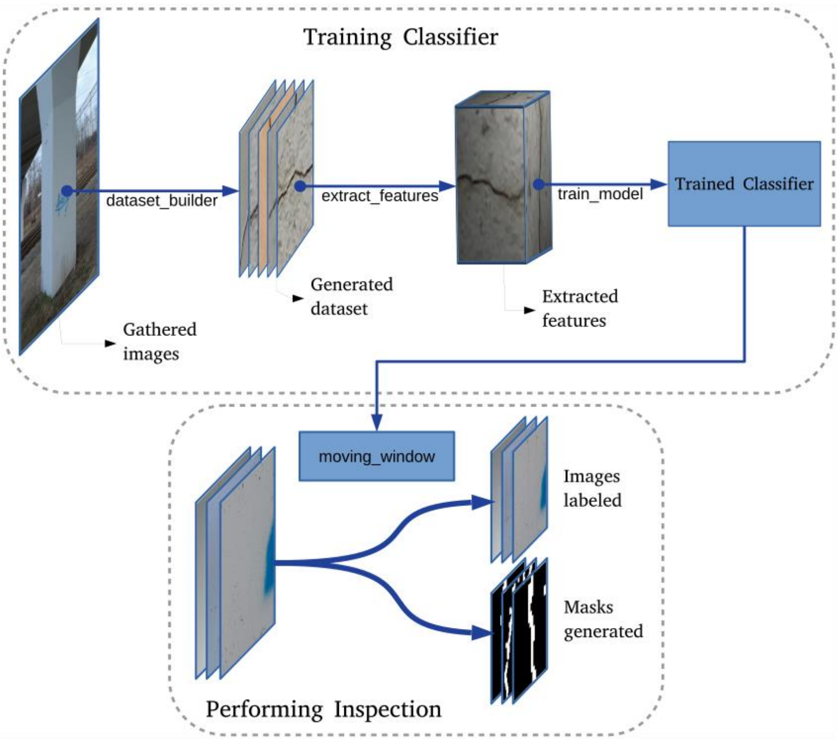

Figure 1. *Workflow of KrakN framework*

KrakN uses sliding window method for object detection in order to be able to detect defects regardless of the image size – it can process large format orthomosaics obtained with photogrammetric reconstruction unlike methods based on single shot detectors which are limited by the size of input image. It also comes as an open source solution and can be easily modified. An example of such a modification is the add-on module prepared by us, which allows for multi-classifier approach presented in earlier works [11], but was never made available publicly.

KrakN was also our initial step to further developing methods based on transfer learning and adapting them for use on edge devices.

3. Transfer learning with pruning

An extension of our methods using transfer learning is the Finicky Transfer Learning [12] – CNN structural pruning method that allows for limiting the demand for computing power by reducing the number of algorithm parameters.

It uses the Jaccard similarity coefficient (**IoU**) for assessing adaptation of subsequent CNN filters trained on the external data set to the new type of the searched object. It is assessed to what extent convolutional filters are able to extract features from multiple images, within the damage area. Then, without retraining the CNN, only its layers with highest mean IoU score are used as feature extractor for the new classifier. The method of obtaining mean IOU score is described with equation 1, where X^{seg} is the set of all images with defect segmented out, $f^i_j(x_k)$ is image segmented with the *j-th* filter of the *i-th* network layer, x^s_k is image segmented manually, and *m* is the total number of images in dataset.

$$\overline{IoU}\left(X^{seg}, f_j^i\right) = \frac{1}{m}\sum_{k=1}^{m}\left(\frac{\left|x_k^s \cap f_j^i(x_k)\right|}{\left|x_k^s \cup f_j^i(x_k)\right|}\right) \tag{1}$$

With the use of Finicky Transfer Learning we were able to reduce the total number of CNN parameters by up to 95% while maintaining its initial accuracy. It enabled the effective use of single board edge devices like Raspberry Pi computers. The comparison of image processing time with CNN pruned with FTL method compared to unpruned network running inference on high-performance, 6 core CPU is shown in Figure 2.

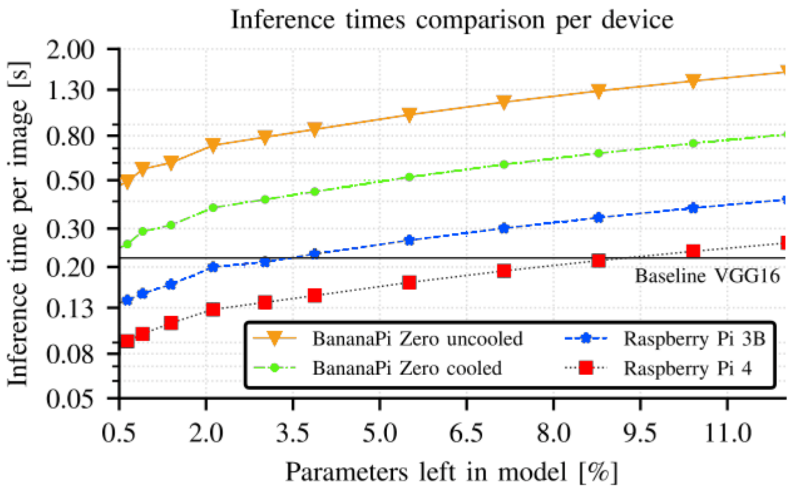

***Figure 2**. Comparison of pruned CNN inference times on edge devices*

It can be seen that single board edge devices can run inference with CNN models pruned with FTL method with performance matching high-performance CPU. Moreover, when considering that FLT uses transfer learning, the classifierpart of the CNN is also trained much faster than with the use of full, unpruned model.

4. Future goals

The experience we have gained while working on the KrakN framework and the FTL method has led us to start working on a method that uses scructual CNN pruning during the training of the model. We believe that by using our pruning method during fine tuning of the model, we will be able to raise the final model accuracy metrics above the initial values of the full, unpruned CNN, while greatly reducing its number of parameters.

We are also considering modyfing the pruned CNN with the utilization of various micro architectures. By doing so we'll be able to enrich the final feature tensor with features extracted by the initial layers of the model and thus increase the number of image characteristics for the classifier.

Acknowledgements

Work presented in this study was supported by the *European Union* through the *European Social Fund* as a part of a *Silesian University of Technology as a Centre of Modern Education based on research and innovation* project, number of grant agreement: **POWR.03.05.00 00.z098/17-00** (BW and MŻ).

References

Any references should be placed at the end of the paper and referenced in the paper through the number in square brackets. The format of a sample book [1] and journal [2] reference is given below.

[1] **H. Singh**, Repair and retrofitting of bridges – present and future, in: Proc. 9th Int. Conf. Bridg. Maintenance, Saf. (IABMAS 2018), 2018: pp. 1209–1215.

[2] **C.S. Shim, N.S. Dang, S. Lon, C.H. Jeon**, Development of a bridge maintenance system for prestressed concrete bridges using 3D digital twin model, Struct. Infrastruct. Eng. 15 (2019) 1319–1332. https://doi.org/10.1080/15732479.2019.1620789.

[3] **B.A. Graybeal, B.M. Phares, D.D. Rolander, M. Moore, G. Washer**, Visual inspection of highway bridges, J. Nondestruct. Eval. 21 (2002) 67–83. https://doi.org/10.1023/A:1022508121821.

[4] **B.M. Phares, B.A. Graybeal, D.D. Rolander, M.E. Moore, G.A. Washer**, Reliability and accuracy of routine inspection of highway bridges, Transp. Res. Rec. (2001) 82–92. https://doi.org/10.3141/1749-13.

[5] **B.M. Phares, G.A. Washer, D.D. Rolander, B.A. Graybeal, M. Moore**, Routine Highway Bridge Inspection Condition Documentation Accuracy and Reliability, J. Bridg. Eng. 9 (2004) 403–413. https://doi.org/10.1061/(ASCE)1084-0702(2004)9:4(403).

[6] **Y. Liu, J. Yao, X. Lu, R. Xie, L. Li**, DeepCrack: A deep hierarchical feature learning architecture for crack segmentation, Neurocomputing. 338 (2019) 139–153. https://doi.org/10.1016/j.neucom.2019.01.036.

[7] **B. Kim, S. Cho**, Automated Vision-Based Detection of Cracks on Concrete Surfaces Using a Deep Learning Technique, Sensors. 18 (2018). https://doi.org/10.3390/s18103452.

[8] **S. Li, X. Zhao, G. Zhou**, Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network, Comput. Civ. Infrastruct. Eng. 34 (2019) 616–634. https://doi.org/10.1111/mice.12433.

[9] **B. Wójcik, M. Żarski**, Asesment of state-of-the-art methods for bridge inspection: case study, Arch. Civ. Eng. 66 (2020) s. 343-362. https://doi.org/10.24425/ace.2020.135225.

[10] **M. Żarski, B. Wójcik, J.A. Miszczak**, KrakN: Transfer Learning framework for thin crack detection in infrastructure maintenance, (2020) 1–23. http://arxiv.org/abs/2004.12337.

[11] **P. Hüthwohl, R. Lu, I. Brilakis**, Multi-classifier for reinforced concrete bridge defects, Autom. Constr. 105 (2019) 102824. https://doi.org/10.1016/j.autcon.2019.04.019.

[12] **M. Żarski, B. Wójcik, K. Książek, J.A. Miszczak**, Finicky Transfer Learning -- a method of pruning convolutional neural networks for cracks classification on edge devices, Comput. Civ. Infrastruct. Eng. (2021). https://doi.org/10.1111/mice.12755.