A convolutional neural network for pavement surface crack segmentation using residual connections and attention gating

Koenig, Jacob; Jenkins, Mark David; Barrie, Peter; Mannion, Mike; Morison, Gordon

Published in:
2019 IEEE International Conference on Image Processing (ICIP)

DOI:
10.1109/ICIP.2019.8803060

Publication date:
2019

Document Version
Author accepted manuscript

Link to publication in ResearchOnline

Citation for published version (Harvard):
Koenig, J, Jenkins, MD, Barrie, P, Mannion, M & Morison, G 2019, A convolutional neural network for pavement surface crack segmentation using residual connections and attention gating. in *2019 IEEE International Conference on Image Processing (ICIP).* IEEE, pp. 1460-1464, 2019 IEEE International Conference on Image Processing, Taipei, Taiwan, 22/09/19. https://doi.org/10.1109/ICIP.2019.8803060

A CONVOLUTIONAL NEURAL NETWORK FOR PAVEMENT SURFACE CRACK SEGMENTATION USING RESIDUAL CONNECTIONS AND ATTENTION GATING

Jacob König[⋆†] *Mark David Jenkins*[†] *Peter Barrie*[⋆] *Mike Mannion*[⋆] *Gordon Morison*[⋆]

[⋆]Glasgow Caledonian University, Glasgow, United Kingdom
[†]Geckotech Solutions Ltd, Edinburgh, United Kingdom
gordon.morison@gcu.ac.uk

ABSTRACT

Conventional surface crack segmentation requires images manually labelled by a trained expert. It is a challenging task as cracks can vary in orientation and size, with some parts of cracks only being one pixel wide. Further, available training data for crack segmentation is sparse. In this work we propose to automate this annotation task, by introducing a fully convolutional U-Net based architecture for semantic segmentation of surface cracks which allows for the use of small datasets through a patch based training process. Our proposed configuration makes use of residual connections inside the convolutional blocks as well as including an attention based gating mechanism between the encoder and decoder section of this architecture, which only propagates relevant activations further. Using our proposed architecture we achieve new state of the art results in two different crack datasets, outperforming the previous best results in two metrics each.

***Index Terms*—** Semantic Segmentation, Attention, Residual Connections, U-Net, Surface Cracks

1. INTRODUCTION

Cracks are a common defect which can appear on horizontal and vertical surfaces, such as roads and walls. They develop through consistent use of the surface or with age and can impact the structural integrity. Therefore, surface monitoring is an important aspect in maintaining such structures. In recent years, many different tasks were able to be automated through the use of deep learning algorithms. Semantic segmentation is the task of assigning a class label onto each pixel of an image. Conducting this crack segmentation task manually, to analyse images at a later stage, is time and resource intensive as some cracks may only one pixel wide.

Several state of the art results in semantic segmentation are achieved using fully convolutional neural networks [1, 2, 3], with a popular area of research being segmentation of medical images [3, 4, 5]. Medical imaging shows many similarities to crack segmentation as various sized objects, often containing important small sized details, are being segmented. Therefore many of these architectures can be used for automatic crack segmentation, which will improve time and cost efficiency as well as improving the safety of the inspector conducting carrying out the manual visual inspection.

The shape of structural cracks varies as they can be interconnected as well as expand into multiple directions [6]. Initial attempts for automated crack detection were based on traditional methods such as morphology [7] or edge detection algorithms [8] whereas further research used machine learning methods such as random structured forests [9] or support vector machines [10]. However, images containing cracks can differ in quality and include artefacts such as shadows, spills or objects such as leafs and cracks may not be of a homogeneous consistency. Due to this, conventional algorithms tend not to perform well. These deep learning based methods are better at adapting to these anomalies, as they do not require handcrafted features which these previous methods rely on [11]. It is shown that crack detection methods using deep learning generally outperform other non deep learning based methods [12, 13, 14]. Whilst much research has been conducted into the field of crack detection, the number of available datasets is still limited and no general consensus on a specific dataset for benchmarking exists.

A popular deep learning based architecture for semantic segmentation is U-Net [3]. This architecture features an encoder as well as a decoder section who are connected through shortcut connections. It has successfully been applied for semantic segmentation on medical images [3, 4] as well as crack segmentation [15]. To improve performance of the base U-Net architecture several new components have been introduced, such as residual connections between convolutions [16, 4] as well as making use of attention to gate the shortcut connections [5].

In this work we propose an encoder-decoder U-Net based architecture for semantic segmentation of cracks. This architecture utilises attention gating for propagation of only relevant features between the encoder and decoding section. Further, we also employ residual connections, inside each encoder and decoder block, for improved performance. To our best knowledge this is the first application which utilises

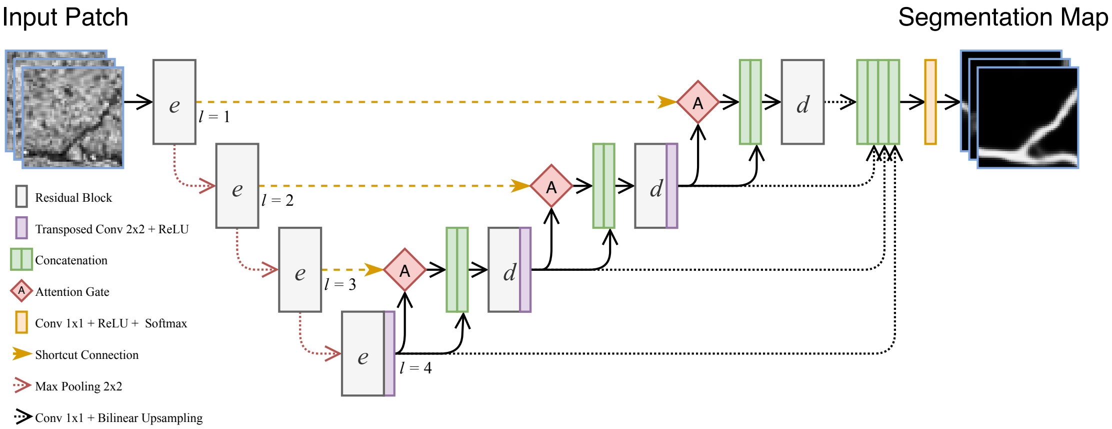

Fig. 1. The proposed U-Net architecture. l denotes the layer, e and d denote encoder and decoder blocks respectively.

both of these components in an U-Net based architecture for semantics segmentation. Through use of this architecture and a patch based training process we achieve new state of the art results in two surface crack datasets: CFD [9] and AigleRN, [17] outperforming the previous in two metrics in each dataset.

2. APPROACH

2.1. Network Architecture

The network architecture used in this work is based on U-Net [3]. The original U-Net architecture was designed for segmentation of microscopic cells with limited data available. This correlates highly with the task of crack segmentation due to limited training data and segmentation of small thin shaped objects. Therefore this architecture poses to be ideally suited for this work.

U-Net based architectures can vary in depth, however this work makes use of a four level architecture. At each level l of this architecture we employ encoding convolutional blocks e^l. They are connected through pooling operations, each downsampling the spatial dimensions of the feature map by a factor of two. Each level is further associated with a number of filters for the convolutional operations in the encoder and decoder blocks. Our architecture uses the following number of filters for each level l: $[16_{l=1}, 32_{l=2}, 64_{l=3}, 128_{l=4}]$. Further, opposite to the encoder blocks at level 1-3 there are decoder blocks d^l. The input to these decoder blocks consists of a concatenation of the activations along the channel dimension of the attention gated opposing encoder block, $\hat{x}^l$, as well as the upscaled activations of the previous convolutional block u^l. The upsampling operation in this architecture is a transposed convolution using a filter size of 2×2 and a stride of 2, therefore upscaling by a factor two. The input to this upsampling operation is either the last encoder block if $l = 4$ or the previous decoder block.

Deep supervision [18] allows filters to learn robust features at different scales of the network. We employ this by passing the output of each decoder block through a 1×1 linear transformation followed by bilinear upsampling to the spatial size of the input. These feature maps extracted from different scales are then concatenated along the channel dimension on which another 1×1 linear transformation followed by a ReLU activation is applied. The Softmax activation function is then used to create the final segmentation output. Figure 1 shows our proposed architecture.

2.2. Residual Blocks

The convolutional building blocks in this architecture are based on residual connections [16]. These residual connections help combating the vanishing gradient problem as well providing an identity mapping. We model our convolutional blocks following the architecture proposed in RU-Net in [4]. A 1×1 linear transformation with the number of filters for the specific level l is performed onto the input of each block $\in \{e^l, d^l\}$. This is followed by consecutively applying 4 blocks of 3×3 convolutions and ReLU activations. At the end of each residual block, the residual connection is created by adding the output of the 1×1 linear transformation to the activations of the convolutional operations as shown in Figure 2.

2.3. Attention Gates

The attention gating mechanism for a U-Net based model was first introduced by [5]. The aim of this attention mechanism is to only retain spatially relevant features of the feature map in the shortcut connection, before propagating it to the decoder stage. Let x^l_i be the features x at pixel i from the shortcut connection at level l of the U-Net architecture, who were created by the encoder block e^l. The attention mechanism creates a scale coefficient $a^l_i \in [0, 1]$ by making use of the activations

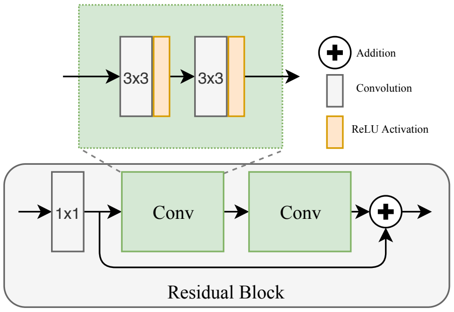

Fig. 2. Diagram of a residual block in our proposed architecture The filters for each convolutional operation are depending on the level l.

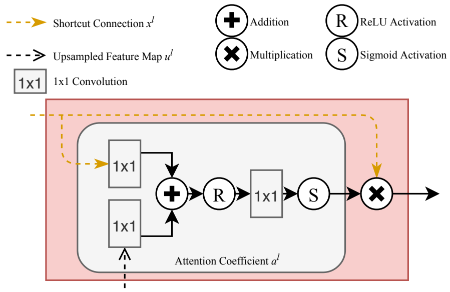

Fig. 3. Diagram of an attention gate using the upsampled feature map u^l as well as the output of the opposing encoder block x^l.

of the previous upsampling operation u_i^l as well as x_i^l to create the scaled output $\hat{x}_i^l$ with $\odot$ denoting the elementwise product: $\hat{x}_i^l = a_i^l \odot x_i^l$. Generating the attention coefficient uses 1×1 linear transformations defined by W_x, W_u , W_v and biases b_u, b_v, as well as the ReLU $\sigma_1(x) = max(0, x)$ and Sigmoid $\sigma_2(x) = \frac{1}{1+\exp -x}$ activation functions:

$$q_{att}^l(x_i^l, u_i^l) = W_v(\sigma_1(W_x x_i^l + W_u u_i^l + b_u) + b_v \quad (1)$$

$$a_i^l = \sigma_2(q_{att}^l(x_i^l, u_i^l)) \quad (2)$$

Training this attention mechanism through backpropagation allows parameters in previous encoder layers to primarily focus on semantically relevant features as the gradients of non relevant regions are being suppressed. Further, the use of denser features upsampled from the previous convolutional block in the gating operation decreases the propagation of noisy or irrelevant activations passing through the shortcut connection to the decoder blocks [5].

In contrast to the implementation in [5], where x_i^l is downsampled to match the spatial dimensions of the output of the previous block followed by later upsampling a^l, we use already upsampled features u^l which match the spatial dimensions of x^l.

3. EXPERIMENTS

3.1. Datasets and Metrics

The Crackforest dataset (CFD) consists of 117 images of size 480 × 320 pixels which contain road surface cracks and their corresponding ground truth segmentation mask. This dataset also features anomalies such as shadows and stains. We also make use of the AigleRN dataset [17]. It features 38 images, half of them being of size 311 × 462 pixels and the other half 991 × 462 pixels. Cracks only occupy a fraction of the total image with the ratios of crack to non crack pixels being 61:1 for the CFD and 139:1 for AigleRN.

To allow a fair evaluation we make use of the commonly used metrics: F1-Score $F1$, Precision Pr and Recall Re. These metrics are calculated using True Positive, False Positive and False Negative predictions. As the annotated ground truth segmentation mask may not be accurate on a per pixel level, a threshold is commonly used to count True Positive pixels [9, 15, 19]. Following the metrics used in [19] we classify a crack as correctly labelled if it lies within a threshold of two pixels to a ground truth crack pixel.

3.2. Training Configuration

The model architecture is trained using a image patch based training approach. Following [19] we split CFD into 71[1] training, 46 testing and AigleRN into 24 training and 14 testing images. This patch based approach extracts a ratio of 48 × 48 pixel patches containing cracks (at minimum one crack pixel) to patches where no crack is present. For both datasets a ratio of 60% crack to non crack patches is used. In the training process for CFD 2000 patches are extracted per image, totalling to 142000 training patches. As AigleRN contains a higher class imbalance as well as varying image sizes the number of patches to extract from each image is set to be dynamic:

$$NP_{image} = \frac{cpx_{image}}{cpx_{train}} * NP_{train} \quad (3)$$

with NP being the number of patches to extract, cpx being the number of pixels containing cracks and $image$ as well as $train$ representing each individual image or the whole training dataset respectively. NP_{train} in AigleRN was set to 84,000.

The Loss function used to train this architecture is the sum of the cross entropy loss function L_{CE} and the dice loss function L_D [20] with y_i denoting the ground truth and $\hat{y}_i$ denoting the prediction at pixel $i \in [0, 1]$ out of all pixels N:

$$Loss = L_{CE} + L_D \quad (4)$$

$$L_{CE} = -\sum_i^N y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i) \quad (5)$$

[1] One image from the training split in CFD was excluded as the authors included an incorrect ground truth segmentation map

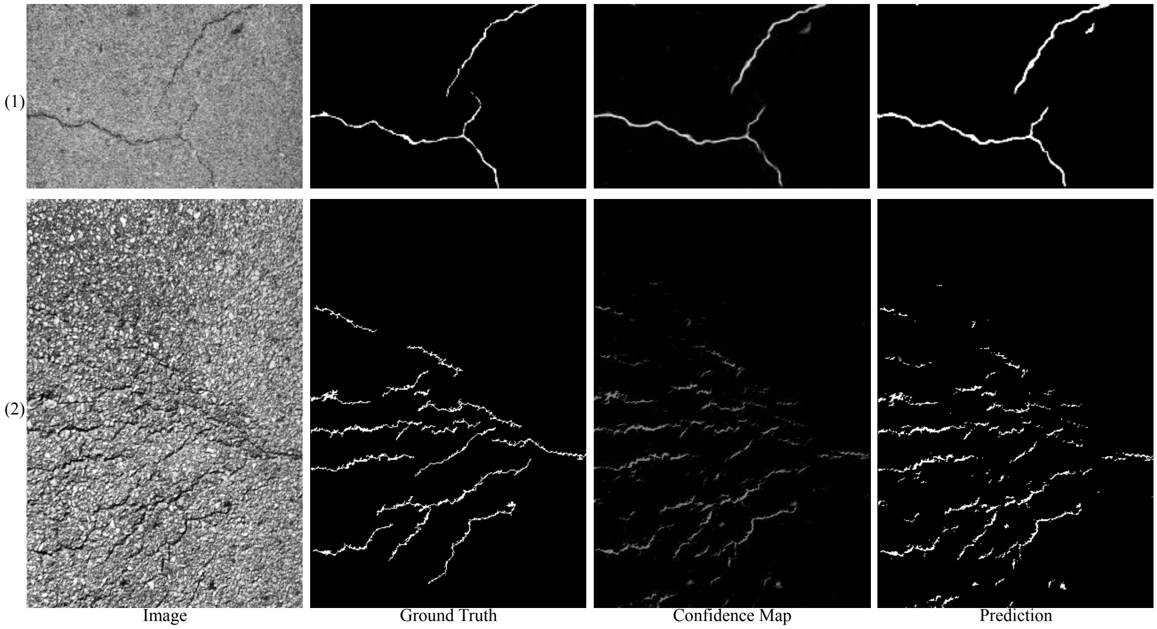

Fig. 4. Sample of segmentation results on (1) CFD and (2) AigleRN

$$L_D = \frac{2\sum_i^N y_i\hat{y}_i + 1}{\sum_i^N y_i + \sum_i^N \hat{y}_i + 1} \tag{6}$$

The architecture is trained on each dataset separately, using 30 epoch on the CFD and 25 epoch on AigleRN. Stochastic Gradient Descent with a momentum of 0.9, a learning rate of 0.001 and weight decay of 1e$-$6 are used for optimisation during training. The batch size is set to 32.

4. RESULTS

Prediction results are generated by extracting patches through a sliding window with a stride of 1 in both the height and width dimension from each image of the testing datasets. The predictions for each pixel of an output segmentation map are then averaged based on all patches containing this pixel. A confidence threshold of 20% is used to generate the segmentation map results. Table 1 shows the results of our proposed architecture on CFD, whereas Table 2 contains the results for AigleRN. It should be noted that in the results on CFD, the Crackforest [9] as well as the U-Net [15] methods use a True Positive pixel threshold of 5 pixels. As it can be seen, our method outperforms the previous state of the art results by 2.5% in $F1$ on CFD and by 0.32% in AigleRN. On CFD this architecture improves PR by 5.18% however it only achieves the second best results in RE. On AigleRN the proposed architecture improves on RE, however it therefore lacks in PR. A sample segmentation map of each dataset is shown in 4.

Table 1. Crack segmentation results on CFD

Method	$\boldsymbol{F1}$	$\boldsymbol{RE}$	$\boldsymbol{PR}$
Crackforest (KNN) [9]	79.44%	78.15%	80.77%
Crackforest (SVM) [9]	85.71%	89.44%	82.28%
U-Net [15]	87.38%	82.82%	92.64%
CNN[19]	92.44%	**95.14%**	91.19%
Proposed Architecture	**94.94%**	93.55%	**96.37%**

Table 2. Crack segmentation results on AigleRN

Method	$\boldsymbol{F1}$	$\boldsymbol{RE}$	$\boldsymbol{PR}$
CNN[19]	89.54%	88.12%	**91.78%**
Proposed Architecture	**89.86%**	**93.04%**	86.90%

5. CONCLUSION

This paper describes a U-Net based architecture utilising an attention mechanism as well as residual convolutional blocks to achieve semantic segmentation of surface cracks. Due to making use of these components, as well as deep supervision and a patch based training and testing approach we achieve new state of the art results on two crack segmentation datasets: Crackforest and AigleRN. This also shows that our proposed architecture is robust and can be utilised on various crack segmentation datasets.

In the future we are interested in applying this architecture to further crack segmentation datasets as well as study if these results carry over to similar fields such as medical image segmentation.

6. REFERENCES

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell, "Fully Convolutional Networks for Semantic Segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[2] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille, "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[3] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2015, pp. 234–241.

[4] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, Tarek M Taha, and Vijayan K Asari, "Recurrent Residual Convolutional Neural Network based on U-Net (R2U-Net) for Medical Image Segmentation," *arXiv preprint arXiv:1802.06955*, 2018.

[5] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, and Others, "Attention U-Net: Learning Where to Look for the Pancreas," *arXiv preprint arXiv:1804.03999*, 2018.

[6] Donald Walker, Lynn Entine, and Susan Kummer, "Pavement Surface Evaluation and Rating Asphalt Road Manual," Tech. Rep., Wisconsin Transportation Information Center, 2002.

[7] Naoki Tanaka and Kenji Uematsu, "A Crack Detection Method in Road Surface Images Using Morphology," *Proceedings of the Workshop on Machine Vision Applications*, pp. 154–157, 1998.

[8] John Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[9] Yong Shi, Limeng Cui, Zhiquan Qi, Fan Meng, and Zhensong Chen, "Automatic Road Crack Detection using Random Structured Forests," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.

[10] Kelwin Fernandes and Lucian Ciobanu, "Pavement Pathologies Classification using Graph-Based Features," in *International Conference on Image Processing*. IEEE, 2014, pp. 793–797.

[11] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436, 2015.

[12] Young-Jin Cha, Wooram Choi, and Oral Büyüköztürk, "Deep Learning Based Crack Damage Detection Using Convolutional Neural Networks," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.

[13] Nhat-Duc Hoang, Quoc-Lam Nguyen, and Van-Duc Tran, "Automatic Recognition of Asphalt Pavement Cracks using Metaheuristic Optimized Edge Detection Algorithms and Convolution Neural Network," *Automation in Construction*, vol. 94, pp. 203–213, 2018.

[14] Qin Zou, Zheng Zhang, Qingquan Li, Xianbiao Qi, Qian Wang, and Song Wang, "DeepCrack: Learning Hierarchical Convolutional Features for Crack Detection," *IEEE Transactions on Image Processing*, vol. 28, no. 3, pp. 1498–1512, 2019.

[15] Mark David Jenkins, Thomas Arthur Carr, Maria Insa Iglesias, Tom Buggy, and Gordon Morison, "A Deep Convolutional Neural Network for Semantic Pixel-Wise Segmentation of Road and Pavement Surface Cracks," in *26th European Signal Processing Conference*, 2018, pp. 2120–2124.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[17] Sylvie Chambon and Jean-Marc Moliard, "Automatic Road Pavement Assessment with Image Processing: Review and Comparison," *International Journal of Geophysics*, vol. 2011, 2011.

[18] Chen-Yu Lee, Saining Xie, Patrick Gallagher, Zhengyou Zhang, and Zhuowen Tu, "Deeply-Supervised Nets," in *Artificial Intelligence and Statistics*, 2015, pp. 562–570.

[19] Zhun Fan, Yuming Wu, Jiewei Lu, and Wenji Li, "Automatic Pavement Crack Detection Based on Structured Prediction with the Convolutional Neural Network," *arXiv preprint arXiv:1802.02208*, 2018.

[20] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi, "V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," in *International Conference on 3D Vision*. IEEE, 2016, pp. 565–571.