Journal of Theoretical and Applied Information Technology
30th November 2019. Vol.97. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EFFICIENT DEEP LEARNING-BASED NETWORK FOR CRACK DETECTION IN PIPELINE SYSTEMS

[1]HOANH NGUYEN, [2]THANH QUYEN NGO

[1] Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

[2] Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

E-mail: [1]nguyenhoanh@iuh.edu.vn, [2]ngothanhquyen@iuh.edu.vn

ABSTRACT

Crack detection is a crucial problem in many tasks such as inspection conditions of concrete pipes or tunnels, diagnosing structural damages, ensuring road safety and so on. Thus, vision-based crack detection had attracted researchers recently, and many approaches for crack detection had been proposed. However, it remains a great challenging task due to the intensity inhomogeneity of cracks and complexity of the background. Inspire by the fast development of deep convolutional neural network (CNN) in image processing recently, we propose a multi-scale deep convolutional network based on encoder-decoder architecture. More specific, our network is based on SegNet network, which is a deep convolutional encoder-decoder architecture designed for pixel-wise semantic segmentation. We first discard the softmax layer in the SegNet network, and then build enhanced modules based on the convolution feature maps from encoder and decoder network. Furthermore, we adopt the focal loss function instead of cross-entropy loss in the original SegNet network to focus on learning the hard examples and down-weighting the numerous easy negatives. Experimental results on public datasets show that our network achieves better results compared to other state-of-the-art methods on crack detection.

Keywords: *Crack Detection, Deep Learning, Convolutional Neural Network, Object Detection, Encoder-Decoder Architecture*

1. INTRODUCTION

Crack is one of the most common defects that can be found on surfaces of various types of physical structures such as the road pavement, the wall, the ceiling of tunnels, concrete pipes or tunnels, and so on. Early detecting and repairing cracks are crucial tasks for preventing the expansion of harms, avoid accidents and keeping the safety in many environments. With the development of technologies in image processing, many approaches have been proposed to applying image processing technologies to perform automatic crack detection. Traditional methods [13-28] usually utilize hand crafted features such as edges, textures, energies, etc. to detect cracks. In the ideal case, if a crack has high contrast to background, traditional methods could detect it with high accuracy. However, in practice cracks may constantly suffer from noise in the background, leading to poor continuity and low contrast. In addition, shadows and lighting conditions may also impact the imaging quality of the crack. These affects commonly lead to degraded performance of the traditional crack detection methods. In recent years, with fast development of deep learning, many methods for crack detection based on deep learning have been proposed and achieved state-of-the-art performances [30-36]. Deep CNN-based methods have also been proposed for tasks such as edge detection [9], [2], contour detection [37], [38], boundary segmentation [3], [39] and so on. These deep frameworks build high-level features from low-level primitives by hierarchically convolving the sensory inputs. In particular, when using deep learning for edge detection, the convolutional features become coarser and coarser in the convolving-pooling pipeline, and the detailed features in larger-scale layers and the abstracted features in the smaller-scale layers can be fused together to improve the performance of edge detection. When using deep learning for image segmentation [12], the convolutional features in the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

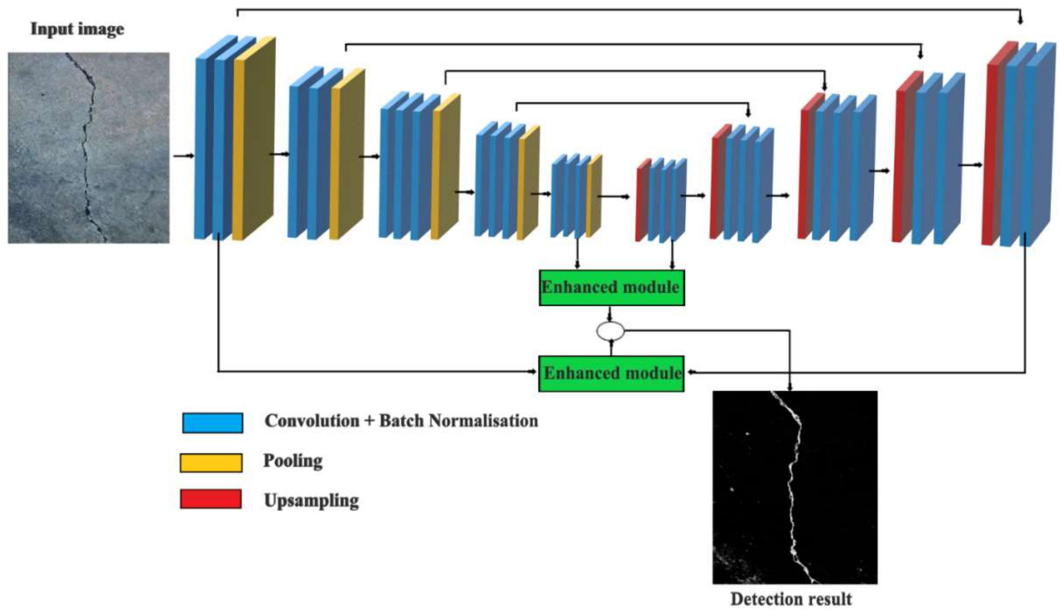

Figure 1: The Structure of The Proposed Framework.

decoder network have been found to be useful to improve the performance of semantic image segmentation, and the indexing of pooling positions can further improve accuracy of boundary localization.

Motivated by the above ideas, we propose a multi-scale deep convolutional network based on encoder-decoder architecture for crack detection. Our proposed method is based on SegNet network [12]. For crack detection, we discard the last Softmax layer in the original SegNet network. Then, we build two enhanced modules in the SegNet network. Enhanced modules take two inputs layer: the convolutional layer before the pooling layer at the first scale and the last scale in the encoder network and the last convolutional layer at the corresponding scale in the decoder network, and generate the overall fused layer in the end of our network. Furthermore, we adopted the focal loss function [29] instead of cross-entropy loss in the original SegNet network to focus on learning the hard examples and down-weighting the numerous easy negatives.

This paper is organized as follows: an overview of previous methods is presented in Section 2. Section 3 describes the detail of the proposed method. Section 4 demonstrates experimental results. Finally, the conclusion is made in Section 5.

2. RELATED WORK

Over the past years, many approaches for crack detection based on vision have been proposed. These approaches can be divided into two categories: traditional approaches and deep learning-based approaches. Traditional approaches often utilize hand crafted features like edges, textures, energies, etc. to detect cracks, including Gabor Filter Invariant to Rotation [13], block binarization [14], thresholding operations [15], edge detection [16], [17], and percolation-based [18], as well as their improved approaches. In [19], Otsu thresholding was used for leakage recognition processes, and a novel algorithm based on the features of the local image grids was developed to recognize cracks. Mohanty et al. [20] showed that image mosaic technology could be effectively used to detect water leakage and cracks in tunnel. In energy minimization-based methods, minimal path searching has been developed for crack detection. In [21] and [22], seed-growing methods built on minimal path searching were proposed for pavement crack detection. In [23], minimal path searching was performed in a path-voting way. In [24], the minimal path searching was used to track cracks in complex background. Machine learning-based methods have also been investigated for crack detection in early works such as Support Vector Machines [25], [26], Artificial Neural Networks [27] and Adabost [28]. Although traditional approaches showed satisfactory performance to some extent, these methods generally relied on exploiting handcrafted low-level

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

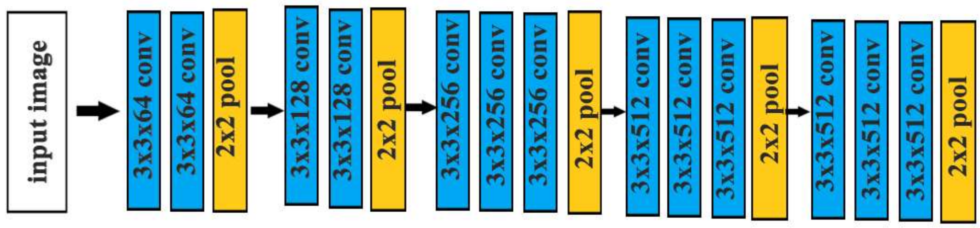

Figure 2: The Architecture of The Encoder Network.

features. These features may be affected under complex environments.

With the fast development of deep convolutional neural network recently, many methods for crack detection based on deep CNN have been proposed and achieved state-of-the-art performances. Li et al. [30] presented a unified and purely vision-based method which had displayed abilities in damage detection and localization networks. Xue and Li [31] achieved automatic intelligent classifications and detections of tunnel lining damages through fully convolutional networks, region proposal networks, and position-sensitive region-of-interest pooling techniques. In order to identify the actual profiles of multiple damage and realize pixel-level detections, Huang et al. [32] used two-stream algorithms of the corresponding FCN, in which one stream was used to recognize crack by a sliding-window-assembling operation, and the other was adopted for leakage by a resizing-interpolation operation. Hoskere et al. [33] proposed a multiscale pixel-wise deep neural network to successfully recognize six different types of structural damages. Li et al. [34] proposed a Fully Convolutional Network for concrete structures at the pixel-level. In [35], deep convolutional neural network was used to classify the image patches into crack blocks and non-crack ones. In [36], fully convolutional neural networks were studied to infer cracks of nuclear power plant using multi-view images. Deep learning-based methods achieve better results compared to traditional methods. However, there still lacks investigation on end-to-end trainable CNN models for robust crack detection.

3. PROPOSED APPROACH

Figure 1 shows the overall framework of the proposed method. Our proposed method is based on SegNet network [12]. SegNet is a deep convolutional encoder-decoder architecture designed for pixel-wise semantic segmentation. SegNet has an encoder network and a corresponding decoder network, followed by a final pixelwise classification layer. For crack detection, we discard the last Softmax layer in the original SegNet. Then, we build two enhanced modules in the SegNet network. Enhanced modules take two inputs layer: the convolutional layer before the pooling layer at the first scale and the last scale in the encoder network and the last convolutional layer at the corresponding scale in the decoder network, and generate the overall fused layer in the end of our network. Furthermore, we adopted the focal loss function [29] instead of cross-entropy loss in the original SegNet network to focus on learning the hard examples and down-weighting the numerous easy negatives. Details of our proposed network is described in next sub-section.

3.1 Encoder Network

The encoder network is based on the VGG-16 [1] architecture. The fully connected layers in original VGG-16 are discard in favor of retaining higher resolution feature maps at the deepest encoder output. Thus, the encoder network consists of 13 convolutional layers and 5 down-sampling pooling layers which correspond to the first 13 convolutional layers and 5 down-sampling pooling layers in the VGG-16 network. This architecture reduces the number of parameters in the encoder network significantly. Figure 2 shows the architecture of the encoder network. Each convolution layer in the encoder network performs convolution with a filter bank to produce a set of feature maps. Then, a batch-normalization step is adopted after these feature maps to speed up the training. Following that, an element-wise rectified-linear non-linearity is applied. At the end of each convolution layer, max-pooling with a 2 x 2 window and stride of 2 are performed to effectively reduce the spatial sizes of the feature maps. In addition, max-pooling indices are generated to capture and record the boundary information in the encoder feature maps when sub-sampling is performed.

3.2 Decoder Network

The decoder network consists of up-sampling layers, convolutional layers, and batch normalizations, which were then followed by a

Journal of Theoretical and Applied Information Technology
30th November 2019. Vol.97. No 22

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Softmax classifier for the purpose of predicting the pixel-wise labels. The decoder network also has 13 convolutional layers, and each decoder layer has a corresponding layer in the encoder network. Each decoder scale in the decoder network upsamples its input feature map using the memorized max-pooling indices from the corresponding encoder feature map. This up-sampling step will produce sparse feature maps. The sparse feature maps obtain more precise location of region boundaries compared with continuous and dense feature maps. Figure 3 illustrates the decoding technique of each decoder scale. These sparse feature maps are then convolved with a trainable decoder filter bank to produce dense feature maps. A batch normalization step is then applied to each of these feature maps.

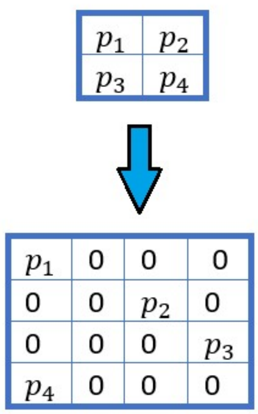

Figure 3: The Decoder in Decoder Network. Here, p_1, p_2, p_3 and p_4 Correspond to Values in A Feature Map.

3.3 Enhanced Module

It has been found that the fusion of the multi-scale convolutional feature maps is proved to be useful for improving the performance of object detection [2], [3], [4], [5]. In this paper, we examine the change of scale of feature maps caused by both the pooling operation in the encoder network and upsampling operation in the decoder network, and build an enhanced module on encoder-decoder architecture of the SegNet. The proposed enhanced module is shown as green color box in Figure 1. At shown in Figure 1, the convolutional layer before the pooling layer at the first scale and the last scale in the encoder network is concatenated to the last convolutional layer at the corresponding scale in the decoder network. The enhanced module handles the concatenated convolutional features with a sequence of operations. Figure 4 shows the structure of the proposed enhanced module in details. First, the feature maps from encoder network and decoder network are concatenated. Then, a 1 x 1 convolution layer is applied to decreases the multi-channel feature maps to 1 channel. Next, in order to calculate pixel-wise prediction loss in each scale, a deconvolution layer is added to up-sample the feature map. Finally, a crop layer is adopted to crop the up-sampling result into the size of the input image. At the end of each enhanced module, we can get the prediction maps of each scale with the same size of the ground-truth crack maps. Furthermore, the prediction maps generated in the enhanced modules are further concatenated, and a 1 x 1 convolution layer is added to fuse the outputs at multiple scales. Thus, we can obtain the prediction maps at each scale and the overall fused layer in the end.

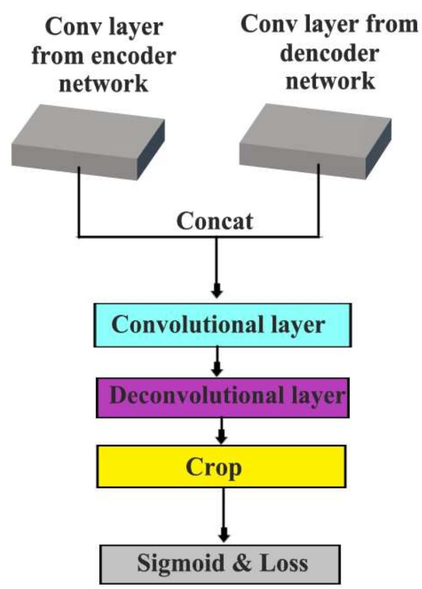

Figure 4: The Structure of The Proposed Enhanced Module.

3.4 Loss Function

The original SegNet is designed for semantic segmentation, so a softmax loss layer was added at the end of the decoder network to measure the prediction error in each object channel. In our proposed network, the output is a 1-channel prediction map that indicates the probability of each pixel belonging to the crack. On the other hands, due to the different sizes of cracks on the same images, cracks at the boundaries and small-scale cracks, the accuracy crack detection using one-stage networks is relatively low. For the purpose of solving above issues, we utilize the focal loss function rather than

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 5: Example Images in CRACK500 Dataset (First Row) and CrackTree Dataset (Second Row).

traditional cross entropy function in order to focus on learning the hard examples and down-weighting the numerous easy negatives. The focal loss function is defined as follows:

$$l(F_t) = -\alpha_t(1 - F_t)^{\gamma}\log(F_t) \quad (1)$$

where $\alpha_t \in [0,1]$ represents the weight variable, and can be set by the inverse class frequency in order to address the class imbalance; and the F_t is defined as follows:

$$F_t = \begin{cases} F & if\ y = 1 \\ 1 - F & otherwise \end{cases} \quad (2)$$

where y represents the ground-truth class and F represents the prediction label with the ground-truth class of 1.

In (1), $(1 - F_t)^{\gamma}$ (with $\gamma \in [0,5]$) is used to reduce the relative loss for the well-classified samples (when $F_t > 0.5$), and focus more on the mis-classified samples (hard samples). In this way, the contributions of the cracks and backgrounds can be balanced to the loss function, which not only improves the efficiency, but also increases the detection accuracy.

4. EXPERIMENTAL RESULTS

In this section, we analyze and compare the performance of the proposed approach with other state-of-the-art approaches on crack detection. The proposed method is implemented on a machine with Intel Core i5 9400 CPU, 8GB of RAM, NVIDIA GeForce GTX 1660Ti GPU. We use TensorFlow for implementing deep CNN frameworks.

4.1 Dataset

In order to train the proposed network and compare the results of our method with other approaches, we adopt recently public datasets for crack detection, including CRACK500 [8] and CrackTree [6]. Figure 5 shows some example images in these datasets. Details of each dataset are described as following:

- CRACK500: In [8], the authors presented a dataset to train and evaluate their proposed method. CRACK500 includes 500 images with resolution around 2000 x 1500. The images were taken on main campus of Temple University using cell phones. The dataset is divided into 250 images of training data, 50 images of validation data, and 200 images of test data. We use all 250 images in training data to train our network. To enlarge the size of the training dataset, data augmentation is applied to all images. We first rotate the images with different angles to create a training set of 1250 images. Then, we flip these images in the vertical and horizontal direction to get 2500 images in total. Finally, we crop 5 sub-images on each image with a size of 480 x 480. After data augmentation, we get a training dataset of 12500 images in total.
- CrackTree: Zou, et al [6] presented a dataset to evaluate their proposed method. CrackTree dataset includes 206 pavement images with resolution 800 x 600. These

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

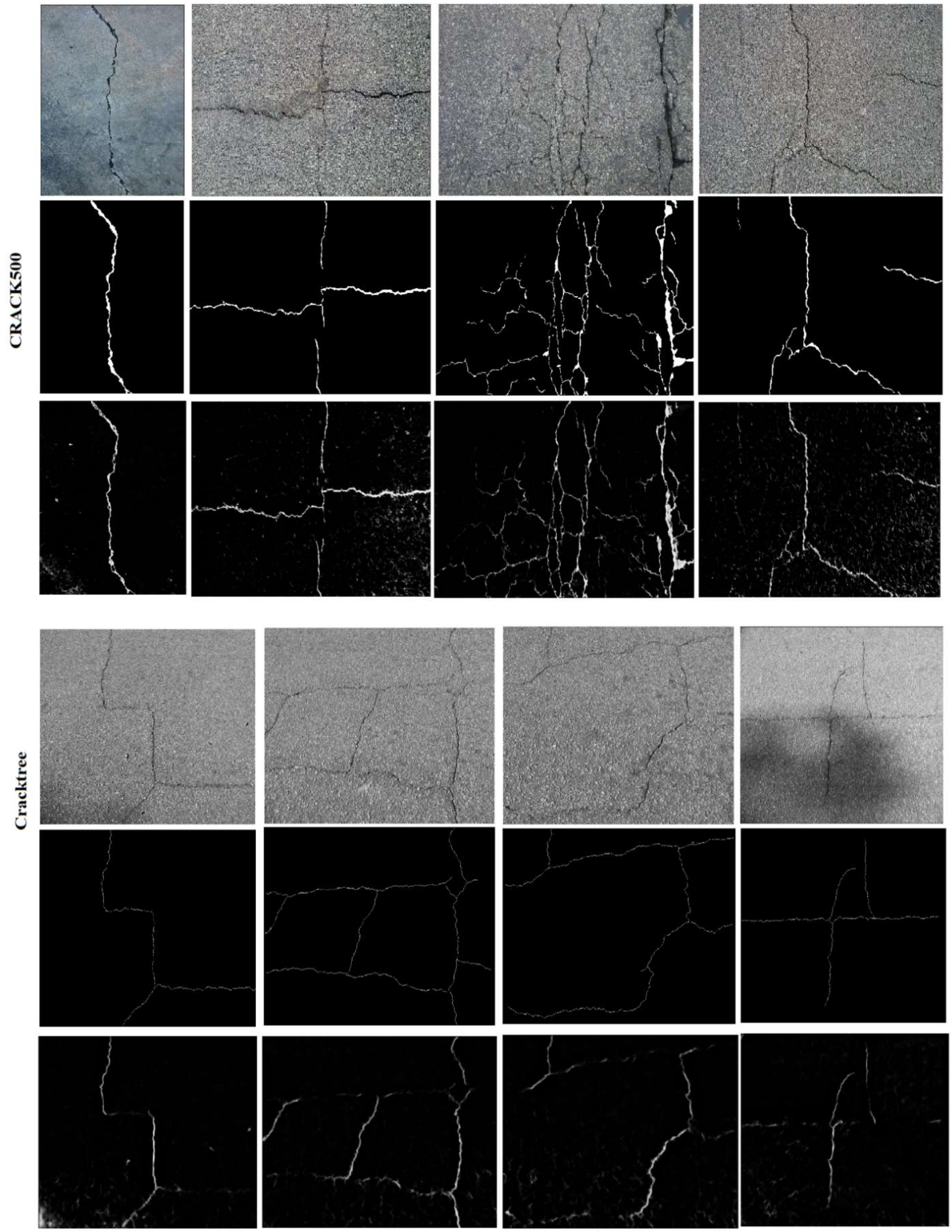

Figure 6: Detection Results of Proposed Method on CRACK500 Dataset and CrackTree Dataset.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1: The AIU, ODS, and OIS of The Proposed Method and Other Methods on CRACK500 and CrackTree Dataset.

Method	CRACK500			CrackTree			Processing time (CRACK500/ CrackTree)
	AIU	ODS	OIS	AIU	ODS	OIS	
HED [9]	0.481	0.575	0.625	0.040	0.317	0.449	0.12s/0.08s
RCF [2]	0.403	0.490	0.586	0.032	0.255	0.487	0.1s/0.0.06s
FCN [10]	0.379	0.513	0.577	0.008	0.334	0.333	0.8s/0.4s
CrackForest [12]	-	0.199	0.199	-	0.080	0.080	6s/4s
Proposed method	0.484	0.599	0.602	0.040	0.424	0.476	0.34s/0.15s

images include various types of cracks in different difficult environments such as shadows, occlusions, low contrast, noise, and so on.

4.2 Evaluation Metrics

Since the similarity with edge detection, many approaches used criteria for edge detection to evaluate crack detection, including the best F-measure on the data set for a fixed scale (ODS) and the aggregate F-measure on the data set for the best scale in each image (OIS) [7]. The ODS and OIS are defined as follows:

$$ODS = max\left\{2\frac{P_t \times R_t}{P_t + R_t} : t = 0.01, 0.02, \dots, 0.99\right\} \quad (3)$$

$$OIS = \frac{1}{N_{img}}\sum_i^{N_{img}} max\left\{2\frac{P_t^i \times R_t^i}{P_t^i + R_t^i} : t = 0.01, 0.02, \dots, 0.99\right\} \quad (4)$$

where t represents the threshold; i represents the index of image; N_{img} represents the total number of images; P_t and R_t represent precision and recall at threshold t over dataset; P_t^i and R_t^i are computed over image i.

For the purpose of comparing the detection results of the proposed method with other state-of-the-art methods, we also use these criteria for evaluating the detection results in this paper. Furthermore, we adopt AIU as in [8] to evaluate our proposed method. AIU is computed on the detection and ground truth without NMS and thinning operation. AIU of an image is defined as follow:

$$AIU = \frac{1}{N_t}\sum_t \frac{N_{pg}^t}{N_p^t + N_g^t - N_{pg}^t} \quad (5)$$

where N_t represents the total number of thresholds $t \in [0.01, 0.99]$ with interval 0.01; for a given threshold t, N_{pg}^t represents the number of pixels of intersected region between the predicted and ground truth crack area; N_p^t and N_g^t represent the number of pixels of predicted and ground truth crack region, respectively. Thus, the AIU is in the range of 0 to 1. The higher value means the better performance. The AIU of a dataset is the average of the AIU of all images in the dataset.

4.3 Performance Results

In this section, we analyze and compare the performance of proposed approach with current state-of-the-art methods, including HED [9], RCF [2], FCN [10] and CrackForest [12]. HED fuses multi-scale convolutional feature maps by using the last convolutional feature map at each stage in VGG-16. We train HED on CRACK500 training dataset. RCF is developed based on HED. RCF fuses multi-scale convolutional features by using all convolutional feature maps at each stage in VGG-16. We train HED on CRACK500 training dataset as the same HED. FCN defined fully convolutional network in semantic segmentation. We replace the loss function with sigmoid cross-entropy loss for crack detection. We train FCN on CRACK500 training dataset with base learning rate is set to 0.00001, momentum is set to 0.99 and weight decay is set to 0.0005. CrackForest uses SE architecture to generate the crack map, and post-processes the crack map to obtain the final crack. We train FCN on CRACK500 training dataset. All the hyperparameters are set as default.

Figure 6 shows examples of detection results on CRACK500 and CrackTree. In this Figure, input test images are shown at the first row (CRACK500) and the fourth row (CrackTree). Ground truth images are shown at the second row (CRACK500) and the fifth row (CrackTree), while detection results images are shown at the third row (CRACK500) and the sixth row (CrackTree). As shown in this Figure, the input images contain shadows and obvious noise. The proposed approach can still generate a crack map very close to the ground truth. Furthermore, the proposed method can exactly detect cracks when input images contain tiny

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

cracks, cracks embedded in the road lane, which can hardly be observed without a careful inspection.

Table 1 shows the quantitative detection results of the proposed method and other state-of-the-art methods on both CRACK500 dataset and CrackTree dataset. As shown, the proposed method achieves the best results on both CRACK500 dataset and CrackTree dataset. More specific, with CRACK500 dataset, the performance of the proposed method is improved comparing with HED, RCF, FCN and CrackForest framework by 0.024, 0.109, 0.086, 0.4 on ODS respectively. With CrackTree dataset, comparing with HED, RCF, FCN and CrackForest framework, the proposed algorithm improves by 0.107, 0.169, 0.09, 0.344 on ODS respectively. For the computational efficiency, our approach needs 0.34 second for processing an image in CRACK500 dataset and 0.15 second for processing an image in CrackForest dataset. HED and RCF can achieve faster speeds, at about 0.12 second and 0.1 second with CRACK500 dataset. CrackForest needs up to 6 seconds to process an image in CRACK500 dataset.

5. CONCLUSIONS

In this paper, we propose a multi-scale deep convolutional network based on SegNet for crack detection. The proposed approach fist discards the Softmax layer in original SegNet architecture. Then, two enhanced modules are build, which take the convolutional layer before the pooling layer at the first scale and the last scale in the encoder network and the last convolutional layer at the corresponding scale in the decoder network as input layers to generate the overall fused layer in the end of proposed network. Furthermore, the focal loss function is adopted to focus on learning the hard examples and down-weighting the numerous easy negatives. Experimental results on two public datasets, including CRACK500 and CrackTree, show that our network achieves better results compared to other state-of-the-art methods.

REFRENCES:

[1] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint* arXiv:1409.1556, 2014.

[2] Y. Liu, M.-M. Cheng, X. Hu, K. Wang, and X. Bai, "Richer convolutional features for edge detection", *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[3] K.-K. Maninis, J. Pont-Tuset, P. Arbel´aez, and L. Van Gool, "Convolutional oriented boundaries", *European Conference on Computer Vision*, 2016, pp. 580–596.

[4] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Convolutional channel features", *IEEE International Conference on Computer Vision*, 2015, pp. 82–90.

[5] Cai Z, Fan Q, Feris RS, VasconcelosN, "A unified multi-scale deep convolutional neural network for fast object detection", Leibe B, Matas J, Sebe N, Welling M (eds) Computer Vision – ECCV 2016. ECCV 2016. *Lecture Notes in Computer Science*, vol 9908. Springer, Cham.

[6] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "CrackTree: Automatic crack detection from pavement images", *Pattern Recognition Letters*, vol. 33, no. 3, pp. 227–238, 2012.

[7] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour detection and hierarchical image segmentation", *PAMI*, vol. 33, no. 5, pp. 898–916, 2011.

[8] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei and H. Ling, "Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection", *IEEE Transactions on Intelligent Transportation Systems*, 2019.

[9] S. Xie and Z. Tu, "Holistically-nested edge detection", *IEEE International Conference on Computer Vision*, 2015, pp. 1395–1403.

[10] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation", *CVPR*, 2015, pp. 3431–3440.

[11] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests", *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.

[12] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[13] Medina. R, Llamas. J, Gómez-García-Bermejo. J, Zalama. E, and Segarra. MJ, (2017). "Crack detection in concrete tunnels using a gabor filter invariant to rotation", *Sensors*, vol. 17, no. 7, Jul. 2017.

[14] Qi. D, Liu. Y, Wu. X, and Zhang. Z, "An algorithm to detect the crack in the tunnel based on the image processing", *2014 Tenth International Conference on Intelligent Information Hiding and Multimedia Signal*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Processing, Kitakyushu, Japan, August 27-29. 2014, pp. 860-863.

[15] Zhang. W, Zhang. Z, Qi. D, and Liu. Y, "Automatic crack detection and classification method for subway tunnel safety monitoring", *Sensors*, vol. 14, no. 10, pp. 19307-19328, Oct. 2014.

[16] Chen. L. C, Jan. H. H, and Haung. C. W, "Mensuration of concrete cracks using digital close-range photographs", *Proceedings of the 22th Asian Conference on Remote Sensing*, Singapore, 2001, pp. 1248-1253.

[17] Santhi, B. & Krishnamurthy, G. & Siddharth, S. & Ramakrishnan, P.K., "Automatic detection of cracks in pavements using edge detection operator", *Journal of Theoretical and Applied Information Technology*, vol. 36, 2012, pp. 199-205.

[18] Yamaguchi. T, and Hashimoto. S, "Fast crack detection method for large-size concrete surface images using percolation-based image processing", *Machine Vision and Applications*, vol. 21, no. 5, pp. 797-809, Aug. 2010.

[19] Huang. H, Sun. Y, Xue. Y, and Wang. F, "Inspection equipment study for subway tunnel defects by grey-scale image processing", *Advanced Engineering Informatics*, vol. 32, pp.188-201, Apr. 2017.

[20] Mohanty. A, and Wang. T. T, "Image mosaicking of a section of a tunnel lining and the detection of cracks through the frequency histogram of connected elements concept", *2012 International Workshop on Image Processing and Optical Engineering*, Harbin, China, January 8-9. 2012, pp. 83351P.

[21] Q. Li, Q. Zou, D. Zhang, and Q. Mao, "FoSA: F* seed-growing approach for crack-line detection from pavement images", *Image and Vision Computing*, vol. 29, no. 12, pp. 861–872, 2011.

[22] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart, "Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection", *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 10, pp. 2718–2729, 2016.

[23] Q. Zou, Q. Li, F. Zhang, Z. Xiong, and Q. Wang, "Path voting based pavement crack detection from laser range images", *International Conference on Digital Signal Processing*, 2016, pp. 432–436.

[24] V. Kaul, A. Yezzi, and Y. Tsai, "Detecting curves with unknown endpoints and arbitrary topology using minimal paths", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 10, pp. 1952–1965, 2012.

[25] Qu. Z, Feng. H, Zeng. Z, Zhuge. J, and Jin. S, "A SVM-based pipeline leakage detection and pre-warning system", *Measurement*, vol. 43, no. 4, pp. 513-519, May. 2010.

[26] Mustafa, R. & Mohamed, E.A., "Concrete crack detection based multi-block CLBP features and SVM classifier", *Journal of Theoretical and Applied Information Technology*, vol. 81, 2015, pp. 151-160.

[27] Butcher. J. B, Day. C. R, Austin. J. C, Haycock. P. W, Verstraeten. D, and Schrauwen. B, "Defect detection in reinforced concrete using random neural architectures", *Computer-Aided Civil and Infrastructure Engineering*, vol. 29, no. 3, pp. 191-207, 2014.

[28] Cord. A, and Chambon. S, "Automatic road defect detection by textural pattern recognition based on AdaBoost", *Computer-Aided Civil and Infrastructure Engineering*, vol. 27, no. 4, pp. 244-259, Apr. 2012.

[29] Lin. T. Y, Goyal. P, Girshick. R, He. K, and Dollár. P, "Focal loss for dense object detection", *Proceedings of the IEEE international conference on computer vision*, Venice, Italy, Oct 22-29. 2017, pp. 2980-2988.

[30] Li. R, Yuan. Y, Zhang. W, and Yuan. Y, "Unified vision-based methodology for simultaneous concrete defect detection and geolocalization", *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, pp. 527-544, Feb. 2018.

[31] Xue. Y, and Li. Y, "A fast detection method via region-based fully convolutional neural networks for shield tunnel lining defects", *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 8, pp. 638-654, 2018.

[32] Huang. H. W, Li. Q. T, and Zhang. D. M, "Deep learning-based image recognition for crack and leakage defects of metro shield tunnel", *Tunnelling & Underground Space Technology*, vol. 77, pp. 166-176, 2018.

[33] Hoskere. V, Narazaki. Y, Hoang. T, and Spencer Jr. B, (2018). "Vision-based structural inspection using multiscale deep convolutional neural networks", *3rd Huixian International Forum on Earthquake Engineering for Young Researchers*, University of Illinois, Urbana-Champaign, United States, Aug 11-12. 2017.

[34] Li. S, Zhao. X, and Zhou. G, "Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network",

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Computer-Aided Civil and Infrastructure Engineering, pp. 1-19, 2019.

[35] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network", *IEEE International Conference on Image Processing*, 2016, pp. 3708–3712.

[36] S. J. Schmugge, L. Rice, J. Lindberg, R. Grizziy, C. Joffey, and M. C. Shin, "Crack segmentation by leveraging multiple frames of varying illumination", *IEEE Winter Conference on Applications of Computer Vision*, 2017, pp. 1045–1053.

[37] W. Shen, X. Wang, Y. Wang, X. Bai, and Z. Zhang, "Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection", *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3982–3991.

[38] J. Yang, B. Price, S. Cohen, H. Lee, and M.-H. Yang, "Object contour detection with a fully convolutional encoder-decoder network", *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 193–202.

[39] A. Khoreva, R. Benenson, M. Omran, M. Hein, and B. Schiele, "Weakly supervised object boundaries", *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 183–192.