Road Pavement Crack Detection Using Deep Learning with Synthetic Data

I.A. Kanaeva, Ju.A. Ivanova
Division for Information Technology
Tomsk Polytechnic University
Tomsk, Russia
iap15@tpu.ru, jbolotova@tpu.ru

***Abstract*—Robust automatic pavement crack detection is critical to automated road condition evaluation. Manual crack detection is extremely time-consuming. Therefore, an automatic road crack detection method is required to boost this process. This study makes literature review of detection issues of road pavement's distress. The paper considers the existing datasets for detection and segmentation distress of road and asphalt pavement. The work presented in this article focuses on deep learning approach based on synthetic training data generation for segmentation of cracks in the driver-view image. A synthetic dataset generation method is presented, and effectiveness of its applicability to the current problem is evaluated. The relevance of the study is emphasized by research on pixel-level automatic damage detection remains a challenging problem, due to heterogeneous pixel intensity, complex crack topology, poor illumination condition, and noisy texture background.**

Keywords—automatic pavement crack detection, synthetic data generation, deep convolutional neural network, semantic segmentation, image processing

I. Introduction

Current progress in computer vision, which is based on deep learning, has been achieved mostly due to creation of large quantity of labeled data sets. Such spheres as for example autonomous driving systems, associated with the analysis of environmental images and the detection and tracking of moving objects are actively developing. Such semantic segmentation datasets as Cityscapes [1], Wilddash [2] and KITTI [3] are commonly used for deep learning. The labeling of such kind of data is performed manually and that is why the process is expensive and labour-consuming. Datasets mainly contain examples of such classes as the roadway, pedestrian crossing, vehicle, sky, road sign and other common elements of the road.

Recently, in the Russian Federation, the national project "Safe and high-quality roads" has been carried and now it is actively developing. The goal of this project is to improve the quality of significant regional roads and road network of urban agglomerations up to the standard state.

Due to the need of computer, processing of high-quality video data of roads in the road industry there is requirement for developing automatic algorithm of road surface defects detection by images. The development of an effective algorithm for detection roadway defects on images is quite an important issue, since its results can be used both for automatic roads diagnostic and for autonomous driving vehicles creation.

II. Data and Methods

A. Overview methods

Over the past decade, many different techniques of image processing have been proposed in the field of automatic detection, classification and segmentation of pavement distress. A. Mohan and S. Poobal in [4] reviewed 50 research papers in the area and provided the collective survey of the different image processing techniques used for the detection of the cracks in the engineering structures.

German researchers in paper [5] divided the algorithms developed for evaluation of the pavement surface into three major groups: Crack image thresholding, patch-based classification, and depth-based algorithms.

The first group of methods, which detect road damage structures, bases on image processing methods that segment distress textures on image by threshold filtering. In order to reduce illumination artifacts image-preprocessing algorithms are preceded. Due to pixels of cracks have minimum intensity threshold filtering is applied after preprocessing. On last stage, the detection is refined by morphological image operations and by searching for connected components. Next papers present the aforementioned approach.

In [6] the research results were implemented as a CrackIT software tool for segmentation of cracks in an image taken directly above the road surface. The CrackTree [7] toolbox is based on the construction of probability map of pixels belonging to a crack on the image that previously was preprocessed by geodesic shadow-removal algorithm. Paper [8] proposes a new unsupervised multi-scale fusion crack detection algorithm that works on a series of images smoothed by different-scale Gauss filters and combines the resulting masks. Gabor filters in [9] are used for searching candidate areas as cracks. Russian scientists [10] implemented an interactive algorithm to isolate a pavement defect on image using active contours method.

The algorithms of the second group apply various types of classifiers to patch of the image to determine distress areas or cracks. One part of the researchers initially selects a certain feature vector from the considered image region, and then use it as input of the classifier. The advantage of this approach is that the size of the region is not fixed, but initially you need to divide the image into a sufficient number of regions. In [11] regions are defined using the over-segmentation algorithm SLIC. Then support vector machines were used as a binary classifier. Despite the fact that the method does not have high accuracy, it allows to calculate the ratio of damaged and non-damaged pavement. In addition, this method can be expanded for different defects detection, such as pothole, manhole and road marking.

With the advent of public available datasets of road images with pavement distress, such as GAPs [5] and CRACK500 [12], many researchers have used deep learning approaches for the problem. For example, in [13] a truncated CNN VGG16 is used for extract the feature vector from the input image. Next, a neural network with one hidden layer of 256 neurons classifies the feature vector.

978-1-7281-2896-2/19/$31.00©2019 IEEE

B. Existing datasets

Consider the most popular and public available datasets for road distress classification and detection tasks:

1) GAPs dataset [5]: includes total 1,969 gray valued images with resolution of 1,920×1,080 pixels. These images are divided into 64×64 patches and each patch is labeled as a crack or not. The pictured surface material contains pavement of three different German federal roads.

2) CRACK500 dataset [14]: consists of 500 RGB images of pavement cracks of size around 2,000×1,500 pixels that were collected on main campus of Temple University using cell phones. Each crack image has a pixel-level annotated binary map.

3) CrackTree200 dataset [7]: contains 206 pavement images of size 800×600 with various types of cracks that were annotated in pixel-wise level. The images have complex asphalt context with shadows, occlusions, low contrast and noise.

4) CFD dataset [15]: have 118 images of annotated road crack of size 480×320 pixels. These images was captured on urban Beijing roads. The images contain a significant amount of noisy pixels like oil spots and water stains, and some of them are under the poor illumination condition.

5) Road Damage dataset [16]: consist of 9,053 labeled road images of size 600×600, acquired from a smartphone camera installed on the dashboard of a car. The main aim of this is capturing general front view from driver's position, as opposed to capturing images above the road surface. It dicreases difficulty of capturing image process and increases practical applicability of such images. The dataset has 15,435 bounding boxes of damages in total, annotated for the dataset.

The Road Damage dataset was collected in seven Japan municipalities and has eight classes of pavement distress: five types of cracks, two types with wear of road marking, one class that combines other damages like rutting, bump, pothole and separation. The dataset has a PASCAL VOC [17] format and was presented at the IEEE International Conference On Big Data Cup in 2018.

The dataset of Japanese scientists has revived interest in solving the challenge of automatic road damage detection using machine learning methods and, in particular, convolutional neural networks. The advantage of their data are sufficiently large scale of dataset for deep learning, as well as the presence of different distress types, not only cracks. The disadvantage of this defects detection method is the use of bounding boxes. Due to various forms and sizes of damage on image, bounding boxes can include a lot of another information, in particular in liner crack detection. Also for assessing a road's pavement quality, the best approach is pixel-level segmentation using a mask, which allows not only precisely localization the damage, but also estimation its area.

Creating dataset like described above, especially with high quality pixel-level annotations, is a laborious and time-consuming process, as it requires manual labeling of object's pixels in the image. However, there is another approach to deal with this problem - synthetic data generation. In next section, we propose algorithm for generation instance-level synthetic dataset for crack segmentation based on well-known collections with a marked road, such as KITTI and Cityscapes dataset.

III. Related Work

A. Synthetic dataset creation

At current stage of machine learning evolution, the formation of a set of training data have paramount importance for the successful solution of the tasks of detection and segmentation. However, the meticulous manual targeting of several thousands of images is an enormous and sufficiently labor-consuming process, so quite important task is to develop methods for obtaining a representative synthetic samples.

A synthetic dataset is a repository of data that is generated programmatically and cannot be collected by any real-life survey or experiment. For creating the synthetic road crack dataset, we decided to use three publically available sets: KITTI and Cityscapes dataset as images of the road scene, CFD as a source of cracks marked at the pixel level (Fig. 1).

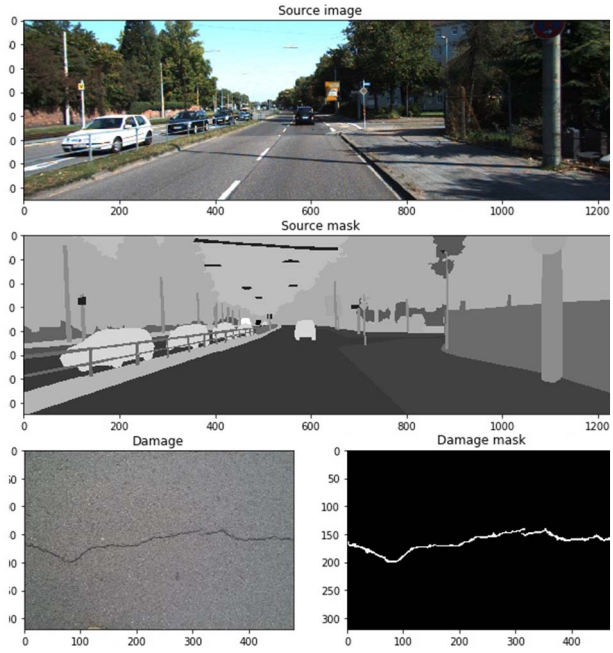


Fig. 1. Road and crack images with masks.

To determine the part of the image corresponding to the roadway, all the pixels of the road mask are chosen. Then, the connected components algorithm with 8-connectivity is applied to the resulting binary mask for determine connected areas. As a result, the area with the maximum number of pixels is taken as the main road mask (Fig. 2, main road mask is highlighted in gray).

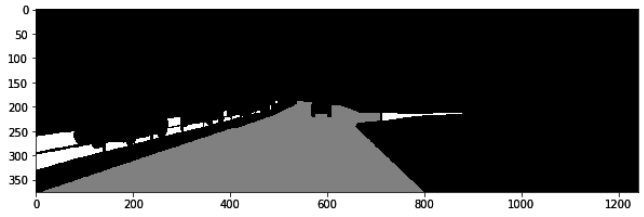


Fig. 2. Maximal area of roadway.

Then the original image of the crack and its mask are cropped at the minimum bounding rectangular for the purpose of reducing following calculations. After that, the image of crack with the mask is scaled and rotated randomly. The result is an image of the crack D and its mask D^{mask} (Fig. 3).

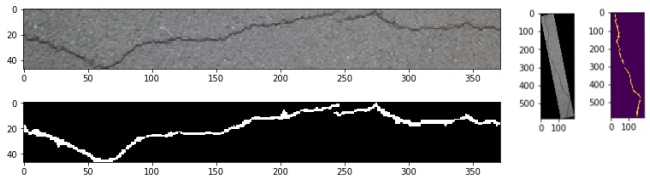


Fig. 3. Cropping, rotating and scaling of crack.

In the next step a point, which is the center of the overlapping area, is selected inside the mask of the main area of the road, and D^{mask} is cut out of the original image mask, equal in size to D^{mask}. To mix two images in the mask area, the mean asphalt value $\overline{D_c}$ are calculated in each channel c for the image of the crack that does not lie under the mask:

$$\overline{D_c} = \frac{1}{k}\sum_p D_c(p)\cdot\left(1-D^{mask}(p)\right), c\in\{R,G,B\}, \quad (1)$$

where $D_c(p)$ – pixel value p of crack image D in channel c, $D^{mask}(p)$ – pixel value p in binary crack mask, k – number of pixels for which $D^{mask}(p)=0$. This calculation process is possible due to the homogeneous texture of the asphalt on the image of the defect. Then, when a crack is applied to an image of a road, only the values under the mask of the roadway and the cracks are considered:

$$M(p) = D^{mask}(p)\cdot S^{mask}(p), \quad (2)$$

The crack is added by changing the pixel values of the original image S. These pixels are located under the common mask M and their values are multiplied by the ratio of the calculated mean asphalt surface to the crack pixel values:

$$S_c(p) = S_c(p)\cdot M(p)\cdot D_c(p)/\overline{D_c}, \quad (3)$$

The algorithm result is shown in Fig. 4.

Fig. 4. The result of synthetic cracks generation.

For the purpose of increasing the informational capacity of the training image sample, from 1 to 5 cracks, which can intersect were generated on one image. As a result, the training set contains 1524 images, and the test dataset includes 505.

B. *Object Segmentation System*

To solve the crack detection problem in pixel level as segmentation task, we decided to use two modern convolution neural network systems: Mask R-CNN [18] and U-Net [19]. Consider its structures and main principles of operation.

a) Mask R-CNN is a state-of-the-art framework for detecting objects in an image while simultaneously generating a high-quality segmentation mask for each inctance. Mask R-CNN extends object detection algorithm Faster R-CNN by adding a module for predicting an object mask. Simplify Mask R-CNN structure is illustrated in Fig. 5. Mask R-CNN have a complex, flexible and powerful block architecture with two stage: generating object proposals and classifying proposals to generate bounding boxes and masks in parallel.

Initially, the image is fed to the input of Mask R-CNN to produce a feature map, which often uses pre-trained VGG16 or ResNet50/101 with excluded layers responsible for classification and named backbone. One of the improvements in this framework is using Feature pyramid networks (FPN) for generation multi-scale feature maps. Sequential layers of FPN with decreasing dimension are considered as a hierarchical pyramid, in which the lower level maps have high resolution, and the upper level maps have high generalizing, semantic ability.

The resulting feature maps are processed in CNN Region Proposals Network (RPN), whose task is to create the regions of interests (RoIs) which may contain objects. For this purpose, each feature map is scanned by lightweight neural network with a 3×3-convolution layer. Output of RPN is based on k anchors - set of boxes with predefined locations and scales relative to images. For each anchor, RPN generates a probability of a proposal having the target object, and a refinement of the coordinates of the bounding box of the object. The purpose of this stage is to identify regions of interests that may contain objects. At the end, duplicate proposal regions are discarded due to non-maximum suppression operation.

Then the proposals are mapped from corresponding feature map levels, extracted from its and resized to the same size using the RoI Align operation. According to RoIs, the final operations of classification, refinement of the bounding box's coordinates and mask prediction are performed at the second stage. The output mask has a greatly reduced size, but contains real values, which allow to obtain sufficient accuracy by scaling the mask to the size of the selected object's bounding box.

b) U-Net model is a fully convolutional network that outputs a classification of each pixel in the image to generate a segmentation mask. U-Net architecture consists of a contracting path to capture context and of a symmetrically expanding path that enables precise localization. The contracting path follows the typical architecture of a convolutional network with alternating convolution and pooling operations and progressively down-samples feature maps, increasing the number of feature maps per layer at the same time. Every step in the expansive path consists of an up-sampling of the feature map followed by a convolution.

Typically, U-Net is trained from scratch starting with randomly initialized weights. In order to account that our training dataset is synthetically generated dataset, it only models plain cracks on road surface without perspective distortion and light aspects, we use transfer learning similar to TernausNet [20]. We used U-Net type architecture improved by the using of the pre-trained VGG16 on ImageNet as the encoder. Initialized weights from the pre-trained network are frozen. This approach allows significantly reducing the number of trainable parameters and shows better performance than training from scratch. To construct an encoder, we kept only first four convolution blocks with last convolutional layer with 512 channels. This U-Net architecture is illustrated in Fig. 6.

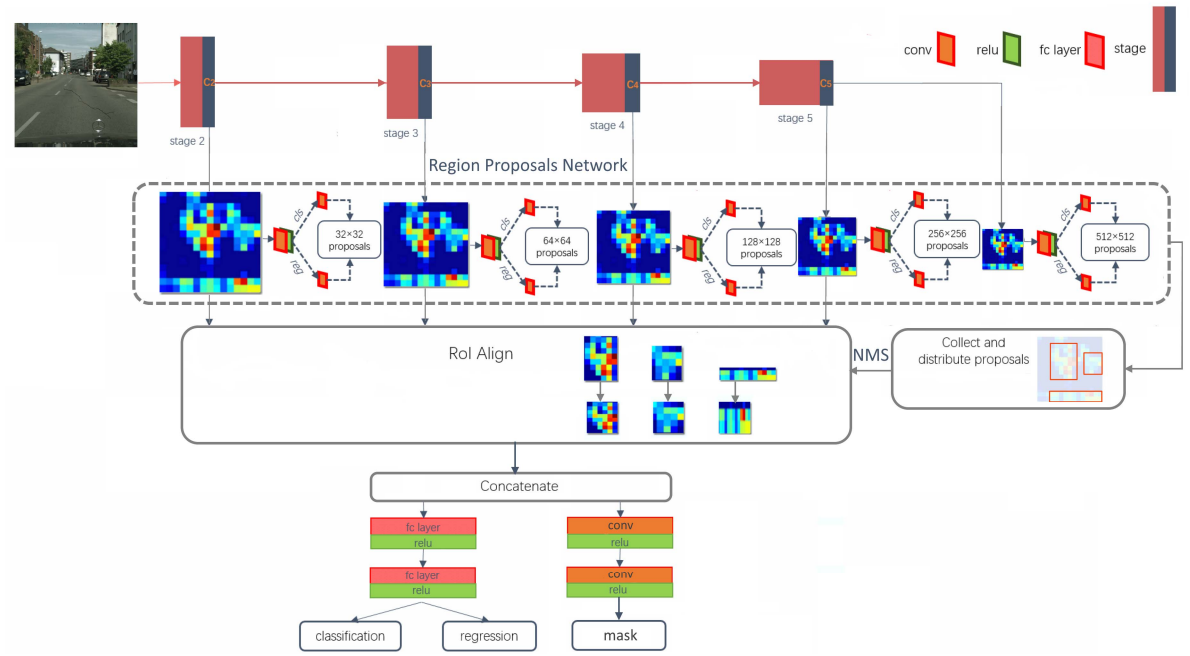


Fig. 5. Mask R-CNN architecture.

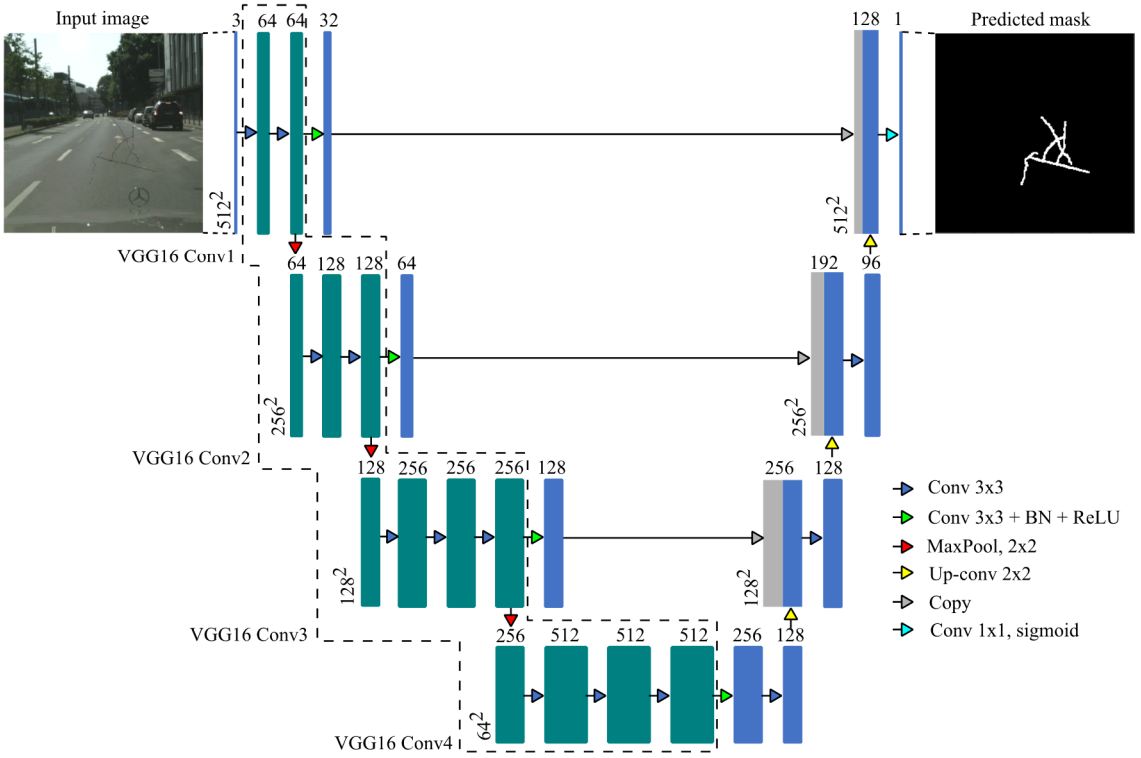


Fig. 6. VGG16 + U-Net architecture.

IV. Experiments

The obtained synthetic set of images with per-pixel labeled road cracks was used to train modern convolutional neural networks Mask R-CNN and U-Net. For our research we used Abdulla et al. [21] Tensorflow + Keras implementation of Mask R-CNN with modifications to perform our dataset. ResNet101 with FPN forms backbone for feature maps construction. Due to the fact that the synthetic dataset cannot contains all characteristics of real asphalt cracks, the transfer learning technology was used in combination with pre-trained model ResNet101 on the MS-COCO dataset. Transfer learning allows accelerating the training process and improving the performance of a model.

We use RGB images of size 1024 × 1024 as input of the Mask R-CNN. In addition, the following values are used as anchor scales: 0.33, 0.5, 1, 2, 3. The training took out 40 epochs of 400 iterations using mini-masks with size 56 × 56 pixels to optimize the used computer memory. We get the best result using learning rate annealing by starting with 0.001 and decreasing it by a factor of 10 every 10 epochs. Removing detection results of the same class happen if there is more than 0.7 value of IoU among bounding boxes.

VGG16 + U-Net realization in input uses RGB image with 512 × 512 resolution and output prediction mask of the same size. Pre-trained VGG16 on ImageNet is used as the encoder. Images from the synthetic dataset resized to 472 × 472 size

and padded to the input size. This operation allows to avoid losing border pixel in the input due to convolution sequence. In these experiments, we use Adam optimizer with a batch size of 8, momentum of 0.9 and a learning rate of 0.001 with decay of 0.000001 and trained the models for 17 epochs.

All the experiments have been conducted on a workstation with a NVIDIA Tesla K80 GPU graphics card machine with 13 GB DDR5X memory on Google Colaboratory platform.

V. Results & Evaluation

Instance segmentation using Mask R-CNN often is evaluated in PASCAL VOC style. The best Mask R-CNN's result estimated by the average precision (AP) metric with the value IoU = 0.5 on the validation synthetic subset is 78.1%.

Jaccard index (Intersection over Union) is chosen as evaluation metric for segmentation task. It reflects similarity measure between a finite number of sets. The measure between two sets A and B is defined as following:

$$IoU(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}, \qquad (4)$$

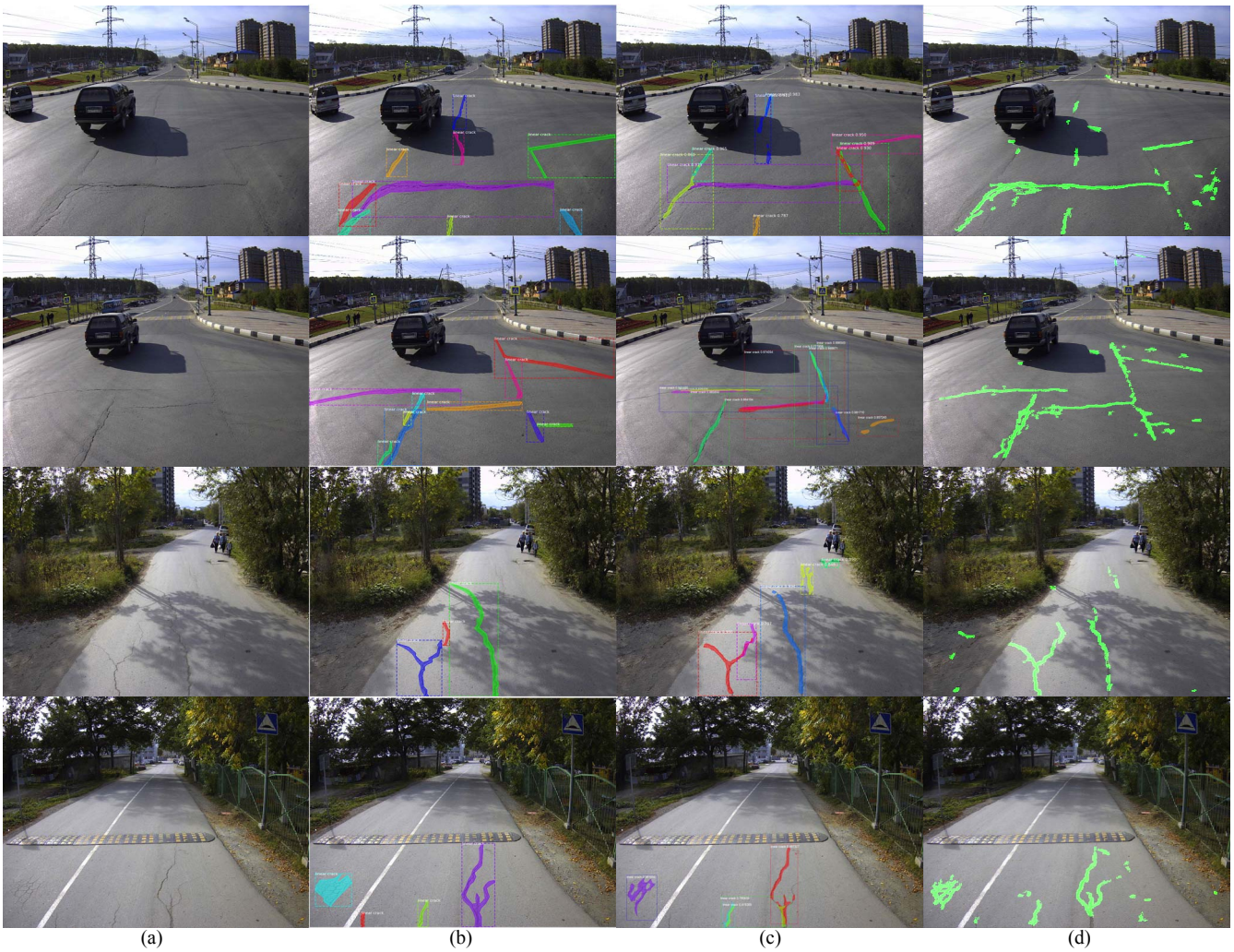
Pixel accuracy metric not used in the evaluation stage, because it can provide misleading results for the reason of small crack class representation within the image in contrast of negative case.

To evaluate the relevance of the training on our synthetic dataset we constructed a small dataset of 67 real images with cracks on carriageways that were manually labeled. These real images were obtained by digital camera mounted at a roof of a vehicle. The collected real-image dataset consists of consecutive frames of two street and has different cracks, potholes, shadows, road marking, manholes, road facilities and equipment. The presence of these elements on the images leads to the conclusion that the real-image dataset is representative.

Finally, the IoU score for segmentation results on synthetic and real image dataset was calculated and summarized in Table 1.

TABLE I. Intersection Over Union Evaluation

Model	**Dataset**		
	Synthetic		***Real-image***
	training	***validation***	
Mask R-CNN	0.79	0.59	0.46
U-Net	0.81	0.56	0.47



(a) (b) (c) (d)

Fig. 7. Segmentation results: (a) examples of pavement surface crack from real-image dataset, (b) their grountruth labels, (c) instance-segmentation result by Mask R-CNN, (d) pixel-wise segmentation results by U-Net.

Fig. 7 shows four examples from real-image dataset and corresponding results of manual human annotation, Mask R-CNN detection and VGG16+U-Net segmentation.

VI. Conclusions

The automatization of road condition control is current trend for practical application of computer vision methods. The conducted analytical review of road distress detection problem has shown that crack detection in pixel level in the driver-view image is a priority and challenging task. We have proposed a novel deep learning approach based on synthetic training data generation for segmentation of cracks in the images with road pavement. The synthetic data generation algorithm allows to easily obtain crack training dataset of any size for instance segmentation task. It makes possible to use state-of-the-art Mask R-CNN-based and U-Net-based segmentation model. The models, trained on synthetic crack data, give acceptable outcomes over 47% of IoU metrics on real images with surfaces' cracks. It should be noted that the proposed approach is weak sensitive to shadows, road marking and light condition. Improving the detection results is possible with additional training of models with attention to road technical facilities, such as manholes and restoration patches.

Acknowledgment

The reported study was funded by RFBR according to the research project № 18-08-00977 A.

References

[1] M. Cordts et al., "The Cityscapes Dataset for Semantic Urban Scene Understanding."

[2] O. Zendel, K. Honauer, M. Murschitz, D. Steininger, and G. Fern, "WildDash - Creating Hazard-Aware Benchmarks."

[3] J. Fritsch, T. Kuhnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC, pp. 1693–1700, 2013.

[4] A. Mohan and S. Poobal, "Crack detection using image processing: A critical review and analysis," Alexandria Eng. J., vol. 57, no. 2, pp. 787–798, 2018.

[5] M. Eisenbach et al., "How to get pavement distress detection ready for deep learning? A systematic approach," in 2017 International Joint Conference on Neural Networks (IJCNN), 2017, pp. 2039–2047.

[6] H. Oliveira and P. L. Correia, "CrackIT - An image processing toolbox for crack detection and characterization," 2014 IEEE Int. Conf. Image Process. ICIP 2014, pp. 798–802, 2014.

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

[8] H. Li, D. Song, Y. Liu, and B. Li, "Automatic Pavement Crack Detection by Multi-Scale Image Fusion," IEEE Trans. Intell. Transp. Syst., pp. 1–12, 2018.

[9] M. Salman, S. Mathavan, K. Kamal, and M. Rahman, "Pavement crack detection using the Gabor filter," IEEE Conf. Intell. Transp. Syst. Proceedings, ITSC, no. October, pp. 2039–2044, 2013.

[10] S. Sudakov, A. Semashko, O. Barinova, A. Konushin, V. Kinshakov, and A. Krylov, "Detection of road lane marking and artifacts of road surface," in Graphicon 2008 Proceedings, 2008.

[11] S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, "Vision for road inspection," 2014 IEEE Winter Conf. Appl. Comput. Vision, WACV 2014, pp. 115–122, 2014.

[12] L. Zhang, F. Yang, Y. Daniel Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," Proc. - Int. Conf. Image Process. ICIP, vol. 2016-Augus, no. October 2017, pp. 3708–3712, 2016.

[13] K. Gopalakrishnan, S. K. Khaitan, A. Choudhary, and A. Agrawal, "Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection," Constr. Build. Mater., vol. 157, pp. 322–330, Dec. 2017.

[14] F. Yang, L. Zhang, S. Yu, D. Prokhorov, X. Mei, and H. Ling, "Feature Pyramid and Hierarchical Boosting Network for Pavement Crack Detection," pp. 1–11, Jan. 2019.

[15] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic Road Crack Detection Using Random Structured Forests," IEEE Trans. Intell. Transp. Syst., vol. 17, no. 12, pp. 3434–3445, Dec. 2016.

[16] H. Maeda, Y. Sekimoto, T. Seto, T. Kashiyama, and H. Omata, "Road Damage Detection and Classification Using Deep Neural Networks with Smartphone Images," Comput. Civ. Infrastruct. Eng., vol. 33, no. 12, pp. 1127–1141, 2018.

[17] M. Everingham et al., "The PASCAL Visual Object Classes Challenge: A Retrospective," Int J Comput Vis, vol. 111, pp. 98–136, 2015.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN."

[19] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation."

[20] V. Iglovikov and A. Shvets, "TernausNet: U-Net with VGG11 Encoder Pre-Trained on ImageNet for Image Segmentation."

[21] W. Abdulla, "Mask r-cnn for object detection and instance segmentation on keras and tensorflow," 2017. [Online]. Available: https://github.com/matterport/Mask_RCNN.