in: Arbeitsgruppentagung Infrastrukturmanagement, Duisburg, Germany, pp. 94-103, 2017

Crack Detection with an Interactive and Adaptive Video Inspection System

Markus Eisenbach, Ronny Stricker, Klaus Debes, Horst-Michael Gross[1]

Ilmenau University of Technology
Neuroinformatics and Cognitive Robotics Lab
98684 Ilmenau, Germany
markus.eisenbach@tu-ilmenau.de

Road condition acquisition and assessment are the key to guarantee their permanent availability. In order to maintain a country's whole road network, millions of high-resolution images have to be analyzed annually. Currently, this requires cost and time excessive manual labor. We aim to automate this process to a high degree by applying deep neural networks. In this paper, we evaluate computer vision based and deep learning based crack detection approaches on the GAPs dataset, which is the first freely available pavement distress dataset of a size, large enough to train high-performing deep neural networks. It provides high quality images, recorded by a standardized process fulfilling German federal regulations, and detailed distress annotations. This is the first fair comparison of research in this field on a large and standardized dataset.

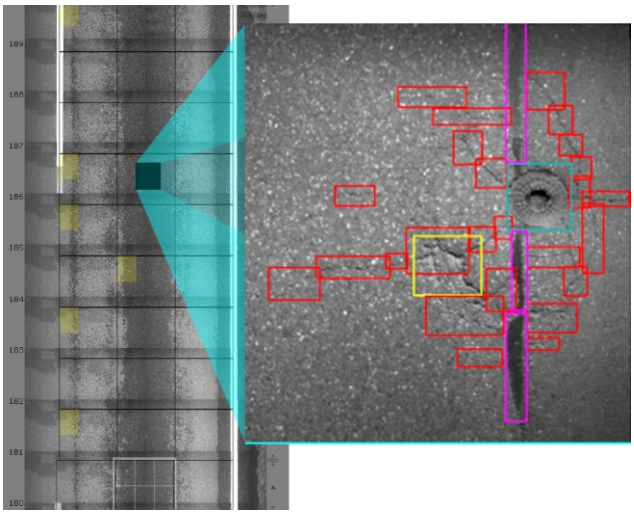

**Fig. 1: Left: Labeling as expected by German FGSV-regulation.
Right: fine labeling of different distress types with bounding boxes used in the GAPs dataset (Eisenbach, Stricker et al. 2017).**

1 Introduction

Public infrastructures are suffering from aging and therefore need frequent inspection. Distress detection and a solid management for maintenance are the key to guarantee their permanent availability. Therefore, condition acquisition and assessment must be applied to the whole road network of a country frequently. For Germany, this results in road surface

[1] This work has received funding from the German Federal Ministry of Education and Research as part of the ASINVOS project under grant agreement no. 01IS15036.

condition acquisition of about 13,000 km freeways and about 40,000 km federal highways[2] with high-speed measurement vehicles and the assessment of the extracted data afterwards. Since the initial acquisition in 1991, the relevant surface characteristics are re-determined periodically in a four year cycle. Since freeways are inspected in both directions on all lanes, a total of approximately 30,000 lane-km need to be covered annually.
Following German federal regulations, the surface characteristics have to be evaluated in terms of evenness, skid resistance, and substance condition. The surface characteristics evenness and skid resistance are mainly measured and analyzed automatically using laser sensors and sideway-force coefficient routine investigation machine technology (SCRIM). The substance condition is captured with camera systems and has to be evaluated by visual inspection of the recorded images. Current evaluation is done manually and therefore requires excessive manual labor (evaluation of 2,000 images per lane-km). Therefore, the time span between the actual inspection and the final evaluation may be up to several months. In the meantime, small damages, like cracks, can lead to substantial downtimes with a high impact for the population.
In the research project ASINVOS[3], we aim to automate this process to a high degree by applying machine learning techniques. The basic idea is to train a self-learning system with manually annotated data from previous inspections so that the system learns to recognize underlying patterns of distress. Once the system is able to robustly identify intact infrastructure, it can reduce the human amount of work by presenting only distress candidates to the operator. This helps to significantly speed up the inspection process and simultaneously reduces costs. Furthermore, inspection intervals can be reduced, which helps to remedy deficiencies in time.

2 Related Work

Automating the distress detection process has already attracted a lot of interest in the literature. Beside commercial all-in-one solutions like Dyntest, Pathway and the Applus System (Laurent, Hébert et al. 2012), whose internal algorithms are relatively unknown, a lot of different image processing approaches for automatic distress detection emerged in the literature during the last decade. The algorithms developed for evaluation of the pavement surface can be coarsely divided into three major groups: Crack image thresholding, patch-based classification, and depth-based algorithms.

2.1 Crack Image Thresholding

The first group of algorithms uses image processing methods to detect road distress structures that can be extracted by thresholding afterwards. Therefore, preprocessing algorithms are applied in order to reduce illumination artifacts. Under the assumption that crack structures can be identified as local intensity minima, thresholding in the image space is applied afterwards. The resulting crack image is further refined by morphological image operations and by searching for connected components. Approaches belonging to the aforementioned group are presented in (Huang, Zhang 2012), (Peng, Chao et al. 2015), (Xu, Wei et al. 2013), (Chambon, Moliard 2011), as well as in (Oliveira, Correia 2014), where the closed source but publicly available *CrackIT* toolbox is presented. This toolbox is included in the experimental evaluation of this paper. Other variants of that group use graph-based crack candidate analysis for further refinement (Zou, Cao et al. 2012), (Tang, Gu 2013),

[2] German Federal Road Research Institute (Bundesanstalt für Straßenwesen BASt): Erfassen und Bewerten von Oberflächeneigenschaften, ZEB – Zustandserfassung und -bewertung von Straßen (Acquisition and assessment of road surface characteristics), info flyer, 2016.
[3] ASINVOS: Assistierendes und Interaktiv lernfähiges Videoinspektiossystem für Oberflächen-strukturen am Beispiel von Straßenbelägen und Rohrleitungen (Interactive machine learning based monitoring system for pavement surface analysis)

(Fernandes, Ciobanu 2014), a *multi-scale curvelet* transform instead of a binary threshold (Wu, Sun et al. 2015), or gabor filters in order to find crack candidates (Salman, Mathavan et al. 2013).

2.2 Patch-based Classification

The algorithms of the second group apply different types of classifiers to patches of the image in order to extract crack or distress regions. Support vector machines (SVM) are commonly used. For example the classifier is applied to Histogram of Oriented Gradient (HOG) features (Kapela, Sniatala et al. 2015) or Local Binary Patterns (LBP) (Quintana, Torres et al. 2016), (Varadharajan, Jose et al. 2014). Neural networks are also applied in this domain, as for instance in (Zakeri, Nejad et al. 2013), which describe an approach that uses a *Multi Layer Perceptron* network in combination with frequency features and image histograms. Other approaches rely on neural networks that do not require a separate feature extraction. For instance (Shi, Gao et al. 2012) use a *Multilayer Autoencoder*, and (Zhang, Yang et al. 2016) use a *Convolutional Neural Network* for distress detection. The latter is included in the experimental evaluation of this paper.

2.3 Depth-based Algorithms

The third group of algorithms relies on depth information of the pavement. E.g. (Mertz 2011) proposes an algorithm for light section based crack detection, while (Yamada, Ito et al. 2013) describe a method that relies on a 2D laser range finder. An algorithm that applies crack detection on 3D point clouds is given in (Yu, Guan et al. 2015). Depth-based algorithms are excluded from our evaluation, since the used dataset includes image data only.

2.4 Datasets

Although, a lot of different methods have been presented so far, there is a lack of publicly available datasets that are of descent size and are recorded in a standardized way. To our best knowledge, there are only three different datasets available, all of which have less than 300 images in total (Chambon, Moliard 2011), (Oliveira, Correia 2014), (Zou, Cao et al. 2012). This hampers comparability, since most publications are using own datasets that have been generated using consumer cameras and are labeled in different ways.
In this paper, we use the GAPs dataset (German Asphalt Pavement Distress), presented in (Eisenbach, Stricker et al. 2017), which is the first free and publicly available pavement distress dataset[4] of a size, large enough to train high-performing deep neural networks. It provides high-quality images, recorded by a standardized process fulfilling German federal regulations, and detailed annotations (see Fig. 1). For the first time, this enables a fair comparison of alternative approaches in this field. We present a first evaluation of the state of the art in pavement distress detection on this dataset, followed by an analysis of the generalization ability of a deep neural network in the given road condition assessment domain. Therefore, we review the effectiveness of state of the art regularization techniques.

3 Evaluated Algorithms

Comparing different methods for distress detection was not possible in the past due to a lack of a sophisticated and publicly available benchmark dataset. Thus, we use the GAPs dataset, presented in (Eisenbach, Stricker et al. 2017), to compare state of the art methods for crack detection.

[4] The GAPs dataset is available at http://www.tu-ilmenau.de/neurob/data-sets-code/gaps/

3.1 Image Thresholding Approaches

We have used the publicly available CrackIT Toolbox (Oliveira, Correia 2014) as representative for the image thresholding based approaches. The toolbox provides different algorithms for image preprocessing and crack detection based on pattern classification techniques. It provides an image preprocessing stage that includes filters for context aware image smoothing and a dedicated lane line detection module to remove lane lines from the input image. Furthermore, the toolbox applies local image block-based normalization to reduce illumination dependence. Assuming low intensity values for crack pixels, the image is thresholded afterwards by analyzing the standard deviation of the intensity values of local image blocks. The resulting binary crack candidates image is refined afterwards by a connected-component algorithm in order to identify relevant cracks pixels.
The results of the toolbox are very sensitive to changes in the parameters used for the different processing steps. Therefore, the authors suggest to tune the parameters for the desired field of application, which we did on the GAPs training data.

3.2 Deep Learning Approaches

As representative for deep learning approaches, we have decided in favor of the promising *Convolutional Neural Network* (CNN) approach (Zhang, Yang et al. 2016) for road crack detection (in the following referred to as RCD net). (Zhang, Yang et al. 2016) presented a relatively small CNN with four blocks with alternating convolutional and max-pooling layers and two fully-connected layers (see Fig. 2), inspired by LeNet (LeCun, Bottou et al. 1998) architecture. By using only small filter sizes for the convolutional layers and only 48 filters per layer, the net is very compact and has only 0.3 M weights. Therefore, this network does not need much regularization. Only weight decay and dropout for the last fully-connected layer is used. The authors provided us their CAFFE (Jia, Shelhamer et al. 2014) code and their dataset, so we could check equality of results. We got a comparable but slightly better results on their data (the exact partitioning of their dataset could not be recovered). We then re-implemented the RCD using Keras (Chollet 2015) based on Theano (Theano Development Team 2016) and integrated it into our framework. By parameter tuning, again, we could slightly improve the net's performance.

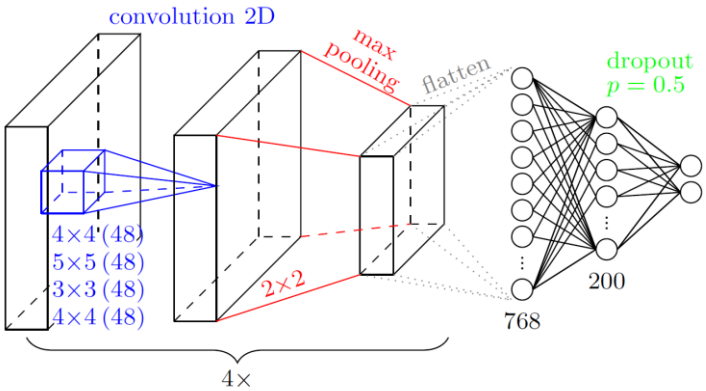

Fig. 2: RCD net (Zhang, Yang et al. 2016)

Since the RCD net is relatively small and does not represent modern CNN architectures, that are deeper, we conceptualize another CNN with eight convolutional layers, three max-pooling layers, and three fully connected layers (referred to as ASINVOS net in the following), and implemented it using Keras (Chollet 2015) based on Theano (Theano Development Team 2016). Its architecture is inspired by the ImageNet winning VGG-models (Simonyan, Zisserman 2015) (multiple units of two convolutional layers followed by one max-pooling layer) and AlexNet (Krizhevsky, Sutskever et al. 2012) (fully connected layers with softmax output). All neurons are ReLUs (Nair, Hinton 2010). For the exact architecture, with

filter sizes and dropout rates see Fig. 3. The ASINVOS net has 4.0 M weights. Thus, regularization is the key to perform well on unknown data.

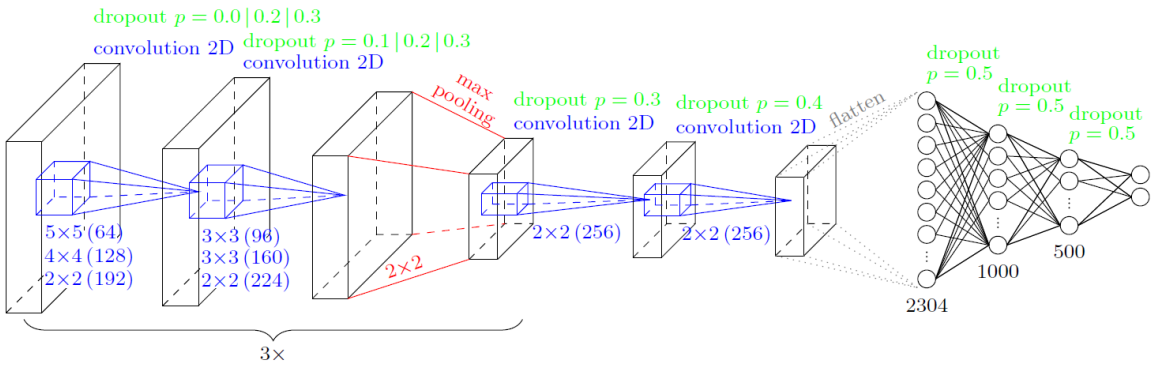

Fig. 3 ASINVOS net

3.2.1 Regularization

Dropout (Srivastava, Hinton et al. 2014) is known to be a very good regularization technique that avoids co-adaption and also improves generalization abilities. Therefore, our first approach is the extensive use of dropout for all layers except the input layer as sole regularization technique. Recently, this has been successfully applied in the person detection domain (Eisenbach, Seichter et al. 2016).
In recent years, batch normalization (Ioffe, Szegedy 2015) replaces dropout in modern neural network architectures. It is well known, that input normalization (zero-mean, unit variance) as a pre-processing step can improve neural network training. Batch normalization takes this idea even further and aims to remove the covariate shift from the internal activation of each subsequent layer. Thus, batch normalization can speed up training and often leads to a higher accuracy. We pursue two approaches: First, we replace dropout with batch normalization. Second, we use dropout in combination with batch normalization.
Since large weights may impair generalization abilities of a neural network, penalizing them is considered as a good regularization mechanism. We evaluated two approaches, namely weight decay (Moody, Hanson et al. 1995) and max-norm regularization (Srivastava, Hinton et al. 2014). To find appropriate hyper-parameters, we plotted the norm of weights in different layers over epochs. Once the network started to overfit, we determined the norms at this epoch and derived suitable hyper parameters.

3.2.2 Network Structure Variation

To evaluate, if the network structure can be improved, we set up two experiments: First, we evaluate the input coding. We recognized that the gray value histogram showed a distribution composed of three normal distributions (road paint, asphalt color, and shadows due to pavement structure). Therefore, we chose a topological input coding with three neurons per pixel.
Second, based on findings in (Szegedy, Vanhoucke et al. 2015), that each convolutional filter larger than 3x3 can be replaced by multiple 3x3-filters, we rearranged our structure. 5x5-filters are replaced by two successive 3x3-filters and the 4x4-filter by a 3x3-filter. (Szegedy, Vanhoucke et al. 2015) also found, that the performance improves when two successive 2x2-filters are replaced by one 3x3-filter. With this replacement, the modified ASINVOS net (referred to as ASINVOS-mod) used 3x3-filters only. Based on modern neural network architectures, e.g. (Szegedy, Vanhoucke et al. 2015), (Ioffe, Szegedy 2015), (He, Zhang et al. 2016), where pooling layers are replaced by filter map reducing convolutional layers with a stride of 2x2, we also replaced all pooling layers by such convolutional layers, that learn the reduction. We also followed these state of the art nets by replacing valid convolutions with size preserving convolutions by the use of zero-padding. The increased number of

layers and the use of size preserving convolutions result in an increase of parameters to 18.3 M. This increases the explanatory power of the neural network, but may negatively affect the generalization abilities.

4 Evaluation of State of the Art on GAPs Dataset

Fig. 4 shows exemplary visual results of the ASINVOS net applied on the GAPs dataset to show the capabilities of deep neural networks for distress detection on road surfaces. In the following, we compare the results with classical computer vision algorithms that are common in the research community.

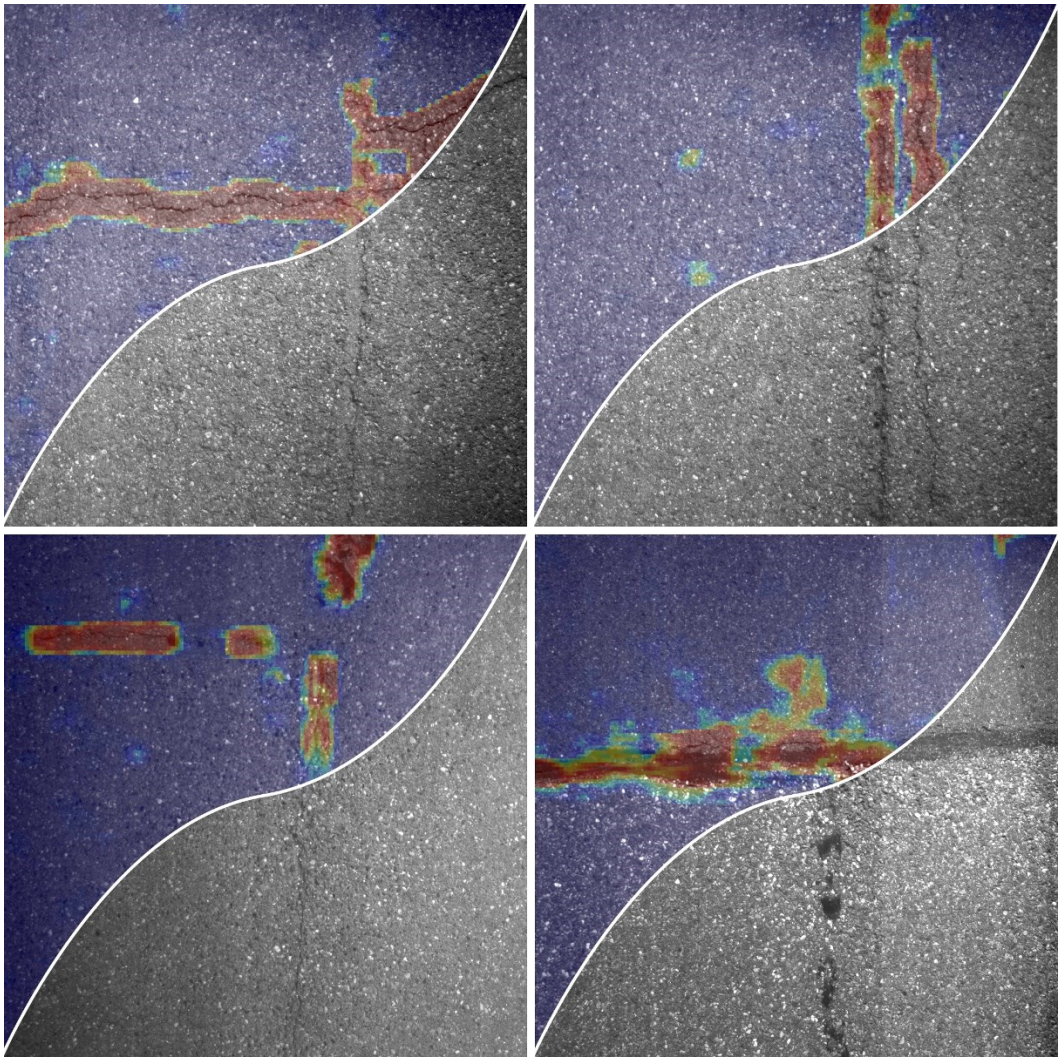

Fig. 4: Visual results of the ASINVOS net applied on the GAPs dataset

4.1 Evaluation Protocol

To evaluate the pure classification performance of the different algorithms instead of the detection performance, we extracted image patches. Using a sampling strategy that favors distress over intact road, we extracted a total of 4.9 M patches for training (approximately 1/8 distress), 200 k patches for validation (1/4 distress) and 1.2 M patches for testing purpose (better pavement condition, only 1/20 distress). All algorithms are evaluated on patches of size 64x64, except the RCD net that is additionally evaluated on 99x99-patches (drawn at the same positions), since this is its intended input size.

The dataset contains different types of road paint (arrows, lane lines, etc.). Since first evaluations with the CrackIT toolbox revealed, that it fails to handle road paint robustly, we excluded patches with road paint from the evaluation. We consider this a fairer comparison.
As performance measure we report the balanced error rate (BER), derived from the ROC curve, and the F_1-Score, derived from the Precision Recall diagram.

4.2 Regularization Evaluation

To be able to evaluate the generalization abilities of deep learning approaches, the GAPs dataset is partitioned such that the validation data is more similar to the training data than the test data. This was achieved by extracting the validation data from one of the roads that was also used for training, but from a different section of that road. In contrast, the test data are extracted from another German federal road with completely other road surface conditions (less distress).
Tab. 1 shows the validation and test results (table sections *original* and *regularization*). It can be seen, that the performance decreases from validation to test for all analyzed networks (original and all regularization). We conclude, that the test data differ more from training data than the validation data and thus, cause worse results. The generalization abilities are not sufficient to cope with unknown and significantly different data. None of the regularization techniques can cause a substantial improvement over the results achieved with dropout in the original ASINVOS net.

	validation		test	
algorithm	BER	F_1	BER	F_1
	original			
ASINVOS net $_{64x64}$	0.07124	0.8876	0.1209	**0.7246**
	regularization			
+ Batch Normalization	0.07551	0.8843	0.1294	0.6857
+ Batch Norm – Dropout	0.08317	0.8698	0.1184	0.6813
+ Weight Decay $_{(0.0002)}$	0.07709	0.8780	0.1358	0.6660
+ Max-Norm $_{(1.45)}$	0.07790	0.8784	0.1218	0.7012
	network structure			
ASINVOS-mod $_{64x64}$	**0.06518**	**0.8973**	0.1211	0.6707
Topo coding $_{3x64x64}$	0.07713	0.8785	0.1395	0.6518
	state of the art			
RCD net $_{64x64}$ (Zhang, Yang et al. 2016)	0.16240	0.7470	0.1511	0.6642
RCD net $_{99x99}$ (Zhang, Yang et al. 2016)	0.06882	0.8821	**0.1029**	0.7184
CrackIT (Oliveira, Correia 2014)	0.24930	0.7123	0.2645	0.4882

Tab. 1: GAPs dataset validation and test results. The best result achieved for each performance measure is highlighted in bold. All performance measures are chosen at the working points of their peak in the respective curve (ROC / PR).

According to the balanced error rate (BER), batch normalization without dropout achieved the best test results, closely followed by dropout only (ASINVOS net). Dropout in combination with batch normalization performs worst. Penalizing large weights with weight decay or max-norm regularization decreases the performance.
If the F_1 score is used to rate the performance (which we prefer), using dropout as sole regularization technique is the best choice, followed by batch normalization only. Combining both approaches is unfavorably. Again, both weight decay and max-norm regularization decreased the performance.
As a result, we propose to use either dropout or batch normalization for all layers, but neither weight decay nor max-norm for regularization.

4.3 Network Structure Evaluation

Tab. 1 (table sections *original* and *network structure*) show, that the chosen topological coding clearly performs worse than the pure gray value input coding (indicated by all performance measures).
We conclude that a CNN is able to learn the input coding by its own. Furthermore, an increase of input dimensions without an increase of information can decrease the performance significantly. Thus, it should be avoided.
An adaption of the network structure substantially improved the validation result, but the generalization abilities have dropped. For the completely different road in the test data the net with the modified structure (ASINVOS-mod) performs worse than the original ASINVOS net due to the increased number of weights. Therefore, we have a mixed result.
The modifications are promising, but to achieve better results on unknown and different data, the regularization must be improved.

4.4 Comparison to State of the Art

For a comparison of the state of the art see test results in Tab. 1 (ASINVOS net, ASINVOS-mod, RCD net, CrackIT). Analyzing the results, it is quite obvious, that deep learning approaches (ASINVOS net, ASINVOS-mod, RCD net) clearly outperform classical computer vision methods like CrackIT.
Furthermore, CrackIT is extremely sensitive to the chosen parameters, leading to bad generalization, as can be seen by the performance drop between similar data (validation) and significantly different data (test).
Surprisingly, the network with significantly more weights (ASINVOS net) performed only slightly better that the relatively small RCD net $_{99x99}$ (Zhang, Yang et al. 2016). However, when comparing results for equal input sizes of 64x64 pixels, the ASINVOS net outperforms the RCD net by far. We conclude, that 64x64-patches do not provide enough context information. Thus larger input patches should be the focus of future evaluations.

5 Conclusion

Since road condition acquisition and assessment is important to maintain a country's road network, millions of high-resolution road surface images are analyzed annually. In order to replace the extensive manual labor by an automatic distress detection system with high-performing deep neural networks, much data for training is needed. Therefore, we used GAPs, a freely available pavement distress dataset of a size, large enough to train modern deep neural networks. For each image, detailed distress annotations are available.
Thus, for the first time we were able to evaluate the state of the art in pavement distress detection in a meaningful way and reporting appropriate performance measures, namely balanced error rate (BER) and F_1 score. Summarized, only deep learning approaches were able to achieve satisfying detection results. Conventional computer vision approaches were beaten by a large margin. Furthermore, we analyzed the effectiveness of state of the art regularization techniques including dropout (Srivastava, Hinton et al. 2014), batch normalization (Ioffe, Szegedy 2015), max-norm regularization (Srivastava, Hinton et al. 2014) and weight decay (Moody, Hanson et al. 1995). The best generalization results were achieved using dropout only, followed by batch normalization only. Penalizing large weights decreased the performance.
With the extensive evaluation of deep neural networks on the GAPs dataset (Eisenbach, Stricker et al. 2017), using standardized images, we made a first step to automate the time and labor intensive process of analyzing millions of road surface images annually.

References

(Chollet 2015) F. Chollet, “Keras,” https://github.com/fchollet/keras, 2015.

(Chambon, Moliard 2011) S. Chambon and J.-M. Moliard, “Automatic road pavement assessment with image processing: review and comparison,” Int. Journal of Geophysics, 2011.

(Eisenbach, Seichter et al. 2016) M. Eisenbach, D. Seichter, T. Wengefeld, and H.-M. Gross, “Cooperative multi-scale convolutional neural networks for person detection,” in International Joint Conference on Neural Networks (IJCNN). IEEE, 2016, pp. 267–276.

(Eisenbach, Stricker et al. 2017) M. Eisenbach, R. Stricker, D. Seichter, K. Amende, K. Debes, M. Sesselmann, D. Ebersbach, U. Stöckert, H.-M. Groß, “How to Get Pavement Distress Detection Ready for Deep Learning? A Systematic Approach,” Submitted to Int. Joint Conf. on Neural Networks (IJCNN), 2017.

(Fernandes, Ciobanu 2014) K. Fernandes and L. Ciobanu, “Pavement pathologies classification using graph-based features,” in Int. Conf. on Image Processing (ICIP). IEEE, 2014, pp. 793–797.

(He, Zhang et al. 2016) K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), 2016.

(Huang, Zhang 2012) W. Huang and N. Zhang, “A novel road crack detection and identification method using digital image processing techniques,” in Int. Conf. on Computing and Convergence Technology (ICCCT). IEEE, 2012, pp. 397–400.

(Ioffe, Szegedy 2015) S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167, 2015.

(Jia, Shelhamer et al. 2014) Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding,” arXiv preprint arXiv:1408.5093, 2014.

(Kapela, Sniatala et al. 2015) R. Kapela, P. Sniatała, A. Turkot, A. Rybarczyk, A. Pozarycki, P. Rydzewski, M.Wyczałek, and A. Błoch, “Asphalt surfaced pavement cracks detection based on histograms of oriented gradients,” in Int. Conf. on Mixed Design of Integrated Circuits & Systems, 2015, pp. 579–584.

(Krizhevsky, Sutskever et al. 2012) A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Advances in neural information processing systems (NIPS), 2012, pp. 1097–1105.

(Laurent, Hébert et al. 2012) J. Laurent, J. F. Hébert, D. Lefebvre, and Y. Savard, “Using 3d laser profiling sensors for the automated measurement of road surface conditions,” in RILEM Int. Conf. on Cracking in Pavements. Springer, 2012, pp. 157–167.

(LeCun, Bottou et al. 1998) Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

(Mertz 2011) C. Mertz, “Continuous road damage detection using regular service vehicles,” in ITS World Congress, 2011.

(Moody, Hanson et al. 1995) J. Moody, S. Hanson, A. Krogh, and J. A. Hertz, “A simple weight decay can improve generalization,” Advances in neural information processing systems, vol. 4, pp. 950–957, 1995.

(Nair, Hinton 2010) V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in Int. Conf. on Machine Learning (ICML), 2010, pp. 807–814.

(Oliveira, Correia 2014) H. Oliveira and P. L. Correia, “Crackit – an image processing toolbox for crack detection and characterization,” in Int. Conf. on Image Processing (ICIP). IEEE, 2014, pp. 798–802.

(Peng, Chao et al. 2015) L. Peng, W. Chao, L. Shuangmiao, and F. Baocai, “Research on crack detection method of airport runway based on twice-threshold segmentation,” in Int.

Conf. on Instrumentation and Measurement, Computer, Communication and Control (IMCCC). IEEE, 2015, pp. 1716–1720.

(Quintana, Torres et al. 2016) M. Quintana, J. Torres, and J. M. Menéndez, “A simplified computer vision system for road surface inspection and maintenance,” Transactions on Intelligent Transportation Systems (ITS), vol. 17, no. 3, pp. 608–619, 2016.

(Salman, Mathavan et al. 2013) M. Salman, S. Mathavan, K. Kamal, and M. Rahman, “Pavement crack detection using the gabor filter,” in Int. Conf. on Intelligent Transportation Systems (ITSC). IEEE, 2013, pp. 2039–2044.

(Shi, Gao et al. 2012) L. Shi, C. Gao, and J. Zhang, “Pavement distress image recognition based on multilayer autoencoders,” in Int. Conf. on Artificial Intelligence and Computational Intelligence (AICI). Springer, 2012, pp. 666–673.

(Simonyan, Zisserman 2015) K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Int. Conf. on Learning Representations (ICLR), 2015, pp. 1–14.

(Srivastava, Hinton et al. 2014) N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, “Dropout: a simple way to prevent neural networks from overfitting.” Journal of Machine Learning Research, vol. 15, no. 1, pp. 1929–1958, 2014.

(Szegedy, Vanhoucke et al. 2015) C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” arXiv preprint arXiv:1512.00567, 2015.

(Tang, Gu 2013) J. Tang and Y. Gu, “Automatic crack detection and segmentation using a hybrid algorithm for road distress analysis,” in Int. Conf. on Systems, Man, and Cybernetics (SMC). IEEE, 2013, pp. 3026–3030.

(Theano Development Team 2016) Theano Development Team, “Theano: A Python framework for fast computation of mathematical expressions,” arXiv:1605.02688, 2016.

(Varadharajan, Jose et al. 2014) S. Varadharajan, S. Jose, K. Sharma, L. Wander, and C. Mertz, “Vision for road inspection,” in Winter Conf. on Applications of Computer Vision (WACV). IEEE, 2014, pp. 115–122.

(Wu, Sun et al. 2015) G. Wu, X. Sun, L. Zhou, H. Zhang, and J. Pu, “Research on crack detection algorithm of asphalt pavement,” in Int. Conf. on Information and Automation (ICIA). IEEE, 2015, pp. 647–652.

(Xu, Wei et al. 2013) K. Xu, N. Wei, and R. Ma, “Pavement crack image detection algorithm under nonuniform illuminance,” in Int. Conf. on Information Science and Technology (ICIST). IEEE, 2013, pp. 1281–1284.

(Yamada, Ito et al. 2013) T. Yamada, T. Ito, and A. Ohya, “Detection of road surface damage using mobile robot equipped with 2d laser scanner,” in Int. Symposium on System Integration (SII). IEEE/SICE, 2013, pp. 250–256.

(Yu, Guan et al. 2015) Y. Yu, H. Guan, and Z. Ji, “Automated detection of urban road manhole covers using mobile laser scanning data,” Transactions on Intelligent Transportation Systems (ITS), vol. 16, no. 6, pp. 3258–3269, 2015.

(Zakeri, Nejad et al. 2013) H. Zakeri, F. M. Nejad, A. Fahimifar, A. D. Torshizi, and M. F. Zarandi, “A multi-stage expert system for classification of pavement cracking,” in Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS). IEEE, 2013, pp. 1125–1130.

(Zhang, Yang et al. 2016) L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, “Road crack detection using deep convolutional neural network,” in Int. Conf. on Image Processing (ICIP). IEEE, 2016, pp. 3708–3712.

(Zou, Cao et al. 2012) 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.