Sheet Metal 2023 Materials Research Forum LLC
Materials Research Proceedings **25** (2023) 371-378 https://doi.org/10.21741/9781644902417-46

Computer vision for industrial defect detection

Johannes Landgraf[1,a *], Moritz Kompenhans[1,b *], Tassilo Christ[1,c *], Theresa Roland[2,d] and Adrian Heinig[3,e]

[1]d-fine GmbH, An der Hauptwache 7, 60313 Frankfurt, Germany

[2]voestalpine Metal Forming GmbH, voestalpine Straße 1, 4020 Linz, Austria

[3]voestalpine Automotive Components Dettingen GmbH & Co. KG, Daimlerstraße 29, 72581 Dettingen/Erms, Germany

[a]Johannes.Landgraf@d-fine.de, [b]Moritz.Kompenhans@d-fine.de, [c]Tassilo.Christ@d-fine.de [d]Theresa.Roland@voestalpine.com, [e]Adrian.Heinig@voestalpine.com

Keywords: Artificial Intelligence, Visual Inspection, Defect Detection

Abstract. Despite the continuous progress in computer vision, its application to many industrial tasks like the detection and size measurement of non-trivial defects is still a demanding problem. In this paper, typical challenges, workflows, and key performance indicators are discussed and the application of AI-based Semantic Image Segmentation methods is demonstrated to the detection of minute damages on metal surfaces using the d-fine vision toolbox. A performance improvement for a public data set above prior results is reported and the successful transfer of the approach to real-world sheet metal parts produced by voestalpine Automotive Components Schmölln GmbH is shown.

Introduction

Computer vision is an essential enabler of the end-to-end digitalization of production processes. Use cases like object tracking, pattern recognition, event detection or visual servoing (vision-based robot control) are among the fundamentals of intelligent, autonomous, and self-optimizing production processes [1]. Industrial computer vision can detect problems in an autonomous, continuous, and reliable manner and supply feedback into production processes and machines directly, e.g., in case of high failure rates due to incorrect machine calibration.

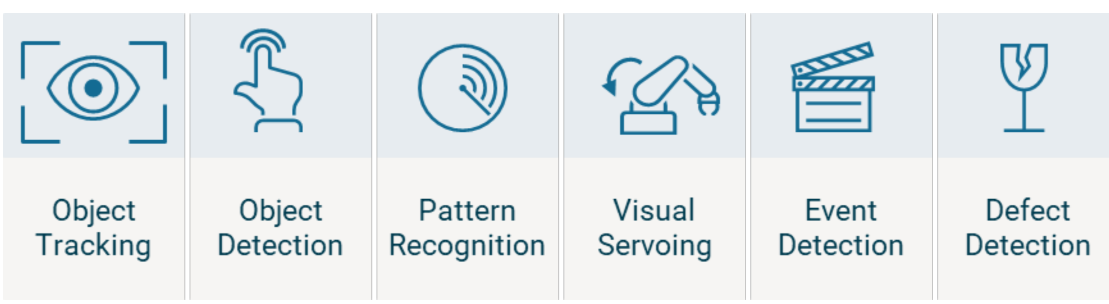


Fig. 1: Industrial computer vision applications

Virtually all modern applications of computer vision are based on neural networks with so called convolutional architectures [2] playing the most prominent role. When applied properly, this technology allows to automate analyses and processes that hitherto required a human's ability to interpret data. The following is an overview of the most important steps and the challenges involved.

Data quality. Stable imaging conditions and good image quality are crucial for the success of computer vision. Deficiencies can only be partially compensated for by smart preprocessing methods and image resolution and lighting conditions must be carefully controlled.

Preprocessing. In most computer vision use-cases, the available data is imbalanced. The training data contain only a small fraction of NOK-parts and do not cover possible damage manifestations. To generate a more representative image base and to at least partially remedy these deficiencies algorithmic preprocessing can generate additional synthetic images (see e.g. [3] for an overview) with altered lighting and noise conditions, aspect ratios and sub-crops.

Deep Learning architecture. (Convolutional) Neural networks (CNNs) are inspired by the connectivity structure of neurons in the mammalian brain but should not be taken to represent a cognitive process. Their performance depends on a suitably chosen architecture, the quality of the input data and the careful tuning of the training strategy. [4] provides an excellent introduction into the architectural principles and the way CNNs build a hierarchical representation of image content.

Postprocessing. Algorithmic segmentation of the image into regions of different characteristics is not sufficient in productive applications. The results must be transformed into a suitable format for (quality-) experts and decision makers. This includes both a visual representation of the identified damages and performance dashboards for error rates or product quality assessments.

Benchmarking computer vision systems

Four practical requirements have proven crucial for a successful defect detection scheme, namely: robustness, data efficiency, accuracy and performance.

Robustness. Computer vision systems are affected by lighting situation and perspective and are negatively impacted by contamination and reflections. A good system should be robust with respect to such real-life disturbances and deviations from ideal conditions.

Data efficiency. Defect types, part shapes as well as location and orientation of the components to be inspected may vary between images. An ideal vision tool can be trained on a limited set of components with few images taken from only a subset of defect types and can still apply its "cognitive" abilities to new and hitherto unseen parts, defect types and perspectives within a given production process.

Accuracy. To ensure that the system can take over tedious and repetitive QA tasks, detected errors must coincide well with defect labels provided by a human supervisor. At the same time False Positives (errors reported without an actual error being present) and False Negatives (errors missed by the AI) must be reduced as much as possible.

Performance. Detection accuracy does not provide a competitive advantage if data throughput is low due to computational complexity. In particular for real-time applications on the shopfloor the achievable processing rate is a critical quantity.

Related work

Deep learning and computer vision provide robust solutions to the tasks shown in Fig. 1, including object tracking [5], object recognition [6] and visual servoing [7].

Vision-based quality inspection offers the possibility to perform non-destructive inspection processes in industry. Huang et al. [8] provide an overview of inspection tasks in the semi-conductor industry that can potentially be automated. Defect detection in an industrial context using CNNs on X-ray images has been investigated in [9]. The detection of cracks is part of many research projects. Detecting cracks in concrete using deep learning has been studied by [10]. Surface inspection is an important part of industrial quality assurance. The authors of [11] investigate the use of deep learning for this application and present essential design requirements.

In [12], U-Nets were presented to segment anomalies in biomedical data. U-Nets comprise a contracting and an expanding data path (see left side of Fig. 2 for an example) and are also used in this work. Their output is a mask with probabilities for each pixel indicating whether the pixel is or is not part of a labelled region (e.g., "DEFECT" / "OK"). The horizontal and vertical dimensions of the output are identical to the input dimensions of the image such that the results can be visually overlaid for ease of interpretation later. [13] has achieved state of the art results using U-Nets for magnetic tile inspection.

Deep learning enables very good detection rates but requires very large amounts of data. Especially in the industrial environment it can be difficult to generate and annotate these correctly for training.

In [14] the authors address the data efficiency by applying networks trained in other domains to new data sets (transfer learning). In [15] a special loss function ("focal loss") for training is introduced that emphasizes hard to learn image areas to increase network performance. Finally, in [16] a special sampling method is proposed that increases the weight of entire images that contain hard to learn content.

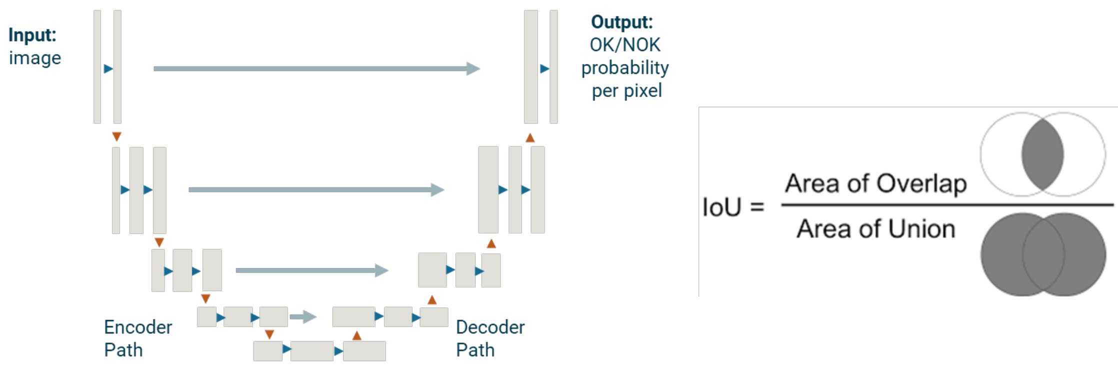


Fig. 2: Left: A simplified U-shaped neural network architecture. Right: Intersection over Union of a training mask and a network output mask as metric for the evaluation of image segmentation methods

Approach

The objective of this work is to design a defect detection system that provides for robustness, data efficiency and computation performance while achieving state-of-the-art results for accuracy and to demonstrate its applicability not only to laboratory data but to real-world images from an industry-grade production process. As described in [15], the training of semantic segmentation algorithms suffers twofold from data imbalance: First, "OK"-parts are overrepresented. Second, the defects to be detected are very small compared to the size of both the component itself and the image on which they are to be found. For this purpose, this paper investigates the visual inspection capabilities of a U-Net trimmed for data efficiency in combination with a modified focal loss function, emphasizing the hard to train image regions and hiding out the component background.

To cope with the small number of input images, the training data set is enriched using various geometric and photometric data augmentations and innovative sampling strategies that enhance the "NOK"-regions' fraction in the data set by transferring damaged regions between images.

The used U-Net architecture includes five skip connections that copy input data from early layers to later layers. This allows the model to keep local information together and prevents information loss. Careful tuning of the network details ensures that the system is already highly data efficient at the architectural level.

Finally, a post-processing routine incorporating domain-specific knowledge about typical defect sizes enhances the performance of the semantic segmentation approach and suppresses false positives. This comprises i.a. a threshold set upon the probability level with which the network considers a pixel to be part of a defective area. To finally measure the recognition accuracy, it is

Materials Research Proceedings **25** (2023) 371-378 https://doi.org/10.21741/9781644902417-46

common practice to compare the manually provided defect mask and the output mask of the network. Their agreement is usually quantified as the ratio between the intersection of the two areas and their union ("IOU-metric") [17] – see right side of Fig. 2.

Example use case: ball screw drives

To demonstrate the performance of the image segmentation and defect detection pipeline and to determine quantitative KPIs, a publicly available data set of ball screw drive images is used and the workflows and the results obtained are discussed.

Data set. The data set "Industrial Machine Tool Element Surface Defect Dataset" [18] by the Karlsruhe Institute of Technology (KIT) contains images of ball screw drives (BSDs). BSDs are roller bearings used for translating rotary motion into linear motion and "one of the most wear-prone machine tools" [19]. The data set consists of 394 images each with at least one defect, which was labelled by a human expert using a polygonal mask along the boundary of the damage region (see Fig. 3). The defect regions are of different sizes and the images show soiling as it is typical for industrial manufacturing environments.

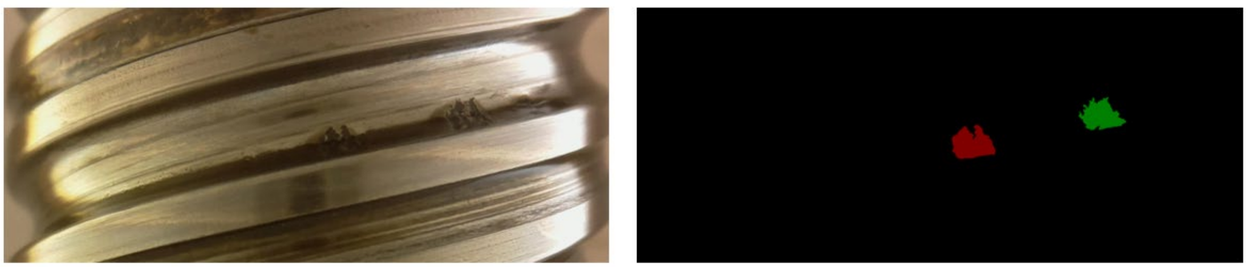


Fig. 3: Image taken from the KIT ball screw drive data set showing a defect (left) and the corresponding defect mask (right).

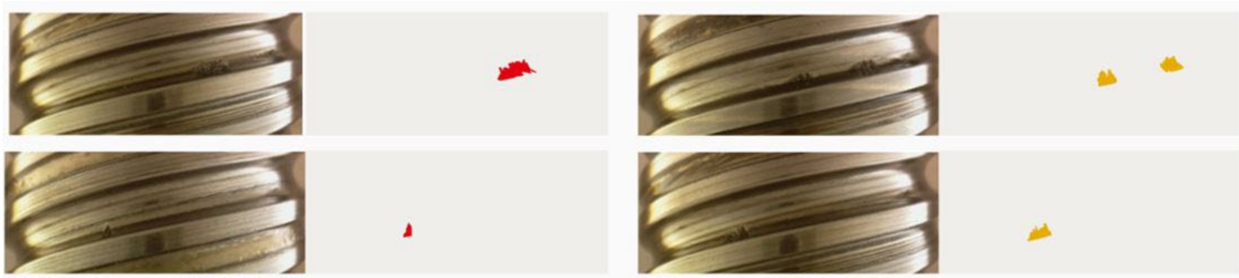
Data split and preprocessing. The data is split into subsets as proposed by the authors [18]. The training proceeds in so called "epochs" during with the system tries to optimize recognition results on the training data set. The performance is measured independently after each training epoch using the validation data set. The final performance KPIs reported below however are determined on the test data set which has never been presented to the network before.

Training details. The system has been trained on Google Cloud Platform for 62 hours using a Nvidia V100 GPU with 16 GB RAM.

Results. Fig. 4 shows the results achieved when applying the network to previously unseen error samples. Defects on the test data set are detected with a mean IoU of 45.3%, which is a noticeable improvement compared to the previously published value of 31.6% (see [19]). For all practical applications it is necessary to decide at some point whether a region with a "defect"-label is to be classified as an actual defect or to be rejected as a false positive candidate (as may be done with very small defect regions in practice). Fig. 5, shows the classification results of the approach for a set of 60 selected test-images. Each image is represented by a dot on a two-dimensional map, with the horizontal and vertical extension of the damaged region (measured relative to the total image size) used as coordinates. Orange dots indicate correctly detected damages, blue ones indicate missed ones. The toolchain presented in this paper detects even small defects more reliably when compared to results shown in [20]. The undetected defects have a small relative size compared to the image's dimension. A higher resolution or an inspection using image sub-crops might help to further improve the performance.

Materials Research Proceedings **25** (2023) 371-378 https://doi.org/10.21741/9781644902417-46

Fig. 4: Left: Training images with damaged regions - defect masks provided during the training phase are shown in red; Right: Separate validation with unseen data - damage masks identified



by the trained network are shown in amber.

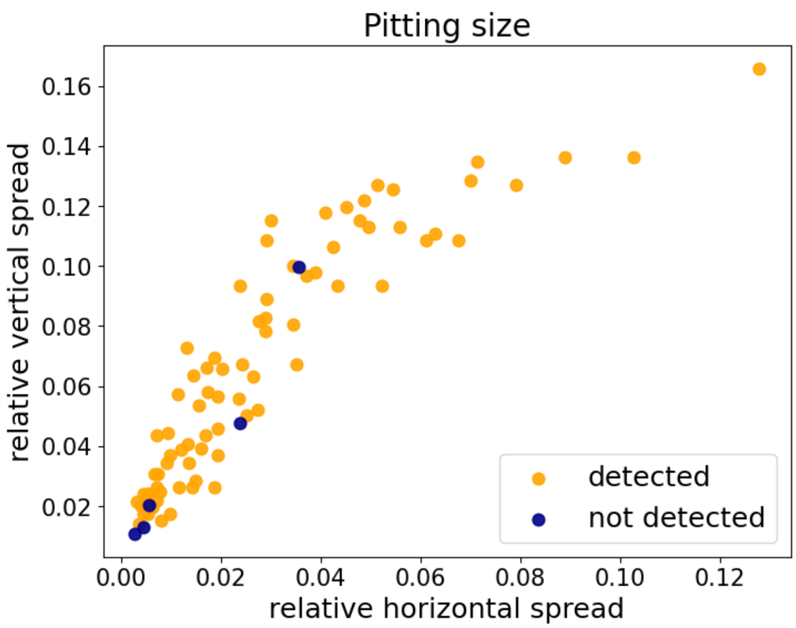


Fig. 5: Detected and missed defects and their relative horizontal and vertical spread in the image. The solution almost perfectly detects all but the most minute damages.

Application to sheet metal components

To demonstrate the feasibility of defect detection outside the lab with real-world shop-floor data with comparable challenges, the method is applied to several sheet metal parts that belong to the automotive chassis. The parts were produced and photographed in the press plant of voestalpine in Schmölln, Germany.

Data set and enrichment. The images for this analysis were taken on a separate inspection island from a single camera perspective under reproducible and calibrated lighting conditions. The dataset contains 179 images taken from 3 different sheet metal component shapes. 107 images were used for training, 36 for validation and 36 for testing. The high resolution of the images and the need to detect also small defects mandate that the defect recognition is applied to image sub-crops to ensure reliable detection. Also, the use of image cropping allows to increase the proportion of damaged regions during training and thus to equalize the ratio between “OK” and “NOK” images in the data set.

Materials Research Proceedings **25** (2023) 371-378 https://doi.org/10.21741/9781644902417-46

Foreground segmentation and damage detection. To find the damaged sections, the AI-based segmentation method has been applied twofold: First it masks the component part (“foreground”) and separates it from the uninformative image “background”. The images on the left of Fig. 6 show typical masks resulting from this step with reddish pixels indicating the image sections that the AI ascribes to the part. This step also works reliably when new shapes are presented to the background segmentation step. Damage detection itself proceeds subsequently. Typical results for images from the test data set are shown in the right side of Fig. 6. Here pixels to which the AI ascribes a “damaged” label are marked in green. Pixels in blue are “missed pixels”: They have been labelled manually as “damage region” before the training phase but have not been recognized by the AI.

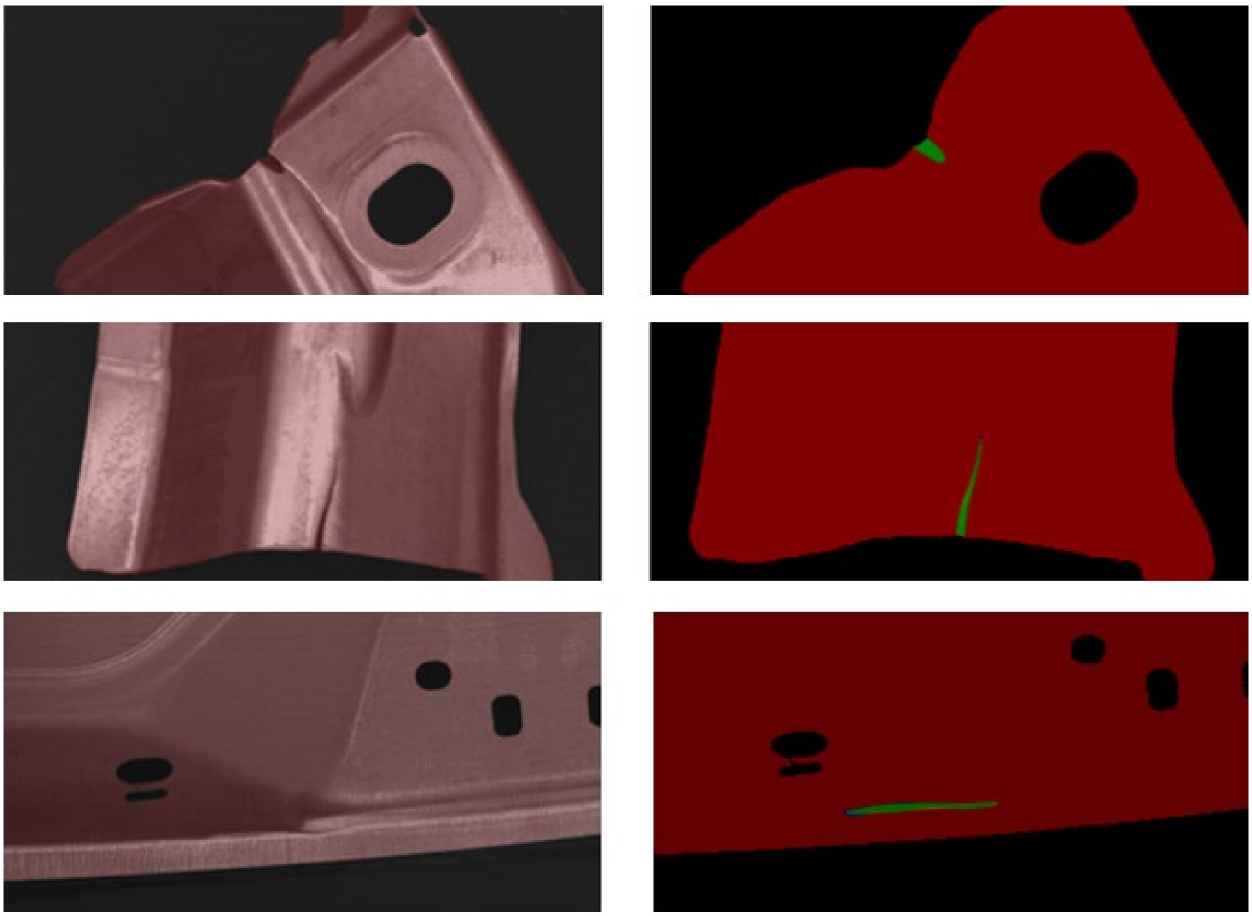


Fig. 6: A sheet metal components inspected with the two-stage segmentation approach. Left: foreground separation; right: damage detection

Results. While applying the methodology allows to train a neural network with the very small data set, a statistical analysis of the results would not provide meaningful and reliable performance numbers. As described in the review paper [21], crack detection algorithms are usually trained on data sets that are about 100 times larger in scale. To provide quantitative, statistically sound results a data set of at least five times the one available at present would be needed. A qualitative analysis shows however, that the trained network detects damages also in the test data set if they are optically similar to defects presented during training. Using sufficiently well resolved image sub-crops, also very small defects can be detected that are hardly visible to the human eye on a full-scale image. As the training of the damage detection step is performed on image sub crops it should be agnostic with respect to the specific component shape under inspection. This is an important feature for real life applications because it substantially reduces the need to retrain the AI for new shapes and different camera perspectives. However, a larger data set is required to verify this unambiguously.

Summary

Using convolutional neural networks, an AI-based image segmentation method is developed which detects even small defects on sizable metallic surfaces. The method is data efficient in that it requires only a very limited amount of training data and thus greatly reduces the labelling effort required before training. It generalizes well to unseen data and outperforms prior results on a publicly available data set. The developed preprocessing, training and postprocessing cascade is applied to real world images from a sheet metal plant. While statistical performance KPIs can not be given due to the limited amount of available data promising qualitative results are achieved. A fully quantitative evaluation of the performance KPIs on a larger data set is required next to develop the system into a flexible, scalable AI system for end-of-line quality control for sheet metal parts and other components with metallic surfaces.

Acknowledgement

The authors would like to thank the staff of voestalpine Automotive Components Schmölln GmbH for explanation of their image acquisition setup and the provision of the data set and the voestalpine data science team for fruitful technical exchange.

References

[1] B. Vogel-Heuser, T. Bauernhansl, M. ten Hompel, eds., Handbuch Industrie 4.0 Bd.4, Allgemeine Grundlagen., Springer, 2017. https://doi.org/10.1007/978-3-662-53254-6

[2] Y. Lecun, Y. Bengio, Convolutional networks for images, speech, and time-series, in: M.A. Arbib (Ed.), The Handbook of Brain Theory and Neural Networks, MIT Press, 1995.

[3] C. Shorten, T.M. Khoshgoftaar, A survey on Image Data Augmentation for Deep Learning, J. Big Data. 6 (2019) 60. https://doi.org/10.1186/s40537-019-0197-0

[4] I. Goodfellow, Y. Bengio, A. Courville, Deep learning, MIT press, Cambridge, MA, USA, 2016.

[5] S. Mojtaba Marvasti-Zadeh, L. Cheng, H. Ghanei-Yakhdan, S. Kasaei, Deep Learning for Visual Tracking: A Comprehensive Survey, ArXiv E-Prints. (2019) earXiv:1912.00535.

[6] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once: Unified, Real-Time Object Detection, ArXiv E-Prints. (2015) earXiv:1506.02640. https://doi.org/10.1109/CVPR.2016.91

[7] Q. Bateux, E. Marchand, J. Leitner, F. Chaumette, P. Corke, Visual Servoing from Deep Neural Networks, ArXiv E-Prints. (2017) earXiv:1705.08940.

[8] S.-H. Huang, Y.-C. Pan, Automated visual inspection in the semiconductor industry: A survey, Computers in Industry. 66 (2015) 1–10. https://doi.org/10.1016/j.compind.2014.10.006

[9] M. Ferguson, R. Ak, Y.-T.T. Lee, K.H. Law, Detection and Segmentation of Manufacturing Defects with Convolutional Neural Networks and Transfer Learning, ArXiv E-Prints. (2018) earXiv:1808.02518. https://doi.org/10.1520/SSMS20180033

[10] D. Einarson, D. Mengistu, Deep Learning Approaches for Crack Detection in Bridge Concrete Structures, in: 2022 International Conference on Electronic Systems and Intelligent Computing (ICESIC), 2022: pp. 7–12. https://doi.org/10.1109/ICESIC53714.2022.9783576

[11] D. Martin, S. Heinzel, J. Kunze von Bischhoffshausen, N. Kühl, Deep Learning Strategies for Industrial Surface Defect Detection Systems, ArXiv E-Prints. (2021) earXiv:2109.11304. https://doi.org/10.24251/HICSS.2022.146

[12] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, ArXiv E-Prints. (2015) earXiv:1505.04597. https://doi.org/10.1007/978-3-319-24574-4_28

[13] Y. Huang, C. Qiu, Y. Guo, X. Wang, K. Yuan, Surface Defect Saliency of Magnetic Tile, in: 2018 IEEE 14th International Conference on Automation Science and Engineering (CASE), 2018: pp. 612–617. https://doi.org/10.1109/COASE.2018.8560423

[14] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, ArXiv E-Prints. (2014) earXiv:1411.1792.

[15] T.-Y. Lin, P. Goyal, R. Girshick, K. He, P. Dollár, Focal Loss for Dense Object Detection, ArXiv E-Prints. (2017) earXiv:1708.02002. https://doi.org/10.1109/ICCV.2017.324

[16] A. Shrivastava, A. Gupta, R. Girshick, Training Region-based Object Detectors with Online Hard Example Mining, ArXiv E-Prints. (2016) earXiv:1604.03540. https://doi.org/10.1109/CVPR.2016.89

[17] T.T. Tanimoto, An Elementary Mathematical Theory of Classification and Prediction, International Business Machines Corporation, 1958.

[18] T. Schlagenhauf, M. Landwehr, J. Fleischer, Industrial Machine Tool Element Surface Defect Dataset, 2021. https://doi.org/10.1016/j.dib.2021.107643

[19] A. Haberkern, Leistungsfähige Kugelgewindetriebe durch Beschichtung, Universität Karlsruhe, Institut für Werkzeugmaschinen und Betriebstechnik, 1998

[20] T. Schlagenhauf, M. Landwehr, Industrial machine tool component surface defect dataset, Data in Brief. 39 (2021) 107643. https://doi.org/10.1016/j.dib.2021.107643

[21] S.D. Nguyen, T. Son, T. Van Phuc, H. Lee, M. Piran, V.P. Le, Deep Learning-Based Crack Detection: A Survey, International Journal of Pavement Research and Technology. (2022). https://doi.org/10.1007/s42947-022-00172-z