d-fine

Computer Vision for Industrial Defect Detection

Image segmentation with the d-fine Vision-AI toolstack

analytical. quantitative. tech.

July 2022

Inhalt

Computer Vision for Industrial Defect Detection, July 2022

© d-fine GmbH

1. Industrial computer vision — Page 3

2. Semantic image segmentation for defect detection and quality inspection — Page 3

3. d-fine's computer vision software solution — Page 5

4. Example use case: ball screw drives — Page 6

5. Working together — Page 8

01.

Industrial computer vision

Computer vision is an essential enabler of digital production processes and industry 4.0 innovations. Use cases like object tracking, pattern recognition, event detection or visual servoing are among the fundamentals of intelligent, autonomous and self-optimizing production processes (see e.g., Vogel-Heuser et al.)[1]. Where nowadays experts are needed to interpret production results (e.g., visual quality checks), industrial computer vision can detect problems in an autonomous, continuous, and reliable manner. Besides merely assisting experts, the automatic analysis allows to supply feedback into production processes and machines directly, e.g., in case of high failure rates due to incorrect machine calibration.

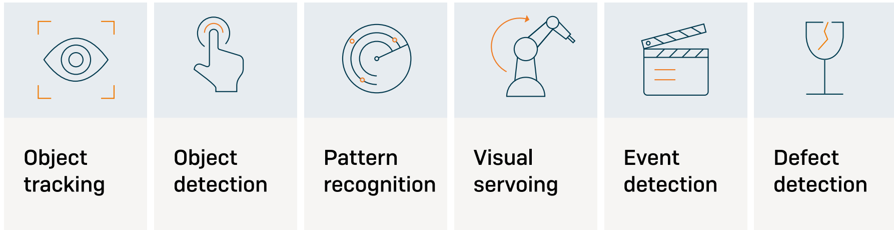

Figure 1: Industrial computer vision applications

Virtually all modern applications of computer vision are based on neural networks with so called convolutional architectures (see e.g., LeCun et al.)[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 interprete data. Despite the enormous increase in computational power and available data as well as algorithmic advances in recent years, the effort to adapt the concept to use cases of more than moderate complexity is still substantial and "out of the box applications" tend to fail for non-standard tasks.

02.

Semantic image segmentation for defect detection and quality inspection

In this paper, we focus on a computer vision task which not only has applications in manufacturing, but also e.g., in autonomous vehicle navigation: Semantic image segmentation aims to not only identify image contents but to identify the boundaries of objects on a per-pixel basis. One of the most obvious applications in manufacturing is the detection and size measurement of defects during quality inspection. Several factors are crucial to implement the method successfully for non-trivial defect detection applications. A robust and reliable detection pipeline builds not only upon a suitably chosen and implemented neural network but requires stable imaging conditions, advanced pre- and post-processing tools and most of all a firm understanding of the manufacturing process.

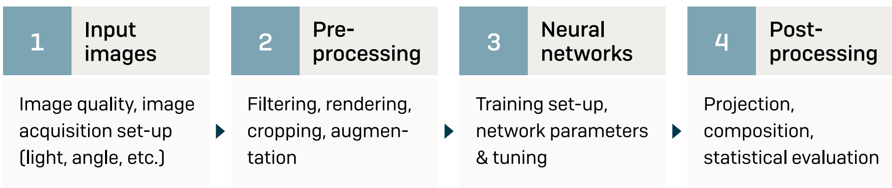

Figure 2: Processing steps of an industrial image detection scheme

1 Vogel-Heuser et al. (2016): Handbuch Industrie 4.0 Bd.1 Springer Vieweg

2 LeCun, Yann, and Yoshua Bengio. "Convolutional networks for images, speech, and time series." The handbook of brain theory and neural networks 3361, no. 10 (1995): 1995..

Input images

It is a common misconception that given enough computing power and data, vision AI architectures can achieve super-human performance even on bad data. On the contrary, stable imaging conditions and good image quality are crucial for the success of computer vision. Deficiencies can only be partially compensated for by smart pre-processing methods. Image resolution and lighting conditions have a major impact on the segmentation results that can be achieved by the neural network and must be carefully controlled.

- Image quality (resolution, contrast, reflection, illumination, contamination)
- Defect types (cracks, dents, bumps, deformations)
- Defect areas (number of pixels, size in relation to total component)
- Dataset (labelled/unlabelled images, total number of images for training)

Figure 3: Image and dataset characteristics with strong influence upon the overall system performance

Pre-processing

In most computer vision use cases, the available data is first of all imbalanced - with e.g., only a small fraction of NOK-parts contained in the training data, second does not represent all possible manifestations of damage characteristics and third suffers to some unavoidable extent from instable imaging conditions. Algorithmic pre-processing is intended to remedy these and other deficiencies by augmentation of entire datasets - e.g., via generation of additional synthetic images with altered lighting and noise conditions, aspect ratios and subcrops. Thereby, a more comprehensive and more representative image base for the training of the neural network is generated.

Neural networks

(Convolutional) Neural networks (CNNs) are inspired by the connectivity structure of neurons in the mammalian brain. Still, they should not be taken to represent a cognitive process, but merely be regarded as a mathematical abstraction of a hierarchical pattern matching process. (Goodfellow et al. 2016)[3] provides an excellent introduction into the architectural prinicples and the way CNNs build a hierarchical representation of image content.

The use case-specific performance of the network depends on a suitably chosen architecture, the quality of the input data and a set of parameters that define the training strategy (frequently called "hyperparameters"). In the example case discussed below a specially crafted parametrization of the penalty for false predictions during training (called the "loss fuction") plays an important role in achieving good segmentation performance.

Post-processing

Using AI to just predict the desired outcome (in our case the segmentation of the image into regions of different characteristics) is not sufficient in productive applications. The results of the neural network 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.

3 Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

03. d-fine's computer vision software solution

In our clients' projects, four practical requirements have proven crucial for a successful defect detection scheme, namely: robustness, data efficiency, accuracy and performance.

Robustness
Image quality depends not only on the lighting situation but also on the perspective. It is negatively impacted by contamination and in case of metallic surfaces, by reflections. Moreover, the location and orientation of the components to be inspected may vary between images. Clients ideally need a solution that nevertheless only requires a single training run and can be used on multiple machines or at least covers multiple perspectives within a given line or process. It is thus essential that the network is also robust to background changes as well as changes in camera position. Over time, we have developed a solution that achieves good results even with geometrically distorted image data and is capable of handling changes in orientation of inspected parts and camera perspective.

Data efficiency & generalization
An ideal algorithmic quality control system can be trained on a limited set of components with few images taken from only a subset of selected defect types and still apply the acquired "cognitive" abilities to new and hitherto unseen parts and to other defect types without loss of recognition performance.

Our solution is specifically built to deliver upon these needs and has proven to be data efficient in the sense that a comparatively small dataset suffices to achieve the above-described ability to generalize. Using tailored sampling and augmentation methods, we extract the maximum of relevant information from even the smallest defect areas such that also images of unknown components can be processed.

Accuracy
To measure the AI's recognition accuracy, one must quantify how well the detected errors coincide with a defect label provided by a human operator. Using the number of pixels as a measure of the "defect surface area" this agreement is usually quantified using the ratio of the intersection of the two areas and their union ("IOU-metric"), (Tanimoto 1958)[4].

4 Tanimoto, T. T. (1958). An elementary mathematical theory of classification and prediction. New York, International Business Machines Corporation

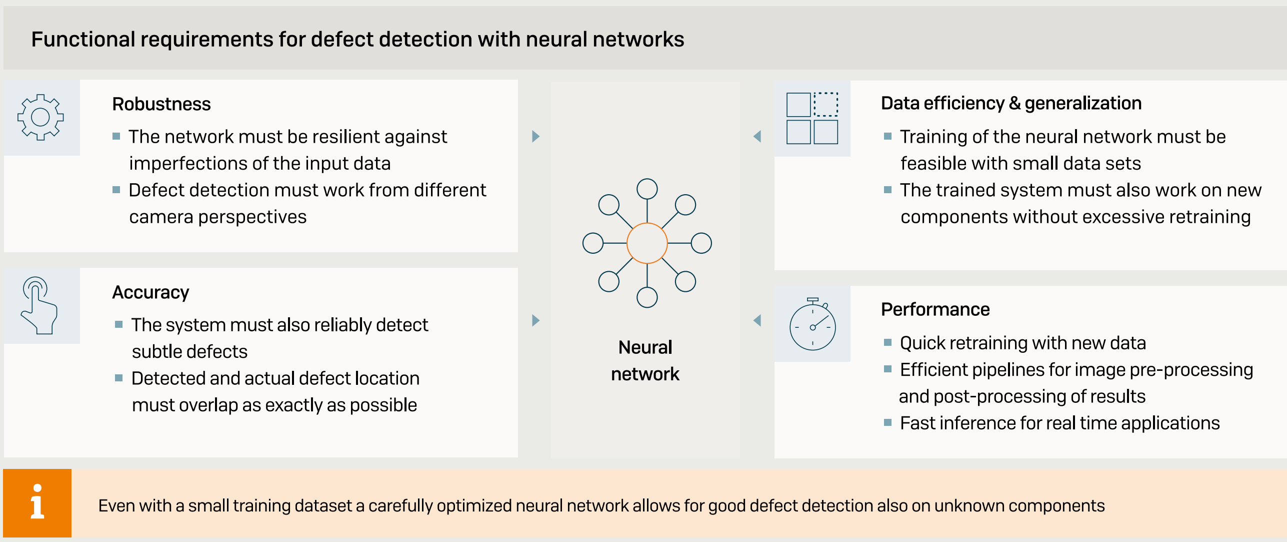

Figure 4: Requirements that a successful industrial image segmentation tool must meet

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. In practice, a perfect recognition is rarely achievable and real-world systems need to strike a balance that is tuned to the requirements of the use case and the customer. Our computer vision pipeline comprises a set of optimization methods to facilitate this tuning and to optimize the overall performance.

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

Our image recognition and segmentation stacks are built on the open-source software Tensorflow (Abadi et al., 2016)[5]. Its components have undergone extensive optimization and are kept up-to-date with the latest scientific developments by a large user and developer community. We have realized both cloud-based solutions and on-premises systems with real-time capabilities. To speed up the training process, we leverage efficient network parameterizations and performance monitoring tools.

04. Example use case: ball screw drives

To demonstrate the performance of our image segmentation and defect detection pipeline, we use a publicly available dataset of ball screw drive images and discuss the workflows and the achieved results.

The dataset "Industrial Machine Tool Element Surface Dataset" (Schlagenhauf et al. 2021)[6] by the Karsruhe Institute of Technology (KIT) contains images of ball screw drives (BSD). BSDs are roller bearings used for translating rotary motion into linear motion and "one of the most wear-prone machine tools" (Haberkern 1989)[7].

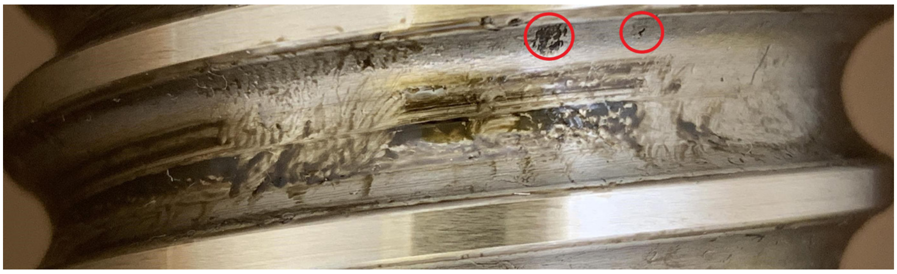

Figure 5: Example image of the KIT ball screw drive dataset - defect highlighted in red

The dataset consists of 394 images. Each of the images contains at least one defect, which was labelled by a human expert using a polygonal mask along the boundary of the damage region. The defect regions are of different sizes and the images show soiling as it is typical for industrial manufacturing environments.

We split the data into a training, a validation and a test dataset, and enrich the training dataset in a pre-processing step using various geometric and photometric data augmentations to cope with the small number of input images. Furthermore, defect areas are sampled more frequently to enhance their weight in the data.

5 Abadi, Martín, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin et al. "TensorFlow: A System for Large-Scale Machine Learning." In 12th USENIX symposium on operating systems design and implementation (OSDI 16), pp. 265-283. 2016.
6 Schlagenhauf et al. (2021): Industrial Machine Tool Element Surface Defect Dataset, 10.5445/IR/1000129520
7 Haberkern: Leistungsfähige Kugelgewindetriebe durch Beschichtung, Universität Karlsruhe, Institut für Werkzeugmaschinen und Betriebstechnik. 1998

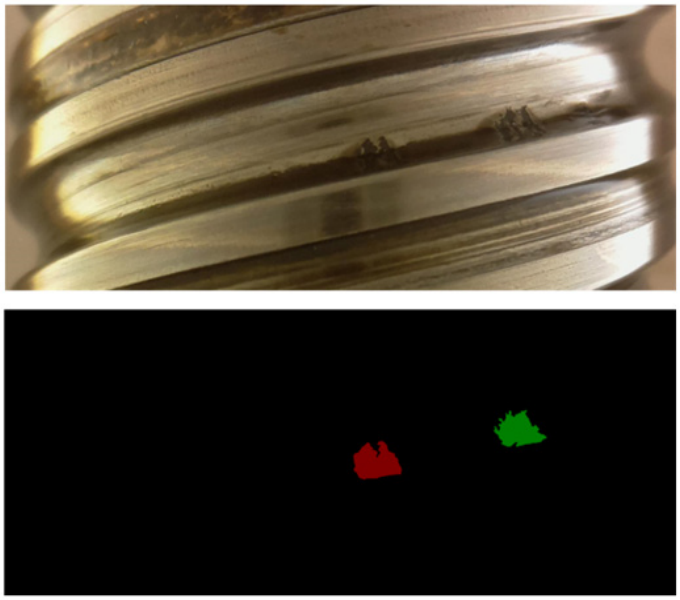

Figure 6: Image from the KIT ball screw drive dataset showing a defect (top) and the corresponding defect mask (bottom)

The trained convolutional network has a so-called U-shape architecture, which comprises a contracting and an expanding data path. Due to careful tuning of the network details, the system is already very data-efficient at the architectural level.

After training of the neural network, a post-processing step is applied to the detected labels which i.a., applies a threshold upon the confidence level with which the network considers a pixel to be part of a defective area.

In Figure 7, we show the results we achieve when applying the network to unseen error samples. We detect the defects on the test dataset with a mean IoU of 45.3%, which is a noticable improvement compared to previously published results (Schlagenhauf et al. 2021)[8]. The metric outputs the pixel-wise correspondence of the prediction and the exact location of the defect. Our toolchain detects even small defects very reliably.

8 Tobias Schlagenhauf, Magnus Landwehr, Industrial machine tool component surface defect dataset, Data in Brief, Volume 39, 2021, 107643, ISSN 2352-3409, https://doi.org/10.1016/j.dib.2021.107643.

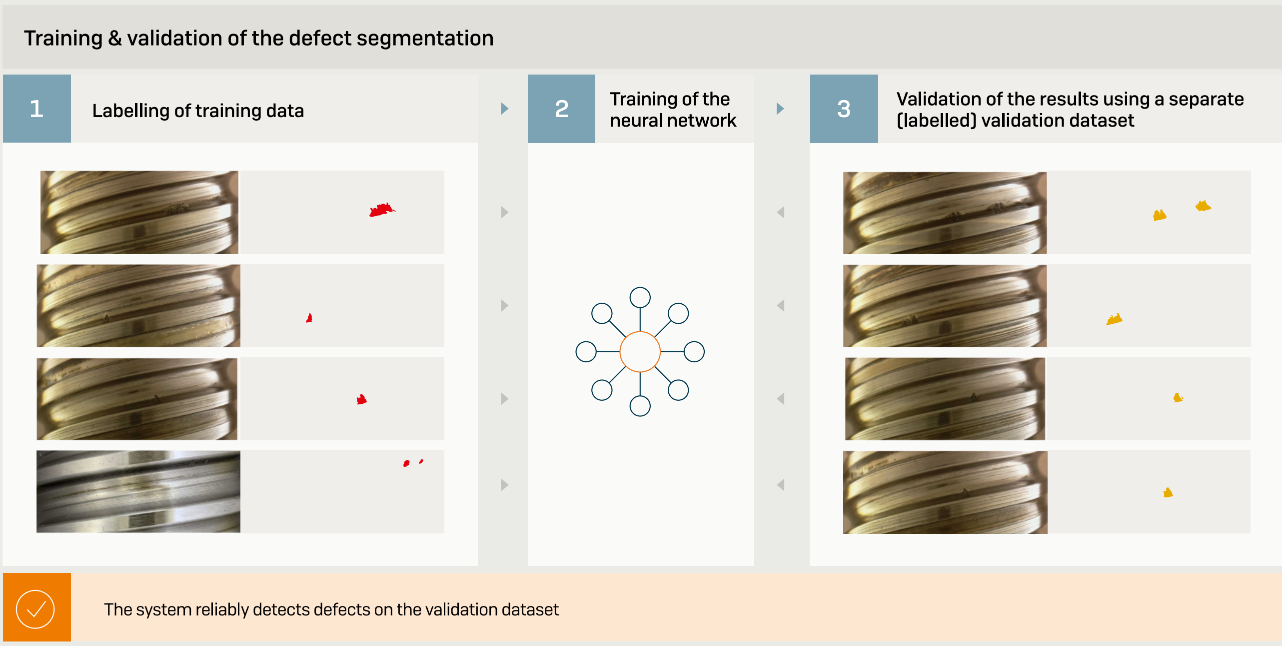

Figure 7: Defect detection workflow: (1) labelling of training data, (2) training of the neural network (3) validation of the results using a separate validation dataset.

05. How may we help you?

In all but the simplest cases, the successful application of Vision-AI in an industrial setting requires a firm mathematical understanding and a strong computer science background. The consultants of our Industrial Solutions Division provide an excellent quantitative background from fields like mathematics, computer science, physics or engineering and many years of experience in developing tailored image recognition solutions for complex industrial applications.

We work end-to-end with our clients as a single team and support you to conceptualize, implement, test, and deploy targeted solutions for your specific application. Contact us to discuss how we can support your business case.

We would like to thank the members of the wbk Institute of Production Science at the Karlsruhe Institute of Technology for providing us with the reference dataset used in this whitepaper and for the fruitful discussion about their related work.

Autoren

Johannes Landgraf
Consultant, d-fine GmbH, München
johannes.landgraf@d-fine.de

Dr. Moritz Kompenhans
Manager, d-fine GmbH, Frankfurt
moritz.kompenhans@d-fine.de

Dr. Tassilo Christ
Senior Manager, d-fine GmbH, München
tassilo.christ@d-fine.de

Berlin
d-fine GmbH
Friedrichstraße 68
10117 Berlin
Germany
berlin@d-fine.de

Dusseldorf
d-fine GmbH
Dreischeibenhaus 1
40211 Dusseldorf
Germany
duesseldorf@d-fine.de

Frankfurt
d-fine GmbH
An der Hauptwache 7
60313 Frankfurt
Germany
frankfurt@d-fine.de

Hamburg
d-fine GmbH
Rödingsmarkt 9
20459 Hamburg
Germany
hamburg@d-fine.de

Munich
d-fine GmbH
Bavariafilmplatz 8
82031 Grünwald
Germany
muenchen@d-fine.de

London
d-fine Ltd
6-7 Queen Street
London, EC4N 1SP
United Kingdom
london@d-fine.co.uk

Vienna
d-fine Austria GmbH
Seilerstätte 13/25-26
1010 Vienna
Austria
wien@d-fine.at

Zurich
d-fine AG
Brandschenkestrasse 150
8002 Zurich
Switzerland
zuerich@d-fine.ch

d-fine