Proceedings of the 35th European Safety and Reliability & the 33rd Society for Risk Analysis Europe Conference
Edited by Eirik Bjorheim Abrahamsen, Terje Aven, Frederic Bouder, Roger Flage, Marja Ylönen
 Published by Research Publishing, Singapore.
doi: 10.3850/978-981-94-3281-3_ESREL-SRA-E2025-P9408-cd

Optimizing Automated Optical Inspection for Printed Circuit Boards Using Computer Vision: A Comparative Study for Beneficial Imagesize Reduction

Jannis Pietruschka

Chair of Reliability Engineering and Risk Analytics, Faculty of Mechanical and Safety Engineering, University of Wuppertal, Germany. E-mail: pietruschka@uni-wuppertal.de

Saeideh Pourghasemian

Chair of Reliability Engineering and Risk Analytics, Faculty of Mechanical and Safety Engineering, University of Wuppertal, Germany. E-mail: pourghasemian@uni-wuppertal.de

Stefan Bracke

Chair of Reliability Engineering and Risk Analytics, Faculty of Mechanical and Safety Engineering, University of Wuppertal, Germany. E-mail: bracke@uni-wuppertal.de

The increasing digitalization makes printed circuit boards (PCBs) an almost indispensable component as a flexible internal interconnection technology in electronic devices. To ensure reliable operation in the final product, the quality of each manufacturing step must be closely monitored. Quality control can be carried out through visual inspection by trained personnel or via automated inspection systems. Since the range of defects that can occur during the manufacturing process is highly diverse and customer requirements for the boards vary, this presents challenging conditions for automated optical inspection (AOI). Several studies indicate that the effective image area, which meets the resolution requirements for evaluation, as well as the optical limitations of the camera lens and sensor, is around 15 to 20 cm^2. As a result, multiple images must be taken with a moving inspection head to fully assess PCBs, which typically have a larger surface area. This significantly limits the integrability into a continuous manufacturing process. Furthermore, real-time evaluation necessitates that the detection of defects be synchronised with production speed, which is also influenced by image size. Consequently, a compromise must be reached between image resolution and the level of detail captured in the area to ensure the reliability of the analysis of defects on the PCB. As part of the present study, images of various PCBs were generated, which were tinned using the hot-air leveling process. The evaluation is based on an One-stage detection model, which was trained with PCB images at different resolutions and identifies the regions relevant for defect detection. This comprises a comparative study conducted using diverse image preprocessing techniques, with the objective of optimising the evaluation speed and accuracy. The long-term objective is to implement the model for practical use, thereby enabling its use as an automated inspection method in continuous manufacturing processes.

Keywords: printed circuit boards (PCB), automated optical inspection (AOI), computer vision, machine learning, instance segmentation, YOLO, robust manufacturing process

1. Introduction

Electronic products have become an integral part of everyday life and industrial production. The use of products with integrated circuit boards and (micro) electrical components is steadily increasing. In an increasingly digitalized world, there is no sign of this trend declining. PCBs serve as a base in applications for mounting and electrically connecting components such as semiconductors or microchips, enabling the overall functionality of the component (Zickert, 2023).

In order to use microelectronic components, conductive substrates, or PCB, are required to provide a compact connection between various electronic components. These are placed on PCBs and soldered to achieve the desired function. The manufacturing process of a printed circuit board goes through many processing steps, each with an individual spectrum of possible defects. In this study, the defects that occur after the tinning process and thus before the assembly of components are investigated using bare PCB. The defects to

be detected vary from minor details, such as copper exposure, to surface defects that extend over the entire board (IPC-A-610J, 2024; IPC-6012-F, 2023). The different susceptibilities to defects require continuous quality controls, some of which can be automated, but some of which are also carried out manually. A large proportion of defects can be detected by visual inspection in particular. While manual optical inspection is characterised by inconsistent and subjective evaluation, automatic optical inspection faces the challenge of adapting to unknown PCB patterns when no precise inspection processes are specified or data on the trace structure (Gerber file format) is known. In cases where an optimal reference or template is available, such as Gerber data, digital image processing technology is frequently implemented in the Golden Board process, as it is adequate for detecting certain defects (Zhang and Cao, 2023). In consequence of the comparison with the optimal template, the pseudo error rate – locations identified as errors despite not being such – is correspondingly high. This may result in restrictions to production operations.

To establish a universal, protocol-independent approach for the automatic optical inspection of printed circuit boards, increasing research is being conducted into methods that leverage image-based artificial intelligence techniques (Jian et al., 2025; Zhou et al., 2023). The present study focuses on the image pre-processing steps that must be given due consideration prior to the identification of errors. One of the most substantial challenges in the pre-processing stage pertains to the selection of the image area to be analysed. This area must be chosen in accordance with the defect to be detected, ensuring that the resolution of the details is sufficiently high to ensure their recognition, and can still be evaluated in adequate time by image algorithms as part of the production process. For instance, (Richter, 2023) mentions effective image areas of approximately 15 – 20 cm^2 for common AOI systems. However, these are insufficient for larger component area defects, such as scratches or solder masking clearance, which can extend over the entire production sheet. Therefore, for this contribution an image data set is generated, which is used for the training of a object segmentation algorithmn. The segmentation focusses the division of the PCB production boards into individual PCBs, thereby enabling the evaluation of smaller patches by the algorithm in a more time-efficient manner while maintaining a high resolution for defect detection. The model is part of the pre-processing of the images to enable real-time inspection in production.

1.1. *State of the art - PCB manufacturing process and inspection*

The process of manufacturing printed circuit boards (PCBs) is comprised of up to 30 distinct production steps. A distinction can be made between multi-layer PCBs and single-layer PCBs, as well as single-sided and double-sided PCBs. The focus of this study is solely on the outer surface of the bare PCBs. One of the final steps in the production of printed circuit boards is the coating of copper surfaces that are still free with a tin alloy. For this reason, this article focuses on the automatic inspection of printed circuit boards after the tinning process in order to identify defective components before applying the microelectronic components, thus reducing rejects. The specific tinning process used to produce the data generated in the article is Hot Air Leveling (HAL). In this process, the PCBs are wetted with flux and then held vertically in liquid tin for a few seconds. When the PCBs are pulled out, excess solder and solder inside the holes is blown out using compressed air. The finished product is only protected from oxidation at the copper points by a tin coating of 2 to 20 μm, which also improves solderability. Hot Air Leveling is one of the most important processes in the pretreatment of printed circuit boards prior to subsequent assembly and remains one of the most widely used industrial processes due to its exceptionally high productivity (Schlaeger, 2007).

This poses a particular challenge to the automatic inspection of PCBs, in that it must be carefully adapted to the continuous feed process while also being capable of real-time evaluation. As illustrated in the section of the Figure 1, there are multiple PCB present on a specific section

Fig. 1. Section of a PCB production sheet: Three PCB and offcut of the production board, which is recognizable as a grid pattern.

of a production panel, each requiring accurate inspection. These production panels are capable of measuring up to approximately 650 mm in both the horizontal and vertical dimensions, thereby enabling images of the entire production panel to contain 8,000 pixels along each edge. This is particularly challenging for effective algorithmic image evaluation, since the images must have a high resolution in order to make the entire defect spectrum detectable, which also results in high computational effort, requiring advanced processing power.

1.2. *Goal of this study*

The present study proposes an instance segmentation approach for the development of a model suitable for industrial AOI applications. The objective of this study is to sample the PCB on the production panel with high precision, in relation to their production borders, while circumventing the difficulties that arose when utilising a fixed, undependent cropping algorithm.

2. Fundamentals

2.1. *Deep Computer Vision*

Optical inspection and computer vision play a pivotal role in monitoring system conditions. In this context, deep learning has emerged as a leading technology (Zhao et al., 2024). Deep neural networks, with their architecture consisting of multiple hidden layers, enable the automatic extraction of complex and abstract features from data (Khanam et al., 2024).

Among various types of deep neural networks, Convolutional Neural Networks (CNNs) have been specifically developed to process grid-structured data such as images. These networks have achieved remarkable success in the field of computer vision as they can adaptively and automatically learn the hierarchical spatial features of images. This capability makes CNNs efficient tools for tasks such as image recognition, object detection, and image segmentation. The ability of CNNs to identify and extract hierarchical and complex features is a key factor in their success across domains such as facial recognition, medical image analysis, and autonomous systems (Chen et al., 2023). The architecture of CNNs typically comprises three main components: convolutional layers, pooling layers, and fully connected layers (Zhao et al., 2024).

In the context of image segmentation, the process of detecting objects involves the generation of pixel-wise masks that delineate the object boundaries. This approach was pioneered by the Fully Convolutional Networks (FCNs)(Long et al., 2014), SegNet (Badrinarayanan et al., 2015) or Mask R-CNN (Region-based Convolutional Neural Network) (He et al., 2017), which utilise mask-based techniques to ensure precise localization and segmentation of objects within an image. CNNs frequently constitute as the fundamental network architecture (the backbone), which is responsible for extracting relevant features at different scales, often based on pre-trained large-scale image classification models such as ImageNet (Terven et al., 2023). The neck is a crucial intermediate component in the such structures, connecting the backbone to the head. Its primary function is to aggregate and refine the features extracted by the backbone, with a focus on enhancing spatial and semantic information at various scales. To enhance feature representation, the neck frequently incorporates additional convolutional layers, feature pyramid networks (FPN) (Lin et al., 2017), or other advanced mechanisms. The final

output, in the form of classification and segmentation, is executed by the head, which accesses the backbone at various levels through task-specific subnetworks via the neck (Terven et al., 2023). The optimisation of these structures for their respective computer vision areas, and the training of them for object detection, remains a subject of ongoing research.

2.2. *Network Architecture - YOLO*

In the field of instance segmentation, two predominant approaches have emerged within the backbone of CNNs: two-stage object detection and one-stage object detection. The two-stage detectors, in their initial phase, subdivide regions of interest (RoI) through the utilisation of bounding boxes. Subsequently, these boxes are subjected to an extraction of features, which is undertaken in a second step. Conversely, the one-stage detector utilises the features of the entire input image, thereby enabling expedited evaluations and rendering it particularly well-suited for real-time applications, such as autonomous driving (Sun et al., 2024). A widely utilised one-stage detection model is the You Only Look Once (YOLO) framework, which was employed in this study in its eighth and eleventh versions for instance segmentation of PCB on a production panel (Jocher et al., 2023; Jocher and Qui, 2024). Since 2015, the YOLO algorithm has been subject to a number of refinements, with the principal objective of optimising inference detection in terms of speed and timeliness. To date, eleven versions of the algorithm have been developed, several of which also contain a changeable head architecture that allows the instances to be segmented (Sun et al., 2024). Based on the CSPDarknet53 framework of the fifth version of the Yolo framework, C2f modules (cross-level partial bottleneck with two folds) were added in the architecture of the eighth version to improve accuracy through deeper combinations with high-level features (Terven et al., 2023). With further architectural adaptations and extended attention mechanisms, YOLO version 11 is geared towards improvements in the detection of small objects. In both cases, different sized frameworks (nano, small, medium, large and extra large) are available for training on custom data sets, which can be selected depending on the hardware requirements (Khanam et al., 2024).

3. Data generation

The quantity and quality of the data play a decisive role in the successful development of machine learning models. For this case study, a database consisting of 91 images was prepared. Each image represents a PCB panel that contains one or more PCB sections.

There are specific regions within each PCB panel where defects are more likely to be observed. To identify these regions, the machine learning model must first identify the overall area of the PCB panel and then independently identify each of the PCB sections. In this way, each image will contain two categories of labels: PCB panel and PCB section.

The image labeling process was performed using the Labelme tool. Initially, the overall area of the PCB panel was identified, followed by the precise identification and labeling of its individual sections. In Figure 2, a sample of a labeled image is shown, which contains one label from the ”PCB panel” category and two labels from the ”PCB section” category.

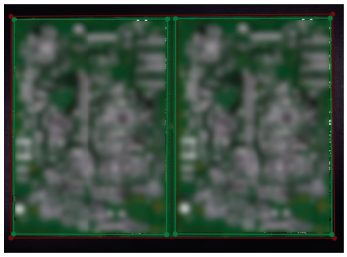


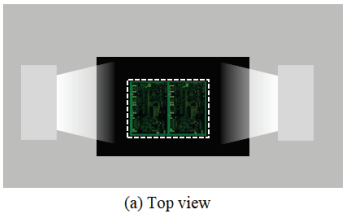
Fig. 2. Sample of a labeled image. Red and green boundaries delineate the ’PCB panel’ and ’PCB section’ categories, respectively.

3.1. *Experimental setup*

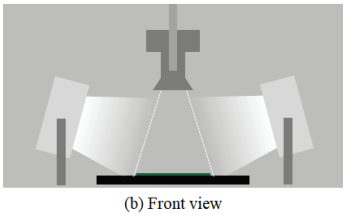
A testing setup, as shown in Figure 3, was designed for uniform lighting when capturing PCB panel images. It features white walls to block ambient light and evenly distribute internal light, with

two LED spotlights aimed at the target to reduce reflections. The matte black background ensures high contrast, mimicking real-world assembly line conditions for easy integration with AOI systems. The images were captured using a Canon EOS 77D camera with a Canon EF-S 18 - 55 mm 1:4 - 5.6 IS STM lens, with a resolution of 6000x4000 pixels, positioned circa 52 cm from the PCB panels. The camera settings were as follows:

- Aperture: F/4.5
- Exposure time: 1/80 sec
- ISO sensitivity: ISO-3200
- Focal length: 21 mm
- Color display: sRGB (three channels)



(a) Top view



(b) Front view

Fig. 3. Schematic illustration of the camera lighting system in the test setup. (a) Top view; (b) Front view

3.2. *Dataset*

Given the limited number of initial images (91) for model training, advanced data augmentation techniques were employed to enhance the diversity and volume of the dataset. This process enables the model to learn a broader range of features, thereby improving its performance in recognizing new and varied patterns. The steps of this process are detailed below:

Resizing to Square Format To align the image dimensions with the requirements of the YOLO model, all images were converted to a square format by adding black borders around each image, resulting in dimensions of 6000 x 6000 pixels. This approach preserves the geometric structure of the images.

Image Rotation To increase data diversity and enhance the model's ability to detect patterns from various angles, the images were rotated at eight distinct angles. Meanwhile, the label coordinates were updated using the mathematical formula for point rotation around a center, as given in Eq. 1:

$$\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} x - c_x \\ y - c_y \end{bmatrix} + \begin{bmatrix} c_x \\ c_y \end{bmatrix} \quad (1)$$

where (x, y) represents the original coordinates, $(\hat{x}, \hat{y})$ are the rotated coordinates, (c_x, c_y) is the center of the image, and θ is the rotation angle. Any samples where rotation caused parts of the panel to fall outside the image boundaries were excluded to maintain dataset precision and quality.

4. Comprehensiv model analysis

4.1. *Model architecture*

The model for the instance segmentation of PCB on a PCB prodcution panel was developed using the software system of YOLO from Ultralytics (Jocher et al., 2023). YOLO v8 and v11 were used as the basis for instance segmentation, which basically have a backbone structure of CSPDarknet53 but with some specific differences regarding the neck to head connection modules. Further information of the structure can be found in the primary sources of the frameworks (Jocher et al., 2023; Jocher and Qui, 2024) and in detailed comparison literature (Terven et al., 2023).

4.2. *Model analysis*

4.2.1. *Training and parameterization*

In this section a detailed analysis of the Training process is performed and discussed. As part of the study, various influencing variables, parameters and models were examined to determine the best possible YOLO configuration for the instance segmentation of PCBs on PCB panels. Based on the data set presented in Section 3.2, a total of 40 models with 200 epochs each were trained and

transferred to the given context. Only images from the RGB range were used, which were divided into training, validation and test sets of 70 % / 17.5 % / 12.5 %. As a consequence of the augmentation, there are rotated versions of the test images in the training and validation data sets. The study utilised an NVIDIA RTX A6000 (Ada Lovelace) GPU with 48 GB VRAM. Overall, the following parameters and their influence on the accuracy and reliability of the segmentation were examined in the study:

- Modelarchitecture: Five different versions of the respective YOLO version
- Resolutions of input images: 320 / 640 / 1024
- Batchsize: 16 / 20 / 32

Depending on the hardware requirements, it is possible to test 5 different sizes of YOLO frameworks. As these have slight differences in architecture and therefore also in performance, they were trained with different batch sizes and input resolutions of images. Since the starting weights of the YOLO models are always aligned with the pre-trained MS COCO data set, the respective models and the individual parameterizations are not reiterated. The performance with regard to the lowest validation segmentation loss (vsl) is the optimization goal of the models.

Table 1. Prediction results obtained by using the different modelversion based on the initial training set.

Model	Mean training seg. loss	Mean validation seg. loss
YOLOv8n	0.471 ± 2.3e-2	0.332 ± 9.9e-3
YOLOv8s	0.437 ± 2.6e-2	0.296 ± 4.6e-3
YOLOv8m	0.421 ± 2.3e-2	0.306 ± 6.2e-3
YOLOv8l	0.451 ± 1.5e-2	0.321 ± 7.7e-3
YOLOv8x	0.335 ± 2.8e-3	0.290 ± 3-0e-3
YOLOv11n	0.393 ± 1.8e-2	0.338 ± 9.0e-3
YOLOv11s	0.402 ± 5.4e-3	0.319 ± 4.1e-3
YOLOv11m	0.381 ± 4.9e-3	0.319 ± 3.9e-3

Table 1 shows that there are no major qualitative differences in the overall view of model architecture performance. A more detailed examination of the results also shows that the lowest vsl was achieved by a YOLO of the eighth version as an small framework. This indicates that in this case the new architectures of YOLO v11 have no positive effect on the general segmentation accuracy.

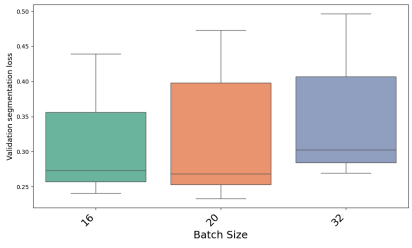


Fig. 4. Comparison of the group results to validate the parameter influence: Boxplots of different batchsizes.

However, the boxplot in Figure 4 shows that the batch size has an influence on the performance of the models. To the advantage of the GPU training load, a smaller batch size can have a positive effect on performance. Despite the significant disparities between the lowest resolution of 320 and 640 pixels per side and the boxplots of the resolution of 1024 and 640 pixels per edge overlap, it is noteworthy that at least half of the models with a higher resolution are below the lowest vsl of the average resolution in this context. Another advantage of the option of a smaller batch size is that higher resolution images can be used. Why this provides a significant performance boost in this use case becomes clear in Figure 5. This leads to the conclusion that a higher resolution provides more necessary details, which are needed to create the masks on the higher levels more precisely.

When testing unseen images from the test data set, it can also be seen that the masks can already create very precise masks with 200 training epochs in order to segment even more complex PCB structures. Figure 6 shows an example of segmentation in which both the production panel and the geometries of the PCB are cropped. Furthermore, the confidence value of the predicted label of 0.97 % demonstrates that the model is highly confident in its segmentation, which is indicative of reliable implementability.

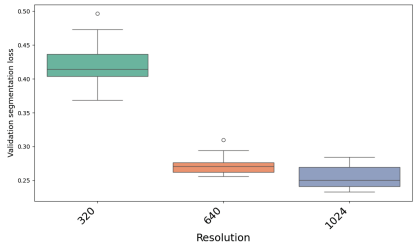


Fig. 5. Comparison of the group results to validate the parameter influence: Boxplots of different image resolutions.

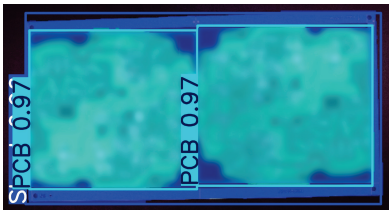


Fig. 6. Predicted Segmentation by best YOLOv8 model of a complex PCB panel structure.

The analysis of the parameter study demonstrated that the YOLO model is entirely suitable for the segmentation task. In order to eliminate the possibility of an unconverged learning process, a final model structure with additional learning iterations and optimised parameters is implemented in the following section, based on the findings of the parameter study.

4.2.2. *Model Testing*

The parameter study showed that a higher resolution and a smaller batch size have a positive influence on the models. Accordingly, a YOLOv8 in small is trained with a further increased image resolution of 2016 pixels per side and a smaller batch size of ten over 500 epochs. The learning process is illustrated in the Figure 7 using the segmentation loss, in which the validation results are compared to the training results. From these results it can be concluded that no substantial enhancements in performance are anticipated through subsequent epochs, thus indicating that this model has likely converged. The final performance metrics of the final model are as follows: 0.1896 vsl, 0.1504 validation box loss, 98.44 % box recall with a 98.17 % box precision. This signifies a notable enhancement in performance when compared to the study results. In addition, the inference time for the evaluation is approx. 6.0 - 7.8 milliseconds per image, which once again demonstrates the suitability of the model for real-time implementation in production environments.

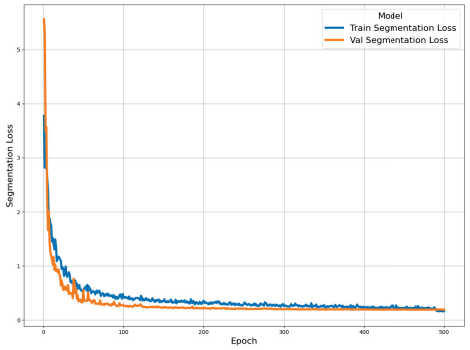


Fig. 7. Learning History of final modelarchitecture with 500 epochs

5. Summary and outlook

In this study, a model to segmentate printed circuit boards from a PCB production panel was developed and optimised for further defect analysis. The model is based on a one-stage detection architecture of YOLO used in different versions. In the first step, a data set of images was generated and prepared for the instance segmentation. In the second step, a training study was performed to evaluate different parameters and their effect on the model performance. The third step of the study involved the training of a final model, which exhibited a substantially enhanced performance in comparison to the models from the parameter study. The results of the final model demonstrated a level of performance that is suitable for direct implementation in the production process of PCBs, with an average inference time per picture of 6.3 ms.

The segmentation model that has been presented provides a foundation upon which more precise analyses for defect detection in the continuous production of PCBs with an AI-based AOI

can be performed. The segmented images reduce the examined areas of a panel into meaningful sections so that each panel can be viewed to detect local defects but not rule out larger defects that extend across the entire panel. The smaller image sections also enable higher resolutions in the evaluation, which increases the probability of detecting smaller, more detailed defects on the circuit board. In the future, the model is to be implemented in a pipeline for automated defect detection in an AOI machine, in order to test its use and transferability to the real process.

Acknowledgement

The authors thank Dr. Sabine Schlaeger-Diegel and Frank Winter from the company PENTAGAL Chemie und Maschinenbau GmbH for the support and the cooperation. The study is part of a research project funded by Federal Ministy for Economic Affairs and Climate Action and Zentrales Innovationsprogramm Mittelstand (ZIM). The authors thank the project sponsors for the financial support.

Supported by:

Federal Ministry for Economic Affairs and Climate Action

ZIM Zentrales Innovationsprogramm Mittelstand

on the basis of a decision by the German Bundestag

References

Badrinarayanan, V., A. Kendall, and R. Cipolla (2015). Segnet: A deep convolutional encoder-decoder architecture for image segmentation.

Chen, X., Y. Wu, X. He, and W. Ming (2023). A comprehensive review of deep learning-based pcb defect detection. *IEEE Access 11*, 139017–139038.

He, K., G. Gkioxari, P. Dollár, and R. Girshick (2017). Mask r-cnn.

IPC-6012-F (09.01.2023). Ipc-6012-f: Qualification and performance specification for rigid printed boards.

IPC-A-610J (03/01/2024). Ipc-a-610j: Acceptability of electronic assemblies.

Jian, T. S., M. H. F. M. Fauadi, S. H. Yahaya, A. Z. M. Noor, and A. Saptari (2025). A deep learning approach for automated pcb defect detection: A comprehensive review. *Multidisciplinary Reviews 8*(1), 2025011.

Jocher, G., A. Chaurasia, and J. Qui (2023). Ultralytics yolov8.

Jocher, G. and J. Qui (2024). Ultralytics yolo11.

Khanam, R., M. Hussain, R. Hill, and P. Allen (2024). A comprehensive review of convolutional neural networks for defect detection in industrial applications. *IEEE Access 12*, 94250–94295.

Lin, T.-Y., P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie (2017). Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 936–944.

Long, J., E. Shelhamer, and T. Darrell (2014). Fully convolutional networks for semantic segmentation.

Richter, J. (2023). *Analyse und Entwicklung einer Softwarearchitektur für die intelligente, optische Inspektion*. Ph. D. thesis, Universitätsverlag Ilmenau.

Schlaeger, H.-G. (2007). Einsatz von bleifreiloten für die heißluft-verzinnung (pentagal chemie und maschinenbau gmbh). *DBU Deutschen Bundesstiftung Umwelt*.

Sun, Y., Z. Sun, and W. Chen (2024). The evolution of object detection methods. *Engineering Applications of Artificial Intelligence 133*, 108458.

Terven, J., D.-M. Córdova-Esparza, and J.-A. Romero-González (2023). A comprehensive review of yolo architectures in computer vision: From yolov1 to yolov8 and yolo-nas. *Machine Learning and Knowledge Extraction 5*(4), 1680–1716.

Zhang, G. and Y. Cao (2023). A novel pcb defect detection method based on digital image processing. *Journal of Physics: Conference Series 2562*(1), 012030.

Zhao, X., L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar (2024). A review of convolutional neural networks in computer vision. *Artificial Intelligence Review 57*(4).

Zhou, Y., M. Yuan, J. Zhang, G. Ding, and S. Qin (2023). Review of vision-based defect detection research and its perspectives for printed circuit board. *Journal of Manufacturing Systems 70*, 557–578.

Zickert, G. (2023). *Leiterplatten Stromlaufplan, Layout und Fertigung* (3., aktualisierte Auflage ed.). München: Hanser.