Intelligent Augmentation Methods for Training Defect Detection on Circuit Boards

Olaf Kähler **Werner Bailer** **Georg Thallinger**
JOANNEUM RESEARCH Forschungsgesellschaft mbH
DIGITAL – Institut für Digitale Technologien
Steyrergasse 17, 8010 Graz, Austria
{olaf.kaehler,werner.bailer,georg.thallinger}@joanneum.at

Abstract

We discuss intelligent data augmentation strategies to help train object detection models from low-volume datasets. In particular, many industrial inspection tasks suffer from a lack of samples showing defects in the training data, and furthermore the failure cases are typically heterogeneous, leaving only a handful of samples for each of them. For our application scenario of printed circuit board (PCB) inspection, we propose and evaluate a strategy for synthesizing defects, as well as a strategy to copy-paste difficult, challenging, or otherwise rare cases into the training images. Maintaining this library of challenging or rare cases offers an easy way to update the model and integrate feedback after deployment. We evaluate the benefits of the augmentation strategies in experiments and present a reliable and accurate PCB inspection model trained with only 25 images.

1 Introduction

While dataset sizes for vision-language or foundation models are ever increasing, low-volume datasets are still a common challenge in many specialized real-world applications of machine learning. In particular, training samples of defects are typically rare in industrial inspection tasks and often show a heterogeneous range of defects with even fewer samples for any given failure mode. In this paper, we present and evaluate two data augmentation techniques to deal with low-volume datasets in an application to printed circuit board (PCB) inspection. The first approach automatically synthesizes defects on annotated instances of intact objects, the second is a variant of copy-paste augmentation, which enables continuous improvement of the model with little effort.

In the intended application scenario, PCBs are to be disassembled at the end of their useful life to recover reusable components from the boards [19] while discarding defective parts. Closely related inspection tasks also arise when assessing PCB repairability, e.g. by replacing individual defective components, or even as a quality control measure in PCB assembly. All of these scenarios basically need a system for identification and classification of PCB components, which is a classical object detection problem, except that there are only very few samples of defective parts available for training.

The approaches we propose for dealing with this lack of training data are widely applicable to a range of similar scenarios – it is very common that samples of real defects are rare when developing and training a model. In our particular case, we synthesize two kinds of typical defects using elementary image processing operations. This dramatically reduces the class imbalance and hence boosts classification performance. As an additional direction, we copy and paste instances from a library of challenging examples to random places into the training images, similar to [3, 4]. However, we propose manually maintaining and extending this library of challenging samples as an avenue for incorporating feedback using targeted adaptation and retraining.

The Third Austrian Symposium on AI, Robotics, and Vision (AIROV26).

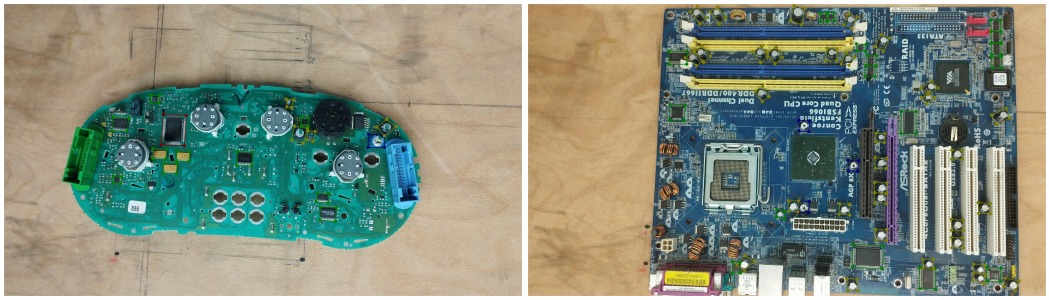

Figure 1: Samples of the images and annotations in the PCB dataset. Defect and intact ICs as well as defect and intact capacitors are annotated, but we focus on ICs in this work. On average, more than 7 ICs appear in each image and they typically appear relatively small. The recording perspective and illumination are constant within the entire dataset.

In the following, we will briefly revisit the relevant literature in Section 2. Continuing in Sections 3 and 4, we will give details of our proposed data augmentation techniques. Finally, we will evaluate our methods in Section 5 and summarize our findings in Section 6.

2 Literature Overview

Automated optical inspection of PCBs during manufacturing is receiving a lot of attention. A recent review of methods focusing on defects in PCB traces is given in [1], and an overview of methods for inspecting mounted components in [20]. In a similar vein, the detection of PCB components and the checking of completeness and correctness have been covered before with publicly available benchmark datasets. For example, the PCB-DSLR dataset [16] focuses on integrated circuits (ICs) and additional components are also considered in the PCB-METAL [12], WACV-PCB [8] and FICS-PCB [10] datasets. In contrast, end-of-life treatment and defects, that occur during use of components, have received little attention so far. Consequently, to the best of our knowledge, no datasets covering such defects have been published.

Synthesizing defects is a common strategy in industrial inspection tasks [13]. Samples of works that use synthetic defects in surface inspection include [6], and a recent work includes a sophisticated combination of a rendering pipeline and style transfer to maximize realism [17]. In addition, in the context of PCB inspection during assembly [18] and for X-ray CT scans of ICs [14], synthetic defects have been evaluated before. Relevant defects are often specific for the respective application scenario, and, to our knowledge, no pipeline has been proposed for synthesizing end-of-life defects from the real use of electronic components so far. Accordingly, we consider the details of our synthetization pipeline novel and different from the state-of-the-art.

In contrast, we see our copy-paste augmentation as a more generic strategy, which follows the idea of [4]. Although existing works evaluated the benefits of considering context when mixing images [5], we focus more on the library of source patches that such strategies can paste into new images. Although our experiments can only verify this in a limited scope, we believe that model training can be guided by deliberately selecting rare or novel out-of-distribution samples for the source library. To our knowledge, this aspect has also been neglected in the existing literature.

3 Synthesizing Defects

In our work, we consider a dataset for PCB inspection as shown in Figure 1, which is a subset of the dataset in [15]. In total, it consists of 101 images and shows 652 instances of intact ICs and additionally 78 instances of defect ICs. The defects are not further subdivided but contain a set of burned ICs with visible black spots on the surface and a set of ICs with broken and merged pins. Samples of these defects are shown in Figure 2. It is a common scenario that defects are rare and heterogeneous, such as in our dataset. To alleviate this class imbalance, we aim to introduce synthetic defects to the known and annotated ICs in our data.

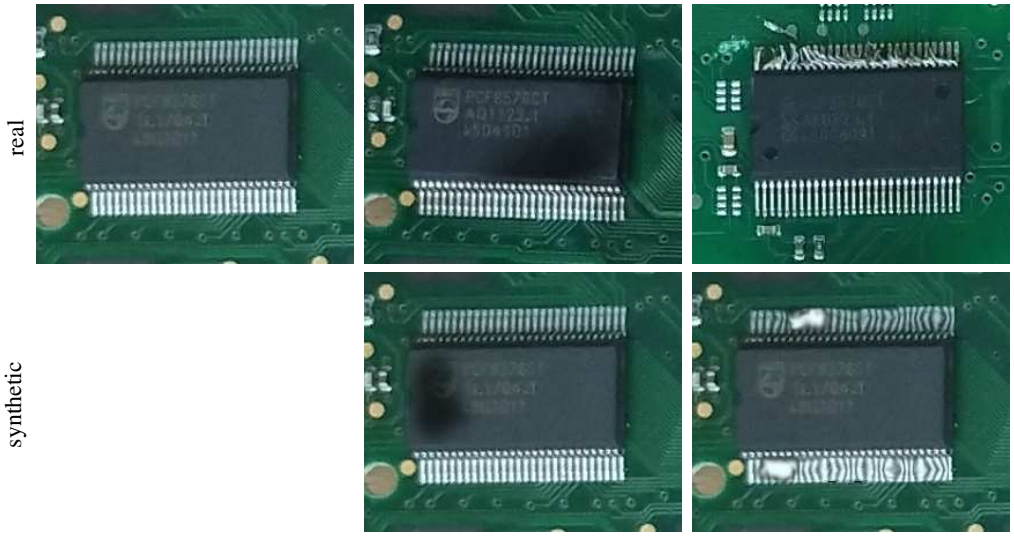

Figure 2: Samples of real defects in the training data in the top row as well as corresponding synthesized additional defects at the bottom. The synthetic defects are all added to the same reference image of an intact IC shown on the top left.

Since the perspective and illumination of images are fairly well constrained in our scenario, we skip any 3D modeling and rendering. Realistic and varied training data can be generated with 2D image processing in our case. In the following, we propose two different approaches to cater for the two modes of defects we observe in the given real sample data, namely dark spots on the IC package and pin deformations.

The first kind of synthetic defect introduces dark burn marks onto the ICs. To this end, we reduce the annotated bounding box of an intact IC by 10%, generate a random polyline chain within these bounds, apply some heavy Gaussian smoothing on its rendering, and alpha-blend the result onto the original input image. This simple but effective strategy produces realistic dark spots as shown in Figure 2.

For pin deformations, we perform three steps to a) identify image areas covered by pins, b) apply a random deformation to the relevant area, and c) add a random white spot on top resembling a solder blob. To identify the pin area, we compute image gradients near the image edges and identify blobs of high gradients. For deformations, a coarse mesh of source and target coordinates is generated randomly and bilinearly interpolated. This will ensure continuous and somewhat realistic looking outputs. In addition, we apply a similar procedure as for burn marks to add white spots onto the pins to simulate solder blobs. Again, a sample of the final result is shown in Figure 2.

The aforementioned transformations are applied to a random subset of all intact ICs in an image, and of course the result is an image with additional realistic-looking defective ICs. The procedures involved are lightweight and robust. They can hence be applied not only offline to generate a synthetic training dataset but also online as an additional data augmentation step.

4 Copy-Paste Augmentation

The aforementioned augmentation strategy significantly increases the variety of data, but does not offer a way of introducing novel variants of ICs or defects and adapting to distribution shifts. Annotating entirely new images for retraining purposes is an option here, but each image will easily contain dozens of objects, imposing a large annotation effort for each new sample.

We therefore argue that a much smaller library of challenging samples or rare cases can be extracted from additional training images. This merely requires cropping e.g. a single IC from a novel image, annotating it, and saving it to the library. Furthermore, with pre-cropped content, the annotation can

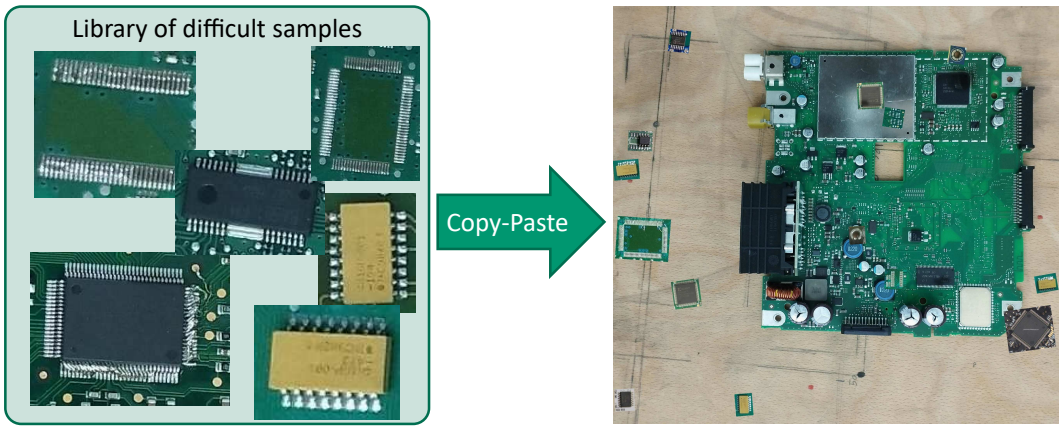

Figure 3: Illustration of pasting challenging samples from a library into an existing training image. The inserted objects appear at random locations and with minor variations in scale and orientation. Note that we do not perform any smoothing at the edges.

Table 1: Libraries of additional training instances used for weak and strong copy-paste augmentation.

	Weak Augmentation			Strong Augmentation		
Source	**intact**	**defect**	**background**	**intact**	**defect**	**background**
Training	7	0	16	7	0	16
Extra	5	3	0	26	5	4
WACV-PCB [8]	6	0	1	37	0	4

be done efficiently with models such as SegmentAnything [7], in most cases with a single inference run. A sample of such a library of rare cases is illustrated in Figure 3.

These crops are then pasted to random locations in the training images, where no other object is already present. This simplifies handling of overlapping boxes or annotation masks and is sufficiently realistic for our images of flat PCBs with components mounted on the top. To further increase variability, we apply random scaling by $\pm 10\%$, flipping, rotations by multiples of $90°$, and additional rotations by up to $\pm 10°$. We do not apply any smoothing to the edges of the pasted crops, as this does not generate any additional benefit. This is in line with reports, e.g. in [4]. If outlines of the PCBs are known, this procedure can be further refined so that the novel samples are only pasted to the foreground regions. With unknown PCB outlines, simply pasting larger numbers of samples in both the foreground and background is also feasible. Pasting of multiple novel objects can be handled sequentially.

Maintaining the library of challenging or rare samples provides an easy way to incorporate feedback into the training, since expanding this library is a quick and efficient process. For a real application, we envisage using this process, whenever a misdetection is observed by an operator. For reproducible results in our experiments, we only incorporate a fixed set of additional training samples. In particular, these samples are extracted using images from the WACV-PCB dataset [8], as well as 7 additional images, that are not part of any other dataset. We also include crops of challenging parts from the original training data to ensure that these have a good chance of being properly learned. To evaluate whether just repeating known parts of the training data improves performance or whether the actually novel out-of-distribution samples help with generalization, we consider weak and strong copy-paste libraries in our experiments, with detailed statistics given in Table 1.

Note that a majority of the ICs in these libraries are in fact intact. Defects therefore remain rare as the WACV-PCB dataset, like other relevant datasets, does not contain any defects. Hence, a combination with the aforementioned methods of synthesizing defects seems to be the most promising to boost performance. In particular, there are no defective samples of the rare yellow ICs shown as part of our library in Figure 3, but with the combination of our proposed techniques, these are also represented in the training process. Examples of augmentation results for the original images from Figure 1 are shown in Figure 4.

Table 2: Evaluation of several object detectors with different augmentation schemes.

Synthesis	Copy-Paste	Mask2Former [2]		Dino [21]		ConvNeXt [9]		RTMDet [11]	
		mAP @0.5	mAP @0.75	mAP @0.5	mAP @0.75	mAP @0.5	mAP @0.75	mAP @0.5	mAP @0.75
-	-	78.4	67.2	83.7	60.1	88.1	78.9	62.2	33.8
-	weak	80.3	69.8	86.1	62.7	89.3	81.0	67.3	46.7
-	strong	80.7	71.1	87.1	63.3	88.8	80.6	69.4	46.6
active	-	87.6	**81.0**	90.8	68.1	91.8	**84.6**	67.6	48.6
active	weak	88.3	79.0	92.3	**70.1**	92.6	82.9	**69.7**	**49.1**
active	strong	**89.1**	80.5	**92.4**	70.0	**93.1**	84.2	68.6	47.7

5 Evaluation

In order to verify the efficiency of the proposed data augmentation schemes, we first perform a qualitative visual analysis and then a thorough quantitative analysis of their effect on the training results.

For visual analysis, we inspect several of the augmented images, such as shown in Figure 4. In these images, a random number of up to 20 patches from the copy-paste library are added, and up to 70% of the intact ICs are modified by adding synthetic defects. Closeups of real defects, synthetic defects, and synthetic defects on pasted ICs from the library are also shown in Figure 5. Although a human observer can spot the difference between real and synthetic defects and the pasted image patches do not line up with the rest of the PCBs, the overall appearance of the modified training data is realistic enough upon casual observation. In any case, we consider it worthwhile to run experiments to validate their impact on the training of machine learning models.

To this end, we performed several experiments on the presented dataset of 101 images. We randomly split this dataset into a fixed training set of 25 images, containing 196 intact and 15 defect ICs respectively, and a validation set of 76 images, showing 456 intact and 63 defect ICs. The training set is deliberately selected as small as 25 images to show the efficiency of our data augmentation strategies for low-volume datasets. Experiments with a split of 75 training images and 26 validation images showed similar trends, but at higher variance due to the small validation dataset.

We train various object detection and instance segmentation models, in particular Mask2Former [2], Dino [21], Cascade Mask-RCNN with a ConvNeXt backbone [9] and RTMDet [11], to ensure that our results are generalizable and transferable. For training, we use stochastic gradient descent running for 1,000 epochs, while we observe that all models achieve convergence and stabilize their performance way before the training ends. The model weights are initialized from pretraining on the COCO dataset. A cosine annealing strategy is applied to reduce the learning rate, and we switch from a strong data augmentation pipeline for the first 600 epochs to a lighter one for the remaining final epochs. If enabled, the first steps in data augmentation are our proposed copy-paste augmentation and the strategies for synthesizing defects on annotated ICs. This is followed by a strong set of standard data augmentation routines of mosaicing, resizing, cropping, HSV and photometric distortions, random horizontal and/or vertical flipping, mixup and random rotations of up to 15°. With this strong set of state-of-the-art augmentation methods, we observe little to no signs of overfitting in any of our experiments.

To evaluate the effects of our novel data augmentation schemes, we first train a baseline model using only the state-of-the-art standard augmentation methods. We then selectively apply and combine our proposed copy-paste augmentation and the pipeline to generate synthetic defects. The resulting values for mAP@0.5 and mAP@0.75 obtained at the bounding box level in our validation set are listed in Table 2.

From these results, it is obvious that additional data augmentation improves the performance in all cases, which is not surprising given the small training dataset. A significant boost is observed whenever the pipeline for synthesizing defects is enabled, indicating that 2D image processing routines provide an adequate way to generate realistic images of damage. The copy-paste augmentation shows some weak gains, even if mainly the challenging samples from the training set are repeated. However,

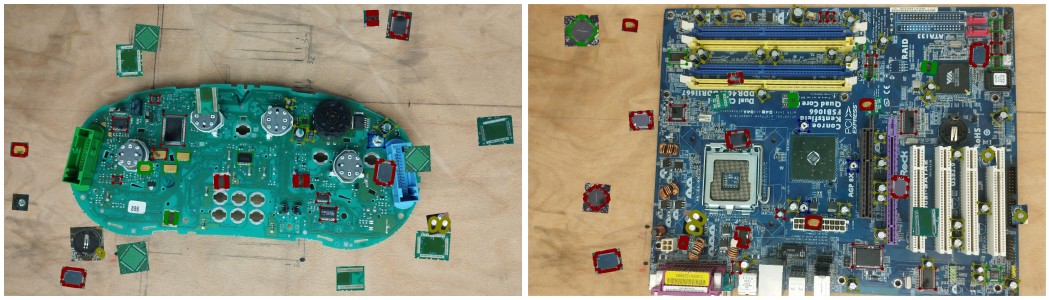

Figure 4: Samples of typical PCBs with several components that were modified using our proposed data augmentation steps. Both our copy-and-paste augmentation and the defect synthesis are enabled and significantly increase the available training samples of defects.

with additional out-of-distribution samples pasted into existing training images, additional gains are consistently achieved, indicating that this is indeed a viable way of extending the training data via online feedback. With the already strong pipeline for synthesizing defects enabled, the additional gains of the copy-paste augmentation only manifest at the mAP@0.5 level. Note that the library of source patches in our copy-paste augmentation already focuses on typical mistakes and challenging samples, that we identified in prior experiments. Some examples of success stories illustrating the benefits of our augmentation schemes are also given in Figure 6.

Comparing the different models, ConvNeXt [9] appears to perform marginally better than Dino [21] and Mask2Former [2] in both the mAP@0.5 and mAP@0.75 metrics, while RTMDet [11] being optimized for speed is trailing behind by a larger margin. In all cases, the proposed data augmentation strategies improve the mAP scores up to a top value of 93.1 for mAP@0.5, thus making the automatic PCB inspection system a viable approach in the intended use case. Some remaining failure cases are shown in Figure 7. We believe some of these could be addressed by collecting additional samples and injecting them into the copy-paste library. However, these are hard to find in our small training set of 25 images. It should also be noted that the presented detectors already generated valuable feedback on annotation errors for the domain experts who prepared the dataset and perform at least on par with human annotators.

6 Conclusions

In this paper, we consider techniques for applying machine learning in use cases with low-volume datasets. Furthermore, the data in our use cases have very specific characteristics, and approaches such as pre-training on web images have little chance of covering the specific defects we are interested in. This is a typical scenario in many industrial vision applications, where datasets show only few defects, that are also heterogeneous. Since each failure case is represented only by very few samples, conventional training methods show only limited success.

We proposed two data augmentation strategies to counter the data scarcity in the application of PCB component detection and inspection. As a first strategy, defects can be synthesized on the intact samples by image transformations. As the second strategy, we propose to create a library of challenging samples and paste them into the training images. This provides a viable approach for incorporating feedback and out-of-distribution samples into training with little manual effort.

Overall, these techniques significantly improve the performance of a detector on the given PCB dataset. They can also be used as a blueprint for similar applications with low-volume datasets.

Acknowledgments and Disclosure of Funding

This work has been supported by the project European Lighthouse to Manifest Trustworthy and Green AI (ENFIELD), which has received funding from the European Union's Horizon Europe research and innovation program under grant agreement No. 101120657.

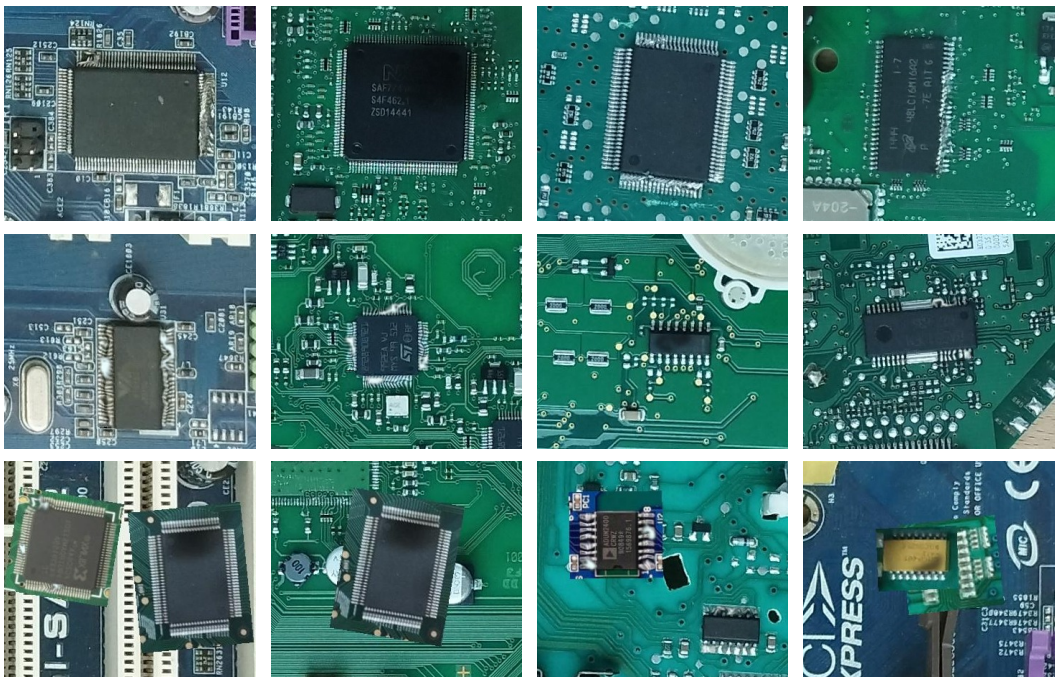

Figure 5: Closeup illustration of real defects (top row), synthesized defects (middle row) and synthesized defects on copy-and-paste augmented samples (bottom row) generated with our proposed data augmentation schemes. While close observation will reveal the synthetic nature of some of the images, they appear convincing and well suited to extend the very limited training set.

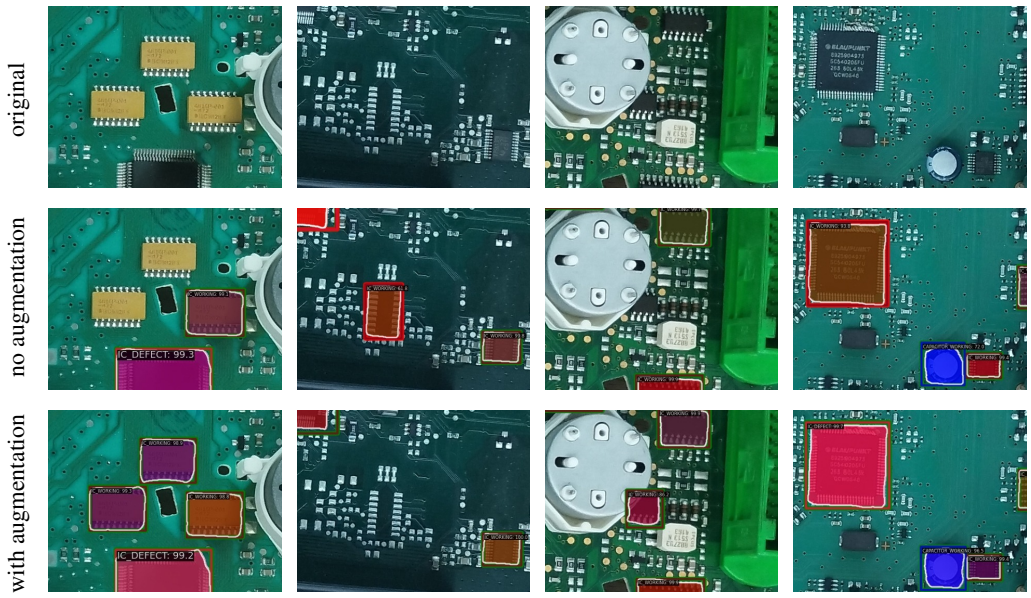

Figure 6: Illustrations of success stories, where the data augmentation schemes improved detection results. Left to right: Additional samples of yellow ICs in the copy-paste library improve their detection, samples of unpopulated IC areas reduce false positives, samples of partly occluded ICs are also included in the copy-paste library and example of a slightly damaged IC, that is correctly classified with our augmentation scheme.

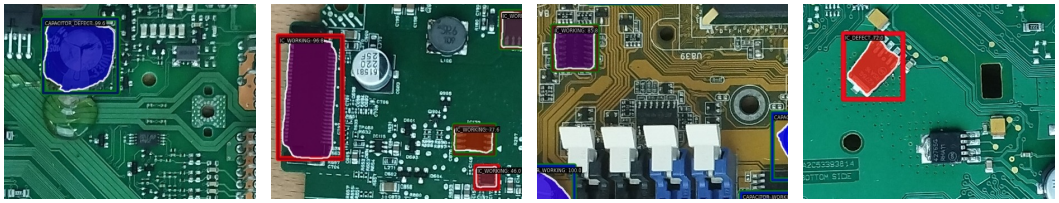

Figure 7: Samples of remaining failure cases: ICs of odd shapes are occasionally missed, in some cases, connectors are mistaken for ICs, not all occlusions can be handled and often, small MOSFETs or other irrelevant components are identified as ICs.

References

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

[2] Cheng, B., Misra, I., Schwing, A. G., Kirillov, A., and Girdhar, R. (2022). Masked-attention mask transformer for universal image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1290–1299.

[3] Dwibedi, D., Misra, I., and Hebert, M. (2017). Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1301–1310.

[4] Ghiasi, G., Cui, Y., Srinivas, A., Qian, R., Lin, T.-Y., Cubuk, E. D., Le, Q. V., and Zoph, B. (2021). Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR*, pages 2918–2928.

[5] Guo, Q., Wang, S., Chang, C., and Rambach, J. (2025). Cacp: Context-aware copy-paste to enrich image content for data augmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR) Workshops*, pages 5177–5186.

[6] Jain, S., Seth, G., Paruthi, A., Soni, U., and Kumar, G. (2022). Synthetic data augmentation for surface defect detection and classification using deep learning. *Journal of Intelligent Manufacturing*, 33(4):1007–1020.

[7] Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A. C., Lo, W.-Y., et al. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026.

[8] Kuo, C.-W., Ashmore, J. D., Huggins, D., and Kira, Z. (2019). Data-efficient graph embedding learning for pcb component detection. In *2019 IEEE winter conference on applications of computer vision (WACV)*, pages 551–560. IEEE.

[9] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., and Xie, S. (2022). A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11976–11986.

[10] Lu, H., Mehta, D., Paradis, O., Asadizanjani, N., Tehranipoor, M., and Woodard, D. L. (2020). FICS-PCB: A multi-modal image dataset for automated printed circuit board visual inspection. Cryptology ePrint Archive, Paper 2020/366.

[11] Lyu, C., Zhang, W., Huang, H., Zhou, Y., Wang, Y., Liu, Y., Zhang, S., and Chen, K. (2022). Rtmdet: An empirical study of designing real-time object detectors.

[12] Mahalingam, G., Gay, K. M., and Ricanek, K. (2019). Pcb-metal: A pcb image dataset for advanced computer vision machine learning component analysis. In *16th International Conference on Machine Vision Applications (MVA)*, pages 1–5. IEEE.

[13] Nikolenko, S. I. (2021). *Synthetic data for deep learning*, volume 174 of *Springer Optimization and Its Applications*. Springer.

[14] Phoulady, A., Suleiman, Y., Choi, H., Moore, T., May, N., Shahbazmohamadi, S., and Tavousi, P. (2023). Synthetic data augmentation to enhance manual and automated defect detection in microelectronics. *Microelectronics Reliability*, 150:115220. Special issue of 34th European Symposium on Reliability of Electron Devices, Failure Physics and Analysis (ESREF 2023).

[15] Pižurica, N., Milović, N., Jovancevic, I., Nasirimajd, A., and Quadrini, W. (2025). Epid: The enfield pcb inspection dataset for visual defect detection. Zenodo, https://doi.org/10.5281/zenodo.16811808.

[16] Pramerdorfer, C. and Kampel, M. (2015). A dataset for computer-vision-based pcb analysis. In *14th IAPR international conference on machine vision applications (MVA)*, pages 378–381. IEEE.

[17] Ren, W., Song, K., Chen, C.-y., Chen, Y., Hong, J., Fan, M., Ouyang, X., Zhu, Y., and Xiao, J. (2025). Dd-aug: A knowledge-to-image synthetic data augmentation pipeline for industrial defect detection. *IEEE Transactions on Industrial Informatics*, 21(3):2284–2293.

[18] Saif, S. S., Aras, K., and Giuseppi, A. (2022). Automated optical inspection for printed circuit board assembly manufacturing with transfer learning and synthetic data generation. In *2022 30th Mediterranean conference on control and automation (MED)*, pages 318–323. IEEE.

[19] Simaei, E. and Rahimifard, S. (2024). Ai-based decision support system for enhancing end-of-life value recovery from e-wastes. *International Journal of Sustainable Engineering*, 17(1):80–96.

[20] Singh, K., Kharche, S., Chauhan, A., and Salvi, P. (2024). Pcb defect detection methods: A review of existing methods and potential enhancements. *Journal of Engineering Science & Technology Review*, 17(1).

[21] Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L., and Shum, H.-Y. (2023). DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *The Eleventh International Conference on Learning Representations*.