

A Review of Deep Learning-Based Methods for Industrial Defect Detection

Zida Bian

Electrical and Information Engineering, Changsha University of Science and Technology, Changsha
410114, China
422353812@qq.com

ABSTRACT

Industrial visual inspection has evolved from methods based on handcrafted features to data-driven approaches based on convolutional detectors, detection Transformers, and vision-language models. This review classifies representative methods according to task output and training-data availability and compares two-stage detectors, YOLO-style one-stage detectors, DETR-style end-to-end detectors, normal-only anomaly detection, few-normal-shot learning, and zero-shot approaches. All claims and quantitative results are drawn from verifiable primary publications. The review identifies the applicability limits of each method family with respect to small defects, class imbalance, domain shift, and edge deployment, and provides deployment-oriented recommendations for model selection and evaluation.

Keywords: Industrial Defect Detection; Convolutional Neural Network; Object Detection; Transformer; Anomaly Detection

1. INTRODUCTION

Industrial defect detection extends beyond binary discrimination between defective and defect-free products: within a constrained production cycle, a system may also need to identify defect categories, localize defects, or delineate pixel-level contours. Industrial images are commonly characterized by small defects, severe class imbalance, repetitive background textures, and imaging conditions that vary across production lines. Consequently, results achieved by general-purpose vision models on industrial datasets do not, by themselves, demonstrate readiness for on-site deployment. Miss rate, false-alarm rate, inference latency, and out-of-domain robustness must be assessed jointly.

Deep learning models jointly optimize feature extraction and task-specific objectives. AlexNet demonstrated the representation capability of CNNs ^[1], while Faster R-CNN and YOLOv3 are representative two-stage and one-stage detection frameworks, respectively ^[2,3]. DETR formulates detection as set prediction ^[4], and Deformable DETR and RT-DETR further improve convergence, small-object detection, and real-time performance ^[5,6]. In anomaly detection, MVTec AD has become a widely used benchmark for methods trained on normal samples ^[7], whereas PatchCore ^[8] and WinCLIP ^[9] represent local feature-memory and vision-language transfer approaches, respectively.

Rather than cataloging models by name, this review organizes methods according to their annotation requirements, output forms, and deployment costs. Performance figures that cannot be verified from the original publications are excluded, and results obtained on different datasets, at different input resolutions, or on different hardware are not directly ranked. The contributions are threefold: (1) a unified taxonomy encompassing supervised defect detection and anomaly detection; (2) a summary of the applicability limits of the principal technical routes; and (3) practical evaluation and model-selection principles for real production lines.

2. METHOD CLASSIFICATION CRITERIA

We organize the methods along two dimensions: task output and training-data availability. In terms of output, the methods are divided into image classification, object detection, and pixel-level segmentation. Learning settings are then distinguished according to two criteria: whether target-category training images are available and whether real anomalous samples are used during training. This distinction yields four regimes: fully supervised learning, normal-only anomaly detection, few-normal-shot anomaly detection, and zero-shot anomaly detection. Models in the latter three regimes may still rely on external pretraining data. Because studies differ in their datasets, input scales, hardware platforms, and metric definitions, reported performance values are directly comparable only under identical experimental conditions.

Accordingly, our comparison considers annotation cost, output granularity, recall, and end-to-end latency in addition to predictive performance.

The resulting data-availability taxonomy is summarized in Table 1.

Table 1. Taxonomy based on target-category data availability.

Learning setting	Target-category training images	Real anomalous samples	Representative methods
Fully supervised	Available	Used	Faster R-CNN ^[2] , YOLOv3 ^[3] , DETR ^[4] , U-Net ^[10]
Normal-only anomaly detection	Normal images available	Not used	PaDiM ^[11] , DRAEM ^[12] , PatchCore ^[8]
Few-normal-shot anomaly detection	A few normal images	Not used	PatchCore ^[8] , WinCLIP ^[9] , PromptAD ^[13]
Zero-shot anomaly detection	Not available	Not used	WinCLIP ^[9]

3. MAJOR TECHNICAL ROUTES

3.1 Fully Supervised Defect Recognition

3.1.1 Image Classification

When workpieces appear at fixed positions and candidate defect regions can be cropped in advance through mechanical positioning or predefined rules, defect inspection can be formulated as an image-classification task. AlexNet demonstrated that deep CNNs can learn discriminative hierarchical representations from data ^[1], whereas ResNet uses residual mappings to alleviate the optimization difficulties associated with very deep networks ^[14]. Classification requires less annotation effort than bounding-box or pixel-level labeling; however, it can only determine whether an input region is anomalous and assign a category, without directly localizing the defect. This approach is therefore appropriate for fixed-view, single-workpiece settings in which defect locations are constrained, but it should not replace object detection or segmentation when precise localization is required.

3.1.2 CNN-Based Object Detection

Two-stage detectors first generate region proposals and then perform classification and bounding-box regression. Faster R-CNN integrates proposal generation into end-to-end network training and generally offers stable localization accuracy, although region-proposal processing increases inference overhead ^[2]. Mask R-CNN further introduces a mask branch and employs RoIAlign to reduce feature-alignment errors, enabling bounding boxes and defect contours to be predicted simultaneously ^[15]. For small targets such as microcracks and pinholes, feature pyramid networks fuse semantic information across scales through top-down pathways and lateral connections, and have consequently become common components of multi-scale detectors ^[16].

One-stage detectors predict classes and bounding boxes directly at dense spatial locations, thereby eliminating the proposal stage. As illustrated in Fig. 1, YOLOv3 combines the Darknet-53 backbone with prediction heads at three scales and provides an influential baseline for balancing speed and accuracy ^[3]. The residual connections in Darknet-53 originate from deep residual learning and alleviate the optimization difficulties of deep networks ^[14]. In industrial applications, modifications to YOLO-style models typically seek to preserve high-resolution shallow features, strengthen cross-scale fusion, adjust positive and negative sample assignment, and compress the network. Nevertheless, a newer version number or an additional module does not automatically imply an advantage across datasets; mAP values obtained with different data splits, image sizes, or evaluation scripts are not directly comparable.

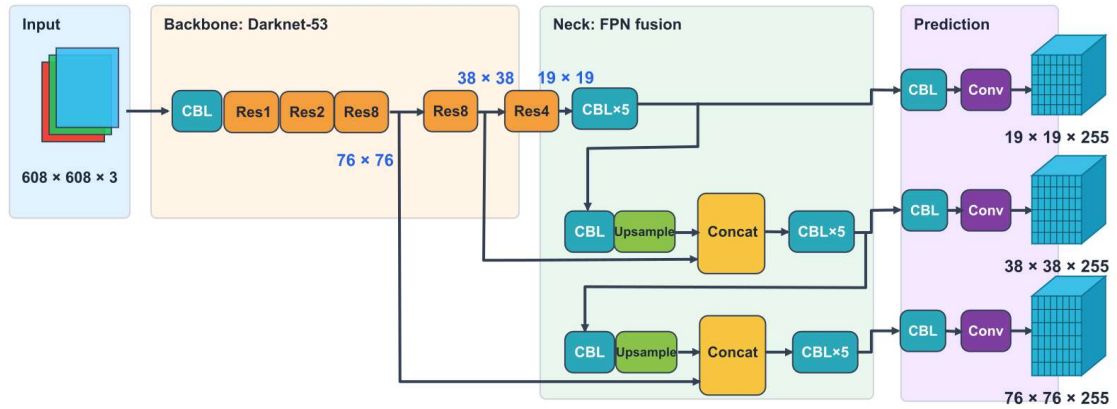


Figure 1. Darknet-53 backbone, multi-scale feature fusion, and three-scale prediction architecture of YOLOv3.

3.1.3 Transformer-Based Object Detection

As illustrated in Fig. 2, the Transformer comprises an encoder-decoder architecture built from self-attention and feed-forward networks^[17], whereas ViT partitions an image into a sequence of patches to enable global representation learning^[18]. DETR performs set prediction through object queries and bipartite matching, eliminating both anchors and non-maximum suppression^[4]; however, it converges slowly and incurs substantial computational cost at high resolutions. Deformable DETR employs sparse sampling and multi-scale features, reducing the required number of training epochs to approximately one-tenth of that for DETR while improving small-object detection^[5]. RT-DETR adopts an efficient hybrid encoder and a query-selection mechanism; the original paper reports 53.1 AP and 108 FPS for the R50 model on an NVIDIA T4 GPU^[6].

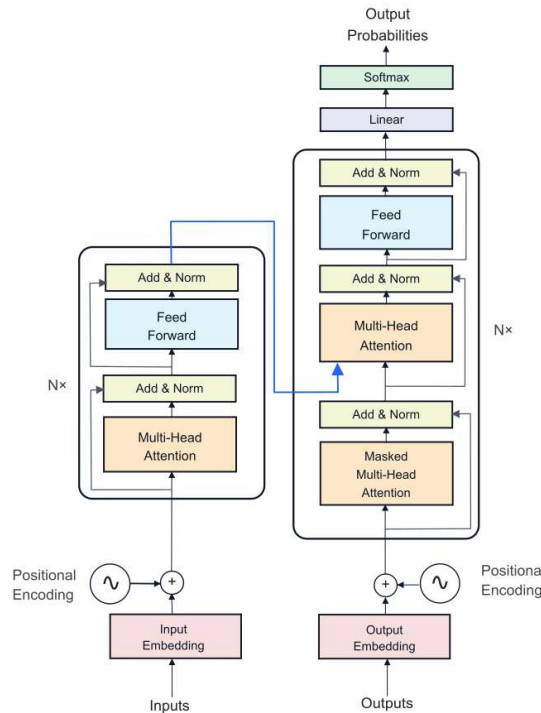


Figure 2. General Transformer encoder-decoder architecture.

3.1.4 Pixel-Level Segmentation

U-Net combines localization details with high-level semantic information through an encoder-decoder architecture and skip connections, making it a foundational architecture for pixel-level defect segmentation^[10]. When defect boundaries determine quality grades or process decisions, segmentation provides richer information than bounding boxes. Its disadvantages include the high cost of pixel-level annotation and the tendency of thin cracks to become disconnected after downsampling. In engineering applications, region-overlap and boundary metrics should be reported together, and any changes in the true defect dimensions introduced by morphological post-processing should be examined.

No fully supervised architecture is universally superior. Classification requires relatively simple annotations but provides no localization; two-stage detectors offer stable localization but are comparatively slow; YOLO-style models are better suited to real-time inspection; DETR-based methods reduce manually designed post-processing while modeling global relationships; and segmentation models provide the finest-grained morphological information. Model selection should therefore proceed from the required output: classification for pass/fail decisions, object detection for localization during automated rejection and reinspection, and segmentation for measuring crack length, defect area, or severity.

3.2 Normal-Only Anomaly Detection

In the normal-only setting, target-category normal images are available for training, whereas real anomalous samples are not used. MVTec AD comprises 15 categories of industrial objects and textures and provides pixel-level anomaly annotations for evaluation, making it a widely used benchmark for industrial anomaly detection^[7]. PaDiM extracts multi-level patch features from a pretrained CNN and fits a multivariate Gaussian distribution at each spatial location^[11]. DRAEM jointly trains reconstruction and discriminative networks using synthetic rather than real anomalies, thereby enabling direct anomaly localization^[12]. PatchCore extracts local, intermediate-level features from a pretrained network, constructs a representative memory bank of normal features, and derives anomaly scores from nearest-neighbor distances^[8].

The original PatchCore paper reports an image-level AUROC of up to 99.6% on MVTec AD^[8]. These results should be interpreted under the normal-only training condition: real anomalous samples are excluded from training even when anomaly labels are available for benchmark evaluation. Although this regime reduces dependence on real defects, memory requirements, synthetic-anomaly bias, threshold transferability, and reliance on external pretraining must still be evaluated and reported.

3.3 Few-Normal-Shot Anomaly Detection

Few-normal-shot methods seek to adapt using only a small number of target-category normal images and no real anomalous samples. In addition to its standard normal-only setting, the original PatchCore paper reports competitive performance under few-shot conditions^[8]. WinCLIP investigates a few-normal-shot setting in which normal images complement the information conveyed by textual prompts^[9]. PromptAD likewise learns prompts using only a limited number of normal samples and therefore belongs to the few-normal-shot regime rather than the zero-shot regime^[13]. These approaches lower the data requirements during new-product introduction but are more sensitive to sample representativeness, pretraining domain bias, and threshold selection. Results obtained from a small random sample cannot substitute for cross-batch validation.

3.4 Zero-Shot Vision-Language Anomaly Detection

In the zero-shot setting, no target-category training images or real anomalous samples are available. CLIP learns transferable representations from large-scale image-text pairs, providing a foundation for open-vocabulary recognition^[19]. WinCLIP combines state words, prompt templates, and window-level features to perform zero-shot anomaly classification and segmentation^[9]. On MVTec AD, WinCLIP reports zero-shot image-level and pixel-level AUROC values of 91.8% and 85.1%, respectively; however, its supplementary material notes that a high pixel-level AUROC may still be accompanied by a low F1 score. Zero-shot vision-language methods are therefore better suited to cold-start inspection and human-assisted review than to replacing production-line validation.

The advantages and limitations of the principal methods are compared in Table 2.

Table 2. Engineering comparison of the principal method families.

Method family	Training data	Output	Main advantages	Main limitations
Two-stage CNN	Bounding-box/mask annotations	Boxes or masks	Stable localization and mature components	Relatively slow and complex to deploy
YOLO-style one-stage CNN	Bounding-box annotations	Bounding boxes	Mature real-time performance and tooling	Requires dedicated optimization for small objects and domain shift
DETR family	Bounding-box annotations	End-to-end object set	Reduces anchor/NMS design and provides global modeling	Training cost and high-resolution computational burden
U-Net-style segmentation	Pixel-level annotations	Pixel masks	Supports area, boundary, and severity analysis	High annotation cost; thin cracks may become disconnected
PatchCore-style anomaly detection	Target-category normal images; no real anomalies	Anomaly score/heatmap	Suitable for cold starts and unknown anomalies	Memory retrieval, thresholding, and domain-shift issues
WinCLIP-style vision-language methods	No or a few target-category normal images; no real anomalies	Anomaly score/heatmap	Open-vocabulary and few-shot transfer	Prompt sensitivity and limited pixel-level accuracy

4. TYPICAL SCENARIOS AND METHOD SELECTION

4.1 Steel Surfaces

Steel-surface defects exhibit strong texture directionality, wide scale variation, and substantial inter-class overlap in appearance. When defect categories are fixed and annotations are sufficient, a one-stage detector equipped with multi-scale features can deliver high throughput. If crack length or defect area must be quantified, segmentation should be used instead. Experimental reports should provide class-wise recall, with particular attention to thin cracks and low-contrast defects, rather than reporting only overall mAP.

4.2 Printed Circuit Boards

PCB inspection commonly benefits from standard templates and stable imaging conditions. These characteristics make it well suited to approaches that combine image registration, image differencing, or rule-based priors with deep models. Open circuits, short circuits, and missing holes may occupy only a few pixels; consequently, input resolution, sliding-window strategies, and the preservation of shallow features are often more important than increasing network depth. Practical acceptance testing should distinguish component-level misses from pixel-level localization errors and should cover variations in production batch, surface reflection, and board color.

4.3 Photovoltaic Modules and Electroluminescence Images

Hidden cracks, broken fingers, and fragments in photovoltaic cells appear as elongated or low-contrast structures in electroluminescence images. Detection or segmentation networks can be used when defect categories are fixed and annotations are sufficient. During new-product introduction, or when anomaly types change continually, a normal-sample anomaly detector can first screen the images, after which a supervised model or human inspector can perform classification. Because imaging devices, exposure settings, and module types introduce substantial domain differences, studies should conduct cross-device testing rather than rely solely on random splits of images from the same production batch.

Table 3 summarizes the recommended methods for different production-line conditions.

Table 3. Model-selection recommendations for production-line conditions.

Condition	Preferred approach	Acceptance criteria
Stable classes, sufficient box annotations, strict cycle time	Lightweight one-stage CNN	Recall at a fixed false-alarm rate, end-to-end latency, and long-term drift
Precise area or boundary estimates required	U-Net- or Mask R-CNN-style segmentation	Boundary F1, connectivity of thin cracks, and dimensional error
Complex object relationships and sufficient training resources	Deformable DETR/RT-DETR	Small-object AP, training resources, and stability across resolutions
Scarce anomalous samples and frequent product switching	PatchCore-style normal-only anomaly detection	Threshold transfer, memory overhead, and recall of unknown anomalies
Zero-shot exploration or human-review assistance	WinCLIP-style vision-language method	Prompt robustness, F1/PR metrics, and out-of-domain false alarms

5. KEY CHALLENGES AND EVALUATION RECOMMENDATIONS

First, class imbalance renders overall accuracy difficult to interpret. Class-wise recall, precision-recall curves, and miss rates under a fixed false-alarm budget should be reported, and the respective contributions of sampling strategies, loss reweighting, and data synthesis should be distinguished. Second, domain shift is a major source of performance degradation after deployment. External testing should span production lines, devices, material batches, or time periods, while normal samples should be retained for post-deployment drift monitoring. Third, small-defect detection is not solely a network-architecture problem; it also depends on imaging resolution, optical contrast, and annotation consistency. Before algorithms are compared, it is necessary to confirm that the defects occupy a sufficient number of pixels in the input images.

Fourth, end-to-end latency should be measured across image acquisition, preprocessing, inference, post-processing, and communication. Fifth, safety-critical applications should incorporate rejection and reinspection mechanisms. Sixth, repeated tuning on public benchmarks may introduce optimistic bias. Real-IAD contains approximately 150,000 multi-view images spanning 30 object categories, and its authors observe that AUROC values above 99% on mainstream small-scale benchmarks no longer distinguish methods effectively [20]. Data splits, threshold-selection rules, and hardware configurations should therefore be disclosed, and both cross-device and cross-batch tests should be included.

6. CONCLUSION

No single model is suitable for every industrial defect-detection scenario. Supervised CNNs offer a mature trade-off between performance and cost; DETR-based methods enhance end-to-end prediction and global representation modeling; and normal-only anomaly detection and vision-language methods alleviate defect scarcity and cold-start challenges associated with new products. Model selection should jointly consider annotation cost, output granularity, real-time performance, and out-of-domain robustness. Reliable conclusions must be grounded in traceable publications and comparable experimental conditions.

REFERENCES

- [1] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* 25 (pp. 1097–1105)
- [2] Ren, S., He, K., Girshick, R., & Sun, J. (2017). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6), 1137–1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- [3] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv:1804.02767*
- [4] Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. In *Computer Vision – ECCV 2020* (pp. 213–229). Springer

- [5] Zhu, X., Su, W., Lu, L., Li, B., Wang, X., & Dai, J. (2021). Deformable DETR: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*
- [6] Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., & Chen, J. (2024). DETRs beat YOLOs on real-time object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 16965–16974)
- [7] Bergmann, P., Fauser, M., Sattlegger, D., & Steger, C. (2021). The MVTec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129, 1038–1059. <https://doi.org/10.1007/s11263-020-01400-4>
- [8] Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., & Gehler, P. (2022). Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 14318–14328)
- [9] Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., & Dabeer, O. (2023). WinCLIP: Zero-/few-shot anomaly classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 19606–19616)
- [10] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015* (pp. 234–241). Springer
- [11] Defard, T., Setkov, A., Loesch, A., & Audigier, R. (2021). PaDiM: A patch distribution modeling framework for anomaly detection and localization. In *Pattern Recognition: ICPR International Workshops and Challenges* (pp. 475–489). Springer. https://doi.org/10.1007/978-3-030-68799-1_35
- [12] Zavrtanik, V., Kristan, M., & Skočaj, D. (2021). DRAEM—A discriminatively trained reconstruction embedding for surface anomaly detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 8330–8339)
- [13] Li, X., Zhang, Z., Tan, X., Chen, C., Qu, Y., Xie, Y., & Ma, L. (2024). PromptAD: Learning prompts with only normal samples for few-shot anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 16838–16848)
- [14] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778)
- [15] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE International Conference on Computer Vision* (pp. 2961–2969)
- [16] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2117–2125)
- [17] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems* 30
- [18] Dosovitskiy, A., Beyer, L., Kolesnikov, A., et al. (2021). An image is worth 16×16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*
- [19] Radford, A., Kim, J. W., Hallacy, C., et al. (2021). Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning* (Vol. 139, pp. 8748–8763). PMLR
- [20] Wang, C., Zhu, W., Gao, B.-B., Gan, Z., Zhang, J., Gu, Z., Qian, S., Chen, M., & Ma, L. (2024). Real-IAD: A real-world multi-view dataset for benchmarking versatile industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 22883–22892)