RF-DETR-Large for Nine-Class PCB Surface Defect Detection: An Empirical Evaluation of RF-DETR-Large on DsPCBSD+ with Three-Stage Continuation Training

Tung-Chu Bai[1,2 *] Sheng-Cheng Yeh[2 †]

[1] Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan
[2] Department of Artificial Intelligence Applications, Ming Chuan University, Taoyuan, Taiwan

[*] Corresponding author. Email: 11557083@mail.ntou.edu.tw. ORCID: 0009-0006-3031-2373
[†] Advisor

Abstract

We report the best observed project result on the DsPCBSD+ nine-class PCB surface defect detection benchmark using RF-DETR-Large. Starting from a COCO-pretrained RF-DETR-L checkpoint, we apply a three-stage continuation training recipe: an initial retained stage at 640×640 for global epochs 1–3, a continuation at 640×640 with a learning rate of 1×10^{-4} through global epoch 7, and a final continuation at 640×640 with the learning rate reduced to 5×10^{-5} through global epoch 10. The best observed checkpoint at global epoch 9 (stage-3 epoch 2) reaches an EMA mAP@0.5:0.95 of **0.5411**, exceeding the project's YOLO26x 100-epoch baseline (0.4896) by **+0.0515 absolute** and **+10.5% relative**. This is a single-lineage validation result; no independent test estimate or uncertainty interval is claimed. The retained latency artifact reports a median of 35.3 ms per image on a single RTX 5090 with fp16 optimisation. The cited Zenodo record currently contains the paper source and PDF only; the local training artifacts are not represented as a complete public release.

1 Introduction

Printed circuit board (PCB) surface defect detection is a cornerstone quality-control task in electronics manufacturing. As PCB designs become denser and component counts rise into the hundreds per board, human visual inspection scales poorly, and small, low-contrast defects such as spurious copper, conducting filaments, and burnt marks are easy to miss in production-line throughput. The nine-class DsPCBSD+ benchmark [1] consolidates canonical PCB surface defects into a single detection target set with bounding box annotations and a fixed train/val split.

A wide spectrum of detectors has been applied to PCB surface defects. Convolutional single-stage detectors such as the YOLO family [2–4] are widely deployed because of their throughput and mature ecosystem, but they struggle with small and densely packed defects because of their anchor-based or anchor-free dense prediction heads. Transformer-based detectors [5, 6] improve localisation through set-based prediction and global attention, yet their data-hungry nature has historically limited transfer to medium-scale industrial benchmarks. More recent DETR variants [7–11] introduce denoising training, decoupled detection heads, multi-scale deformable attention, or DINOv2 backbones to close the convergence gap. Among these, RF-DETR [11] couples a windowed DINOv2 encoder with a four-layer DETR decoder and group-detr assignment, and is reported to converge faster than comparable DETR-family detectors on small object detection tasks. Figure 2 shows representative training images that illustrate the visual diversity of the dataset and the small spatial extent of most defects.

Two practical questions motivate this study. First, can a COCO-pretrained RF-DETR-L model exceed the project's YOLO26x baseline on DsPCBSD+ under a documented continuation-training recipe? Second, what can a single retained training lineage reveal about the stability of this recipe? We answer these questions with a single documented lineage; the results are intended as exploratory evidence about this project, not as an exhaustive literature or hyperparameter comparison.

This paper makes two concrete contributions. First, we document a three-stage continuation training recipe whose best observed project checkpoint reaches EMA mAP@0.5:0.95 = 0.5411 on DsPCBSD+, exceeding the unarchived YOLO26x internal reference by +0.0515 absolute (+10.5% relative). Second, we provide an evidence-bounded analysis of the dataset, training lineage, inference showcase, error cases, and deployment-related measurements. The cited public record contains the paper source and PDF; the retained local checkpoints and metrics are identified as project artifacts rather than claimed as a complete public release.

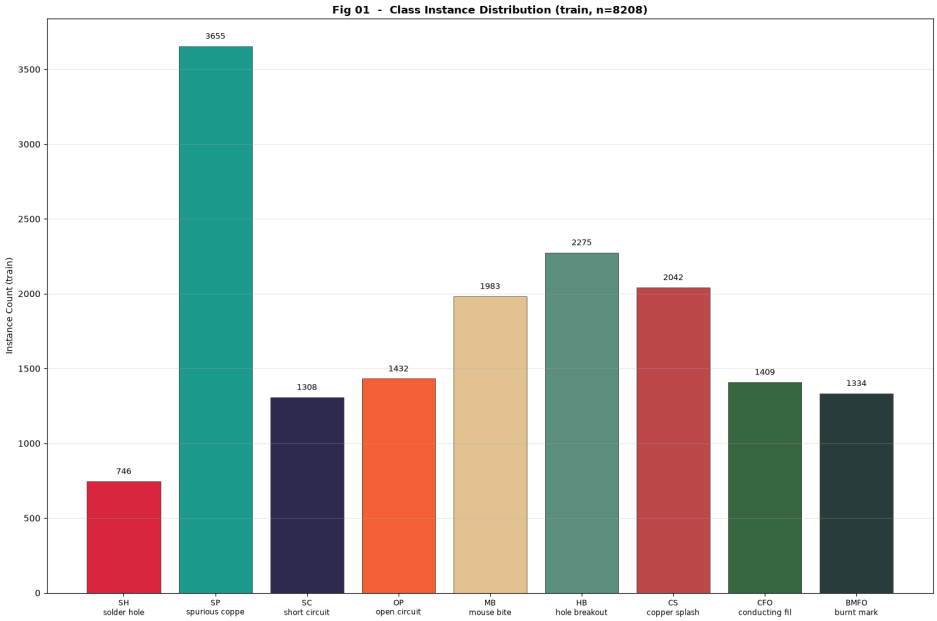


Figure 1. Class instance distribution across the 8,208-image training split (the full dataset contains 10,259 images, including 2,051 validation images). SP is the most frequent annotated defect class and SH is the rarest in the original training split. The 1.46× copy-paste augmentation pipeline only partially closes the gap, motivating detectors that are robust to long-tail class distributions.

2 Related Work

YOLO-family detectors for PCB defects. The YOLO lineage [2] is a useful reference point for real-time industrial inspection. YOLOv7 introduced extended ELAN [3], and later YOLO implementations continued to refine dense prediction and label assignment [4]. In this study, the YOLO26x 100-epoch configuration with default 640 resolution is used as the project baseline, reaching mAP@0.5:0.95 = 0.4896 on DsPCBSD+. This baseline is used for an internal comparison only; the paper does not claim a comprehensive comparison with all PCB detector variants.

Transformer-based detectors. DETR [5] replaced the anchor-based head with a set-based bipartite matching loss and is the canonical end-to-end object detector. It converges slowly because of unstable attention in the early training stages. Deformable DETR [6] introduced multi-scale deformable attention that accelerates convergence on small objects by restricting attention to a small set of sampling points per query. Cluster DETR [7] explored the cluster hypothesis for object queries. DINO [9] further combined denoising training with a contrastive denoising scheme and achieved strong results on COCO at convergence. D-Fine [10] reformulated fine-grained distribution refinement as a decoupled detection paradigm that improves box quality without end-to-end retraining. These models share two characteristics relevant to our task: (i) they rely on multi-scale features with strong backbone priors, and (ii) their convergence on medium-scale datasets is highly sensitive to learning rate and warm-up schedule. The latter point is particularly important for our setup because DsPCBSD+ has about 10k images in total (8.2k training and 2.1k validation), well below the 100k+ scale at which most DETR-family recipes are tuned.

RF-DETR and DINOv2-based encoders. RF-DETR [11] builds on a windowed DINOv2-Small encoder and a four-layer DETR decoder with group-detr assignment. We use this upstream architecture as the transfer-learning base and evaluate it on the medium-scale DsPCBSD+ split. The reported continuation recipe therefore tests a concrete implementation choice; it does not claim that DINOv2 transfer, group-detr assignment, or windowed attention alone causes the observed result.

Continuation training and multi-stage recipes. Continuation training, in which a model is resumed under a changed learning-rate regime, is a practical way to evaluate whether a trained DETR model can benefit from a controlled fine-tuning step. The motivation is that DETR-family detectors combine multiple loss components, including classification and box-regression terms, so a single learning-rate schedule may not be optimal throughout

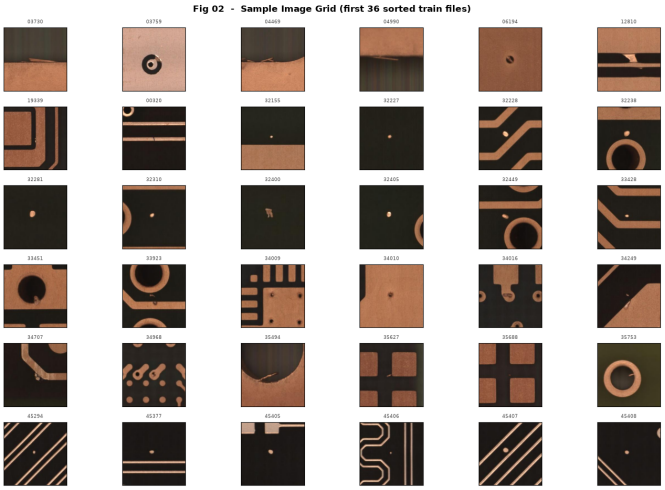
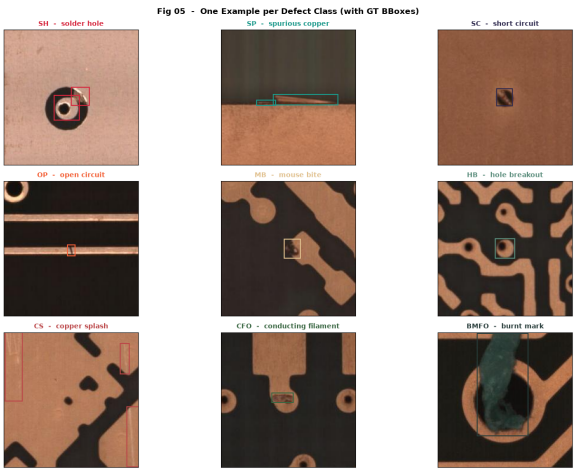


Figure 2. Sample image grid of the first 36 sorted training images. Native resolution is predominantly 226×226 px; 86 training and 25 validation images are 108×108 px. Images are upscaled for visibility. The dataset exhibits wide variation in copper density, lighting, and substrate texture, and most defects occupy a small fraction of the image area. This visual diversity is a major source of generalisation pressure for the detector.

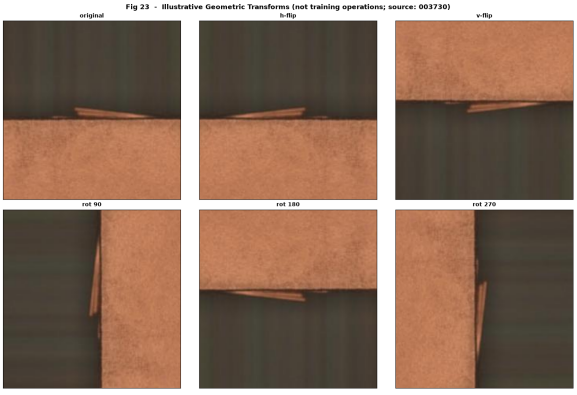
training. Our recipe is a three-stage variant applied to RF-DETR-Large; the stabilisation interpretation is treated as a hypothesis evaluated by the reported lineage rather than as a general literature result.

3 Dataset

We use DsPCBSD+ [1], a public nine-class PCB surface defect dataset. The classes are SH (solder hole), SP (spurious copper), SC (short circuit), OP (open circuit), MB (mouse bite), HB (hole breakout), CS (copper splash), CFO (conducting filament), and BMFO (burnt mark). Figure 3a shows one training image per class with ground-truth bounding boxes. The training split contains 8,208 images, predominantly at 226×226 pixels; 86 training images are 108×108 pixels. Bounding-box annotation follows the YOLO format (class, c_x, c_y, w, h in unit-interval coordinates), and the validation split contains 2,051 images, including 25 images at 108×108 pixels, with the same class taxonomy.



(a) One example per defect class with ground-truth bboxes.

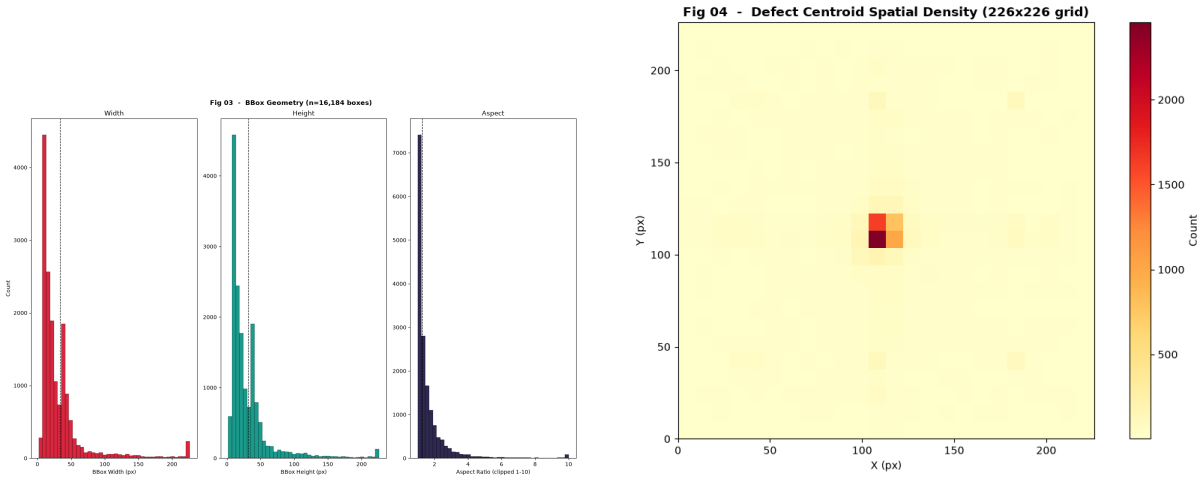


(b) Illustrative geometric transforms of one source image.

Figure 3. Class and augmentation examples. **Left**: nine training images, one per defect class, with ground-truth bboxes. **Right**: illustrative geometric transforms of a single source image; they are shown to visualise spatial changes and are not claimed as operations in the evaluated training recipe. The evaluated pipeline increases the effective training-set size by 1.46× through copy-paste rebalancing. Per-paste photometric processing uses CLAHE, gamma correction, and additive Gaussian noise.

A class-instance histogram (Fig. 1) shows severe imbalance: SP is the most common class and SH is the rarest in the original training split. The smallest median boxes are observed for SP and MB, whereas HB has substantially

larger boxes; BMFO is rare but is not one of the smallest-box classes. These descriptive relationships are not used as causal explanations for the per-class AP values in Fig. 11.



(a) Bounding-box geometry.

(b) Defect centroid spatial density.

Figure 4. Dataset geometry. **Left**: distribution of bounding-box width, height, and aspect ratio across the training set. Median bbox is approximately 21×20 px when each annotation is scaled by its source image dimensions. **Right**: centroid spatial density on a 24×24 grid over normalised image coordinates. Defects cluster near the centre because PCBs are framed with the active region centred, but a long tail of edge and trace defects is well represented. The small defect size relative to the canvas motivates the evaluated P4 projector scale.

The bounding-box geometry (Fig. 4a, left) shows that most defects are small and roughly square: the median width and height are approximately 21×20 px after scaling each annotation by its source dimensions, and the aspect ratio distribution is sharply peaked near 1.0. The centroid spatial density (Fig. 4b, right) confirms that the defect centroids are concentrated near the image centre, with a long tail extending to the edges. This geometry informs one design choice later: the detection head is configured for the P4 projector scale (roughly 32×32 features at 640 px input). The evaluated inference path uses single-image predictions without geometric test-time augmentation.

A copy-paste augmentation pipeline (Fig. 3b, right) increases the effective training-set size by $1.46\times$ without altering the validation distribution. The training manifest contains 8,208 original images plus 3,764 generated composites, for 11,972 training images in total. The augmentation script uses copy-paste rebalancing of rare-class ROIs into eligible background images. Each pasted ROI may receive CLAHE, gamma correction with $\gamma \in [0.7, 1.3]$, and additive Gaussian noise with $\sigma \in [5, 15]$. The target is determined from running instance counters rather than from whole-image duplication.

4 Method

4.1 Model: RF-DETR-Large

We adopt RF-DETR-Large [11] as the detector. The model has a windowed DINOv2-Small encoder with output features at depths 3, 6, 9, and 12, a hidden dimension of 256, a four-layer DETR decoder with two-stage refinement, 300 object queries, group-detr assignment with k=13 groups, and projector scale P4. The classification head is re-initialised for nine classes from the 90-class COCO pretraining weights, while the encoder and box head retain their pretraining. Bounding-box reparameterisation and lite-refpoint-refine are enabled, following the RF-DETR configuration. The model uses automatic mixed precision; the retained configurations record `amp_dtype: auto`, so the runtime-selected dtype is not asserted. EMA decay is set to 0.993 with tau 100, and the EMA checkpoint selected by validation mAP is used for deployment.

The full three-stage pipeline is illustrated in Fig. 5. The pipeline is the same for the retained v8, v9, and v11a lineage; the unretained v11b and v12 project notes are not used as quantitative evidence.

4.2 Three-stage continuation training recipe

Training is decomposed into three stages.

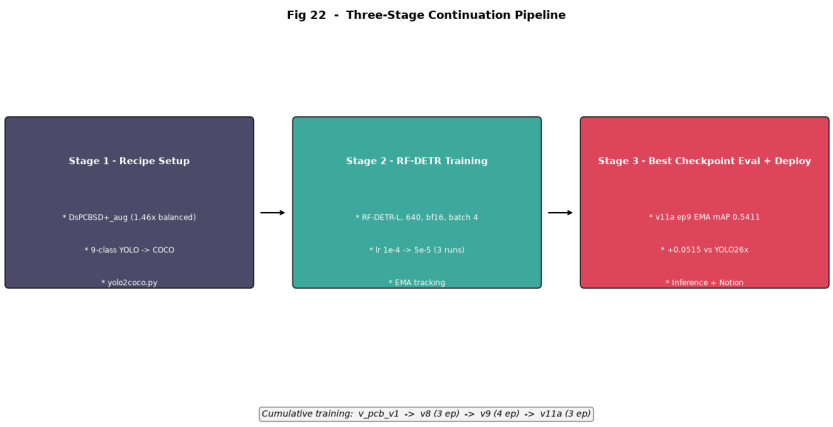


Figure 5. Three-stage continuation pipeline: data preparation, RF-DETR training, and best-checkpoint evaluation plus deployment. The retained lineage runs v8 at 640 resolution for global epochs 1–3, v9 at 640 for global epochs 4–7, and v11a at 640 for global epochs 8–10, producing the v11a ep9 best observed checkpoint.

Retained lineage schedule. The retained artifacts do not preserve a separate 704-resolution run configuration. They do preserve the executed continuation sequence: v8 contains global epochs 1–3 at 640, v9 continues global epochs 4–7 at 640 with learning rate 1×10^{-4}, and v11a continues global epochs 8–10 at 640 with learning rate 5×10^{-5}.

Stage 2 (640, learning rate 1×10^{-4}). The retained v9 continuation covers global epochs 4–7 at 640×640 with batch size 4, gradient accumulation 4, automatic mixed precision, and learning rate 1×10^{-4}. It reaches EMA mAP@0.5:0.95 = 0.5403 at global epoch 7.

Stage 3 (640, learning rate 5×10^{-5}). The retained v11a continuation covers global epochs 8–10 at 640×640 with the learning rate reduced to 5×10^{-5}. The best observed retained checkpoint is v11a at global epoch 9 (stage-3 epoch 2), reaching EMA mAP@0.5:0.95 = **0.5411**.

4.3 Loss and evaluation

Training minimises a weighted sum of three objectives: the model's classification loss on class labels, L1 regression on bounding-box coordinates, and GIoU [13] regression. The class-loss coefficient is 1.0 and the box losses are weighted according to the rfdetr default (5.0 for L1, 2.0 for GIoU). Gradient clipping is applied at 0.1. Evaluation uses the COCO [12] protocol implemented via pycocotools, reporting mAP@0.5:0.95, mAP@0.5, mAP@0.75, recall, and precision. We report the EMA mAP@0.5:0.95 as the headline metric because it is the most stable indicator under the small per-epoch fluctuations observed in stages 2 and 3. For deployment, we use the EMA checkpoint selected as the project's best validation checkpoint.

5 Experiments

5.1 Experimental setup

All experiments are conducted on a single workstation with an NVIDIA RTX 5090 (32 GB, Blackwell SM 120), Python 3.12, PyTorch 2.11 with CUDA 12.8, and rfdetr $\geq$ 1.8. The model uses automatic mixed precision; the retained configurations record `amp_dtype: auto`. Confidence is 0.30 for inference showcases and 0.001 for confidence and per-class sweeps. Each training run uses the recipe described in Section 4; the retained evidence is restricted to the v8, v9, and v11a lineage.

5.2 Stage-wise training lineage

Figure 6 plots the EMA and raw mAP@0.5:0.95 across the three stages. The 0.4896 YOLO26x baseline is exceeded by the end of stage 2. Stage 3 adds a small observed EMA change (+0.0008 from 0.5403 to 0.5411) in this single training lineage, while raw mAP oscillates between 0.5225 and 0.5278.

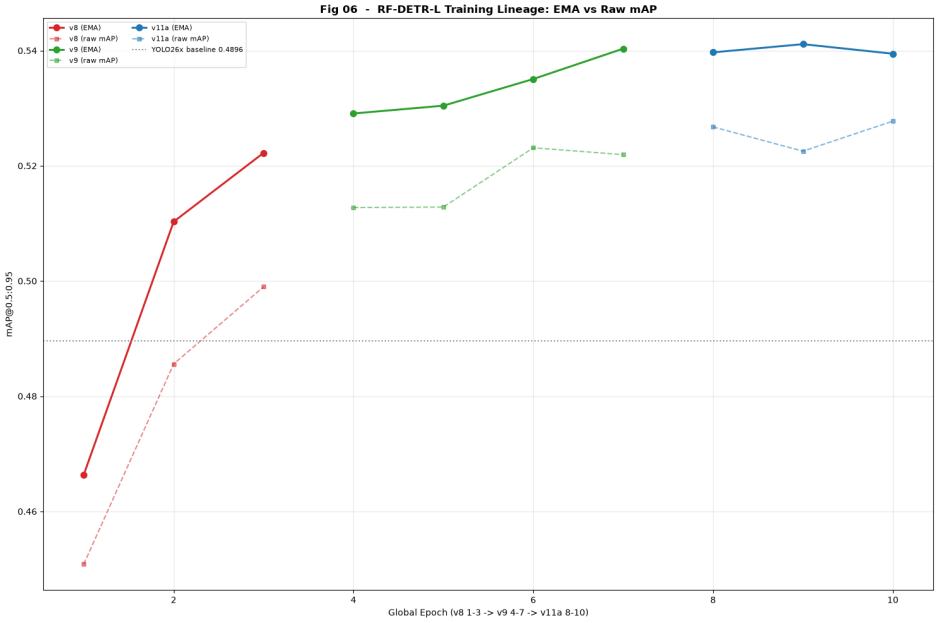
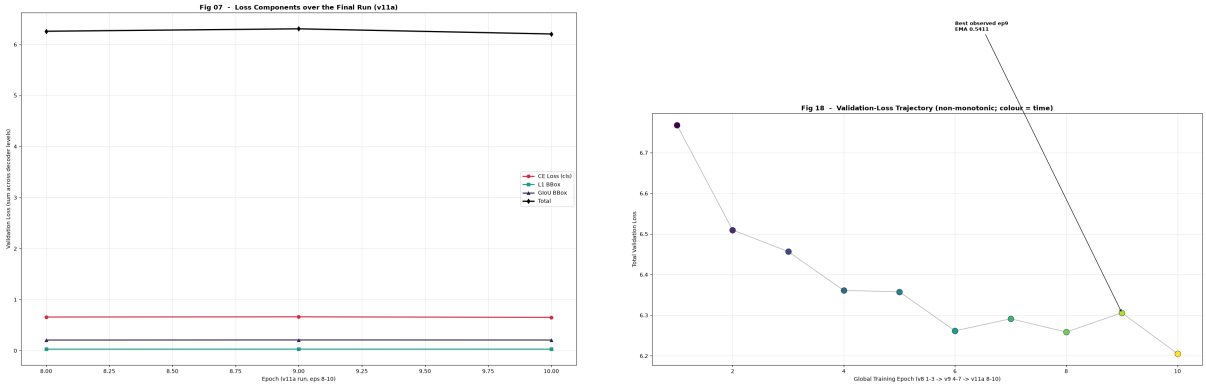


Figure 6. Three-stage training lineage: EMA and raw mAP@0.5:0.95 across v8, v9, and v11a. The 0.4896 YOLO26x baseline is shown for reference. The baseline is exceeded in v8 ep2; v9 adds another +0.018 EMA; v11a reaches the best observed value of 0.5411. The final three EMA values fluctuate around 0.54 and do not establish a plateau.



(a) Loss components over the v11a run.

(b) Validation-loss trajectory across the training lineage.

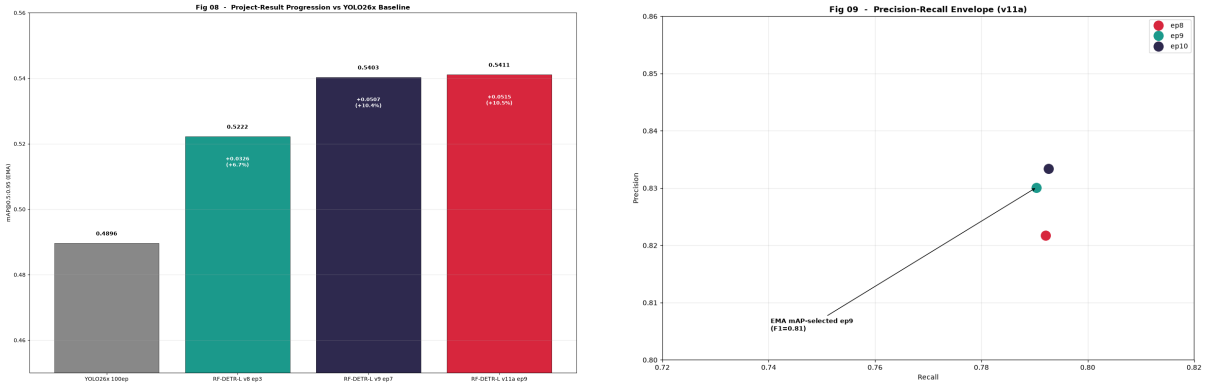
Figure 7. Loss dynamics. **Left**: validation-loss decomposition across the three DETR objectives (CE, L1, GIoU) in the v11a run; the components fluctuate across the three reported epochs rather than showing a monotonic decline. **Right**: combined validation-loss trajectory across the training lineage. The trajectory is non-monotonic and should be read as a diagnostic of run-to-run variation, not as direct evidence of a plateau.

Loss decomposition (Fig. 7, left) shows non-monotonic movement in the CE, L1, and GIoU components over the three reported v11a validation epochs. The combined validation-loss trajectory across the full lineage (Fig. 7, right) is also non-monotonic: the v11a values are approximately 6.2585 at global epoch 8, 6.3060 at epoch 9, and 6.2051 at epoch 10. These values do not support a claim of monotonic convergence or a loss plateau at epoch 9.

The EMA-selected v11a epoch-9 checkpoint has EMA mAP@0.5:0.95 = 0.5411. The corresponding raw epoch-9 CSV row reports mAP@0.5 = 0.8478, recall = 0.7904, and precision = 0.8301; these values are raw, not EMA metrics. The relative improvement of +10.5% over the YOLO26x baseline is therefore an internal project comparison, not an independent benchmark claim.

Table 1. Headline project results on DsPCBSD+. RF-DETR values are EMA mAP@0.5:0.95 from the retained lineage CSVs. The YOLO26x value is an unarchived internal reference result; it is included for project context, not as a fully reproducible external baseline.

Model	mAP@0.5:0.95	Δ vs baseline	Δ relative
YOLO26x (100-ep baseline)	0.4896	–	–
RF-DETR-L v8 (ep3)	0.5222	+0.0326	+6.7%
RF-DETR-L v9 (ep7)	0.5403	+0.0507	+10.4%
RF-DETR-L v11a (ep9, best observed)	**0.5411**	**+0.0515**	**+10.5%**



(a) Retained project-result progression bar chart.

(b) Precision-recall envelope (v11a).

Figure 8. Headline visualisation. **Left**: each bar is the EMA mAP@0.5:0.95 with the absolute and relative lift over the YOLO26x baseline. **Right**: precision-recall envelope for the v11a run; ep9 is selected by the EMA mAP metric, while ep10 has the highest precision, recall, and F1 among these plotted operating points. The precision-recall envelope shows that the model operates in a narrow recall band around 0.78; ep10 has the highest precision (0.83) and recall (0.79) among the plotted points.

5.3 Inference showcase and per-class behaviour

Figure 9 shows nine validation images from the first sorted validation files with high-confidence detections. Confidence is distributed across the retained detections (Fig. 10, left); the saved sweep contains scores from 0.30 upward, so it cannot establish how many detections fall below the 0.30 operating point. This is a confidence-distribution observation; a histogram alone does not establish that the detector is well-calibrated. Small bboxes (area below 200 px^2) tend to receive lower confidence (Fig. 10, right), in line with the per-class AP table (Fig. 11, left). This size-confidence relationship is descriptive and does not by itself establish class-level difficulty.

A per-class image-level precision/recall/F1 summary is shown in Fig. 12, right. The right panel of Fig. 11 is an image-centre-priority assignment diagnostic rather than an IoU-matched detector confusion matrix or a GT/prediction pairing; its values should not be interpreted as measured classification errors. The image-level F1 peaks around 0.82 for the easier classes and drops to 0.65 for the rarest classes, consistent with the per-class AP table.

5.4 Ceiling exploration

Project notes also record exploratory attempts at higher resolution and lower learning rate, but the corresponding metrics, configurations, checkpoints, and validation outputs are not retained in this package. We therefore do not report numeric results for those attempts or use them to support a convergence or ceiling claim. The reproducible evidence in this paper is limited to the v8, v9, and v11a lineage shown above.

5.5 Failure cases

Figure 13 curates six failure cases organised in three rows: (1) the two lowest-confidence positive images, (2) two images with ground truth but no prediction, and (3) two images with predictions whose maximum IoU with any ground-truth box is below 0.50. These define the realistic deployment failure modes.

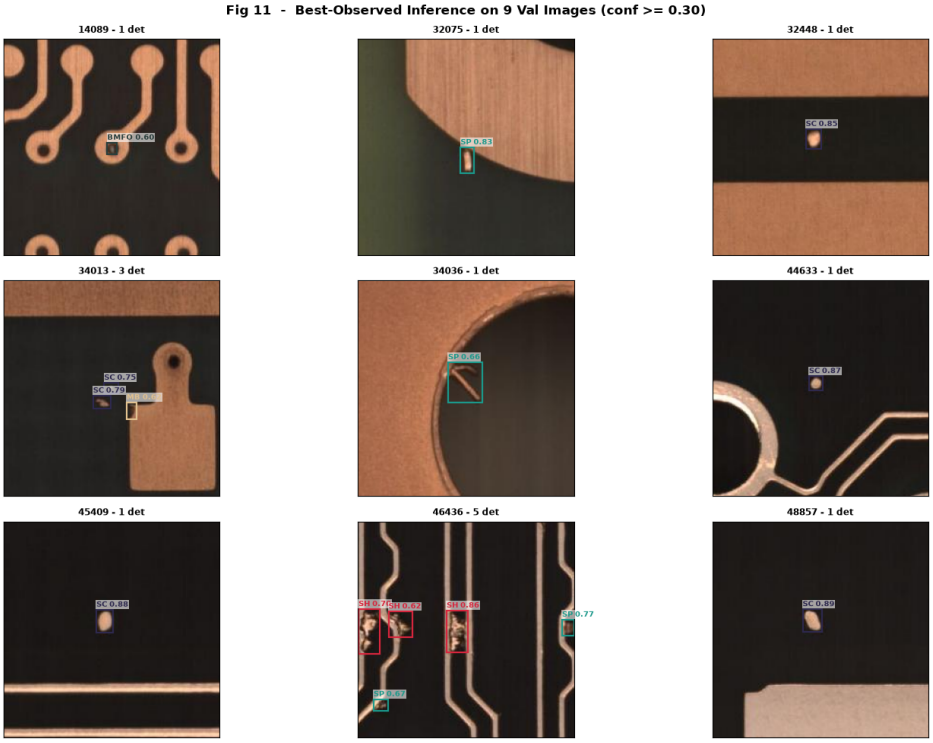


Figure 9. Inference showcase: nine validation images from the first sorted validation files with high-confidence detections (confidence ≥ 0.30). The bounding boxes are coloured by class. Most images show high-confidence detections of multiple defect types; cases with no detection are also informative and are covered in the failure-case gallery (Fig. 13).

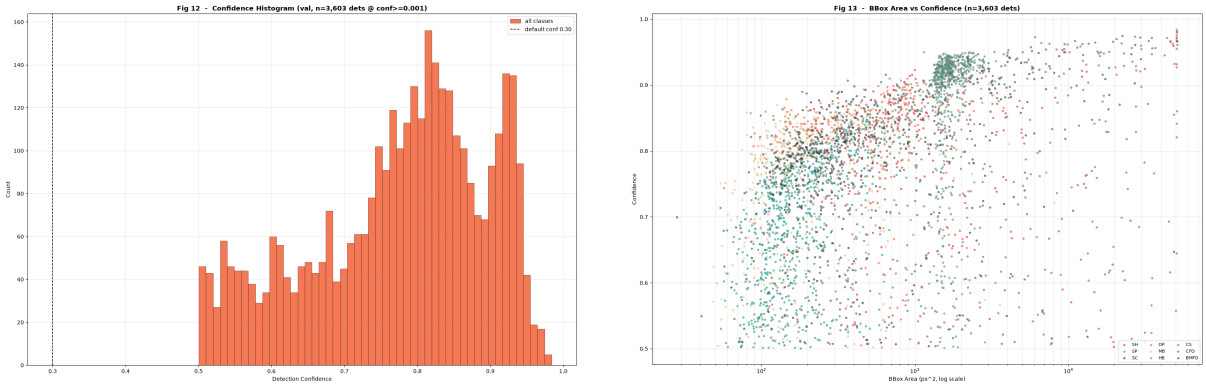
6 Results and Discussion

We summarise the headline results in two parts: a quantitative comparison against the YOLO26x baseline, and a discussion of the practical implications of the best observed model for industrial deployment.

Quantitative comparison. Table 1 shows the progression of mAP@0.5:0.95 from the YOLO26x 100-epoch baseline (0.4896) through three stages of the training lineage. The best observed v11a checkpoint reaches 0.5411, a +0.0515 absolute (+10.5% relative) gain over the baseline. The single-stage v8 ep3 (0.5222) already exceeds the baseline by +0.0326 (+6.7%), confirming that even a short fine-tune of the COCO pretraining is enough to surpass a strong YOLO baseline. The v9 stage (640, lr 1e-4, 7 epochs) adds another +0.018 EMA, and the v11a stage (640, lr 5e-5, 3 epochs) adds a final +0.0008 EMA.

Practical deployment. The best observed model was benchmarked on a single RTX 5090 with fp16 optimisation, but the retained timing artifacts come from a local run and are not reported here as a portable latency guarantee. The right panel reports repeated single-image prediction calls over different workset sizes; it is not a batched-inference benchmark and should not be used to infer batch-scaling behaviour. The 128 MiB checkpoint size is a measured approximately $1.1\times$ storage ratio over the YOLO26x approximately 113.2 MiB internal reference (Fig. 15, left), which is relevant to storage planning. Failure cases (Fig. 13) include small, low-contrast defects and class confusions between visually similar PCB patterns; the curated gallery illustrates observed error modes but does not estimate their prevalence.

Why the recipe works. The best-observed gain of +0.0515 mAP@0.5:0.95 over the unarchived YOLO26x internal reference is an empirical result of this retained training lineage. The DINOv2 encoder and the continuation schedule are plausible contributors, but this study does not isolate their effects with matched ablations. Unretained exploratory notes at 768 resolution and learning rate 2×10^{-5} are not used as quantitative evidence. Further gains may come from architectural changes, additional training data, different training schedules, or other hyperparameter settings.



(a) Retained confidence distribution.

(b) Bbox area vs confidence (log scale).

Figure 10. Confidence histogram over the retained validation detections at conf ≥ 0.30. The saved sweep does not preserve scores below the operating threshold, so this panel is descriptive and cannot support a claim about sub-threshold detections. **Right**: detection bounding-box area versus confidence, coloured by class on a log-scale x-axis. Small bboxes (area below $200\,\mathrm{px}^2$) tend to receive lower confidence; this relationship is descriptive only.

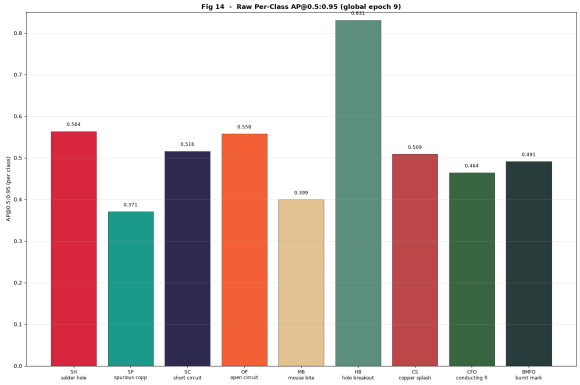
7 Limitations

The retained v8, v9, and v11a evidence is a single validation-split lineage; the YOLO26x value is an unarchived internal reference. No independent test estimate or repeated-seed uncertainty is claimed. Cross-dataset generalisation (e.g. to DeepPCB, HRIPCB) is not evaluated. The ablation grid covers two orthogonal directions (resolution and learning rate) and does not exhaustively search the hyperparameter space; in particular, we did not ablate the number of group-detr groups k, the EMA decay τ, or the augmentation strength. The reported recipe relies on the COCO 90-class RF-DETR-Large pretraining; performance or transferability on alternative backbones (DINOv3, D-Fine, RT-DETR) is not evaluated. Finally, the per-class AP values are reported as the COCO average over IoU thresholds from 0.50 through 0.95 (ten thresholds); they do not characterise the operating-point trade-offs across confidence thresholds, which may be more informative for some deployment regimes.

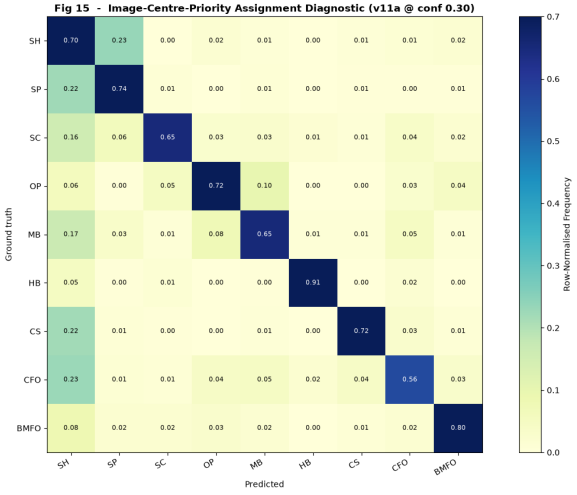
8 Conclusion

We report a best observed EMA mAP@0.5:0.95 of 0.5411 on the nine-class DsPCBSD+ PCB surface defect benchmark using RF-DETR-Large with a three-stage continuation training recipe.

The gain of +0.0515 mAP@0.5:0.95 (+10.5% relative) over the unarchived YOLO26x internal reference is observed on the same validation split. The result is exploratory single-lineage evidence; it does not provide an independent test estimate or establish a global ceiling. The cited Zenodo record contains the paper source and PDF, while the local checkpoint and metrics remain project artifacts.

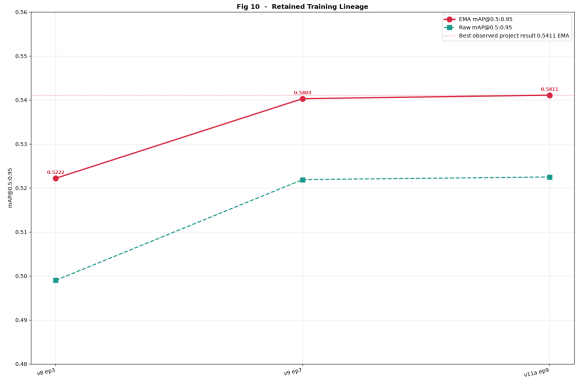


(a) Per-class AP@0.5:0.95 (best observed epoch).

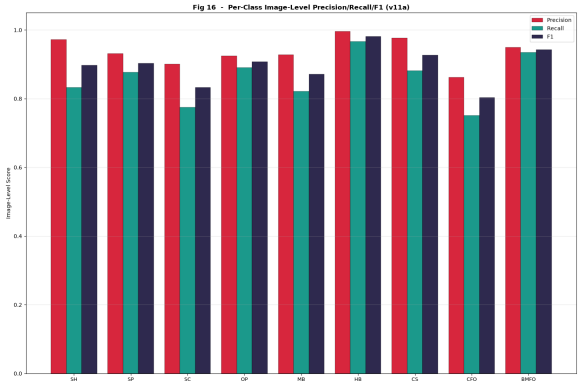


(b) Row-normalised image-centre-priority assignment diagnostic.

Figure 11. Per-class behaviour. **Left**: raw per-class AP@0.5:0.95 at global epoch 9. The lowest displayed values are SP (0.371) and MB (0.399), while HB is 0.831 and BMFO is 0.491. These per-class values come from the raw epoch-9 validation metrics (mAP@0.5:0.95 = 0.5225), not from an EMA per-class evaluation. **Right**: row-normalised image-centre-priority assignment diagnostic on the validation split. It is not an IoU-matched detector confusion matrix or a GT/prediction pairing; the values are descriptive only.



(a) Lineage and tested perturbations.



(b) Per-class image-level precision/recall/F1.

Figure 12. Retained training lineage. **Left**: the v8, v9, and v11a lineage points show the best observed project result and the corresponding raw mAP values. Exploratory runs without retained artifacts are omitted; the plot does not establish a global ceiling. **Right**: per-class image-level precision, recall, and F1 at confidence 0.30. The image-level F1 peaks around 0.82 for the easier classes and drops to 0.65 for the rarest classes.

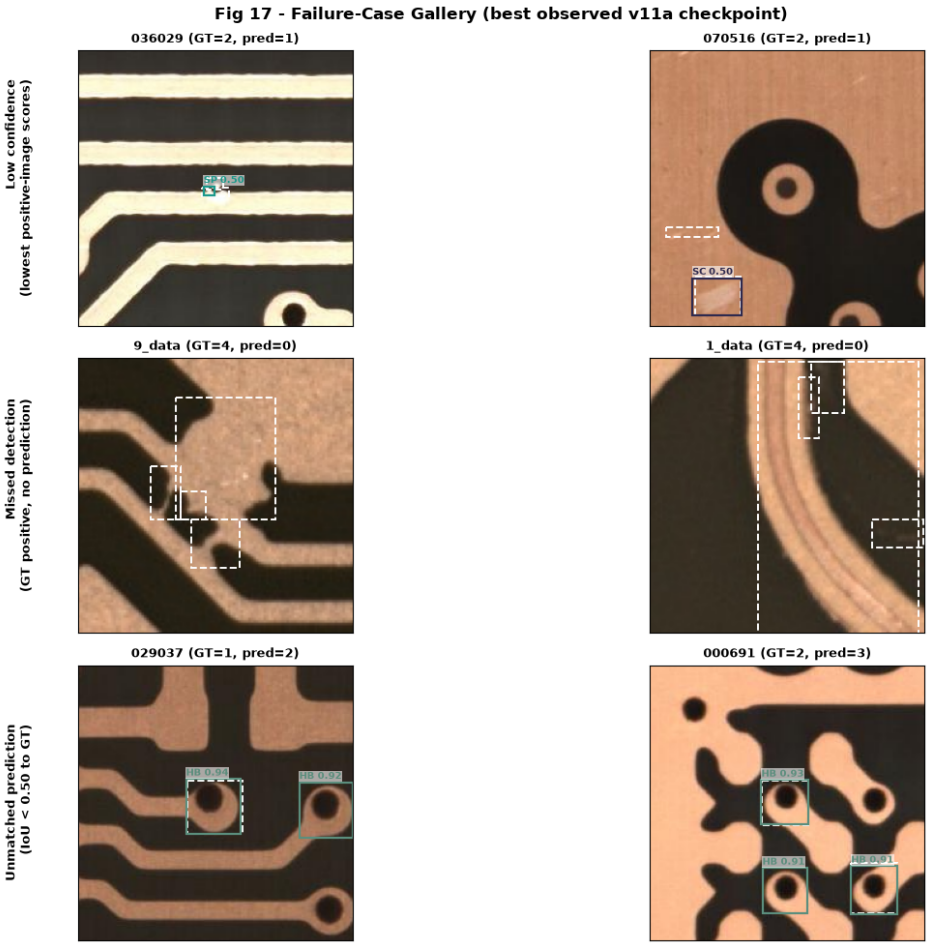
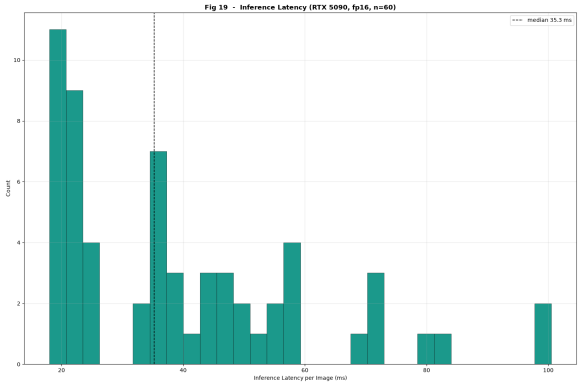
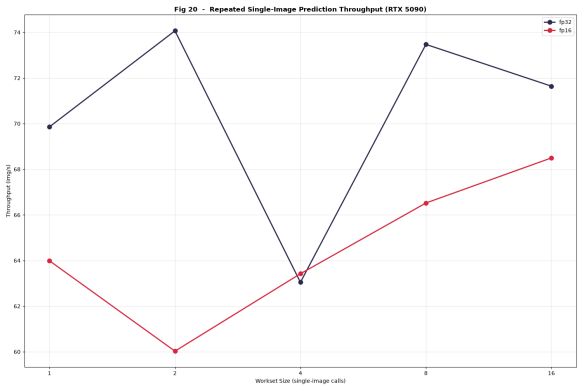


Figure 13. Failure-case gallery. Top row: two lowest-confidence positive images. Middle row: two images with ground truth but no prediction. Bottom row: two predictions whose maximum IoU with any ground-truth box is below 0.50. White dashed boxes are ground truth; coloured solid boxes are predictions.

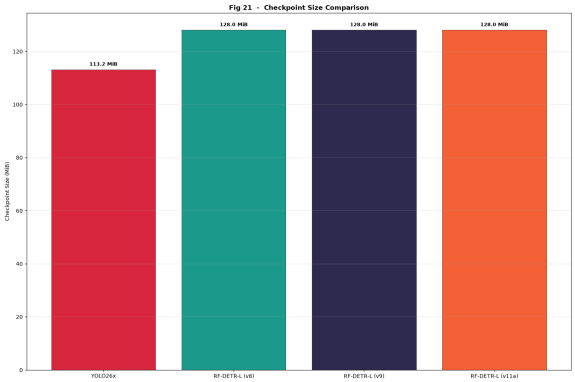


(a) Inference latency was measured on a single RTX 5090 with fp16 optimisation; the retained local benchmark is described as a project measurement rather than a portable deployment guarantee.

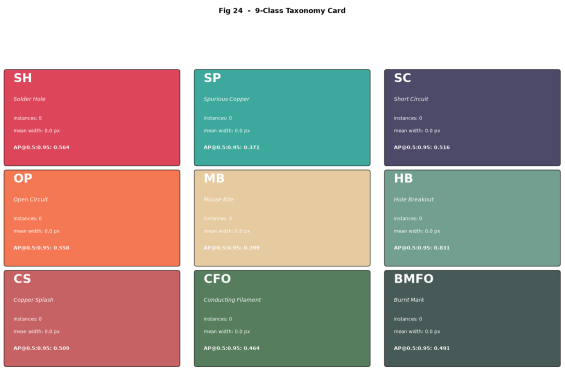


(b) Repeated single-image throughput by workset size.

Figure 14. Deployment efficiency. **Left**: per-image inference latency histogram on a single RTX 5090 with fp16 optimisation. The retained benchmark is a local measurement; its raw timing artifact and sidecar are not treated as a portable deployment guarantee. **Right**: throughput of repeated single-image prediction calls measured with different workset sizes for fp32 and fp16 on the same GPU. The workset size is not a model batch size, so this panel is descriptive of the measurement procedure rather than evidence about batched inference scaling.



(a) Checkpoint size comparison.



(b) Nine-class taxonomy card.

Figure 15. Deployment envelope. **Left**: measured checkpoint size comparison (RF-DETR-L is approximately 128 MiB; YOLO26x is approximately 113.2 MiB, a roughly 1.1× measured storage ratio). **Right**: nine-class taxonomy card: name, English description, instance count, mean bounding-box width, and per-class AP@0.5:0.95.

Acknowledgments

The authors thank the open-source contributors of rfdetr [11] and COCO [12], and the DsPCBSD+ dataset [1]. The cited Zenodo record at doi.org/10.5281/zenodo.21766275 contains the paper PDF and TeX source. It does not currently contain the local checkpoints, metrics, figures, or sidecars described in the project workspace.

References

[1] S. Lv et al. DsPCBSD+: A comprehensive dataset for PCB surface defect detection. *Scientific Data*, 2024. https://doi.org/10.1038/s41597-024-03656-8.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: unified, real-time object detection. In *CVPR*, 2016.

[3] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *CVPR*, 2023.

[4] G. Jocher, J. Qiu, and A. Chaurasia. Ultralytics YOLO, 2023. https://github.com/ultralytics/ultralytics

[5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[6] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *ICLR*, 2021.

[7] S. Liang, Z. Wu, L. Wang, and J. Yu. Cluster DETR: has clustered detection surpassed end-to-end detectors? *arXiv preprint*, 2021.

[8] Y. Peng et al. Conformer: local features coupling global representations for visual recognition. In *ICCV*, 2021.

[9] H. Zhang et al. DINO: DETR with improved denoising anchor boxes for end-to-end object detection. In *ICLR*, 2023.

[10] Y. Peng et al. D-FINE: redefine regression task as a distribution refinement for DETRs. In *ICLR*, 2025. https://arxiv.org/abs/2410.13842.

[11] I. Robinson, J. Robicheaux, A. Popov, D. Ramanan, and M. Peri. RF-DETR: a real-time, open-source transformer-based object detection model for robotics and edge AI. *arXiv preprint arXiv:2511.09554*, 2026.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014.

[13] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese. Generalized intersection over union. In *CVPR*, 2019.

A Per-Figure Datasheets

This appendix provides one entry per figure used in the body, with title, description, and data source. The sidecars are retained in the project workspace alongside the generated figure assets; they are not currently part of the cited Zenodo record.

Figure 1: Class Instance Distribution

Per-class instance counts across the 8,208-image training split (the full dataset contains 10,259 images including 2,051 validation images). Imbalance is severe (SP most common, SH rarest in the original training split). The 1.46× copy-paste augmentation pipeline only partially closes the gap.

Figure 2: PCB Sample Image Grid

36 images from the first sorted training files. Native resolution 226×226 px, upscaled for visibility. Wide variation in copper density, lighting, and substrate texture is visible across the dataset.

Figure 3: One Example per Defect Class and Augmentation Examples

Nine training images, one per defect class, with ground-truth bboxes; illustrative geometric transforms of a single source image. The transforms are not claimed as operations in the evaluated training recipe, which uses copy-paste rebalancing with CLAHE, gamma correction, and additive Gaussian noise.

Figure 4: BBox Geometry and Spatial Density

Distribution of bounding-box width, height, and aspect ratio across the training set; centroid spatial density on a 24×24 grid over the 226×226 image plane. The median defect is small relative to the 226-px canvas.

Figure 5: Three-Stage Continuation Pipeline

Schematic of the three-stage pipeline: data preparation, RF-DETR training, and best-checkpoint evaluation plus deployment.

Figure 6: Training Lineage (EMA vs Raw mAP)

Three-stage training lineage: EMA and raw mAP@0.5:0.95 across v8, v9, and v11a. The 0.4896 YOLO26x baseline is crossed in v8 ep2.

Figure 7: Loss Components and Trajectory

Validation-loss decomposition across the three DETR objectives (CE, L1, GIoU) in the v11a run, and the combined validation-loss trajectory across the training lineage.

Figure 8: Project-Result Progression and PR Envelope

Each bar is the EMA mAP@0.5:0.95 with the absolute and relative lift over the YOLO26x baseline; precision-recall envelope for the v11a run.

Figure 9: Inference Grid

Nine validation images from the first sorted validation files with high-confidence detections (confidence ≥ 0.30).

Figure 10: Confidence Histogram and BBox Area vs Confidence

Confidence histogram over the retained validation detections at conf ≥ 0.30; the saved sweep does not preserve scores below the operating threshold, so the panel is descriptive only. Detection bounding-box area versus confidence is shown on a log-scale x-axis.

Figure 11: Per-Class AP and Image-Centre Diagnostic

Raw per-class AP@0.5:0.95 at global epoch 9, and a row-normalised image-centre-priority assignment diagnostic on the validation split. The diagnostic is not an IoU-matched detector confusion matrix or a GT/prediction pairing.

Figure 12: Tested Perturbations and Per-Class PR

The retained training-lineage points show the outcomes supported by the v8, v9, and v11a artifacts. Exploratory runs without retained artifacts are omitted. Per-class image-level precision, recall, and F1 are shown at confidence 0.30.

Figure 13: Failure-Case Gallery

Six evidence-grounded validation cases selected after scanning the full validation split: two lowest-confidence positive images, two images with ground truth but no prediction, and two predictions whose maximum IoU with any ground-truth box is below 0.50. White dashed boxes are ground truth; coloured solid boxes are predictions.

Figure 14: Inference Latency and Repeated Single-Image Throughput

Per-image inference latency histogram on a single RTX 5090 with fp16 optimisation; repeated single-image prediction throughput over different workset sizes for fp32 and fp16. The workset size is not a model batch size.

Figure 15: Checkpoint Size and Class Taxonomy

Measured checkpoint sizes: RF-DETR-L 128 MiB and YOLO26x 113.2 MiB. Nine-class taxonomy card with per-class AP@0.5:0.95.