Detecting and Classifying Defects In Chips With Computer Vision

Team 4

Team Members

- PRAYAG NIKUL PURANI - 017416737
- SAI VIVEK CHUNDURI - 017435301
- SINDHU NAGESHA - 017419987
- SYED FARAAZ AHMED - 017428619

Overview

01 Problem Statement
02 Literature Survey
03 Technological Survey
04 Data Collection
05 Data Process
06 Data Transformation
07 Data Statistics
08 System Architecture
09 Machine Learning Results
10 Data Pipeline Demo
11 Conclusion

Problem Statement

- Detecting defects in PCB chips and Wafers in semiconductor manufacturing is critical to ensuring product quality and reliability. Current methods, often manual, are time-consuming, error-prone, and struggle with increasing complexity and volume.

- Developing an automated defect detection system for PCB chips and Wafers using advanced image processing and machine learning techniques.

Background

- .PCB and wafer manufacturing face defect challenges like scratches, edge-locs, shorts, and mouse bites.
- Existing automated systems struggle with subtle, overlapping, or small-scale defects, especially in modern high-density layouts.

Objectives and Needs

- Develop a machine learning-based system for automated defect detection and classification using PCB and wafer datasets.
- Ensure high accuracy, real-time performance, and user accessibility through cloud deployment and interface integration.

Deliverables

- Train multiple deep learning models and select the best based on accuracy, mAP, and IoU metrics.
- Deploy the selected model in a web-based interface allowing users to upload images and receive instant defect classification results.

Literature Survey

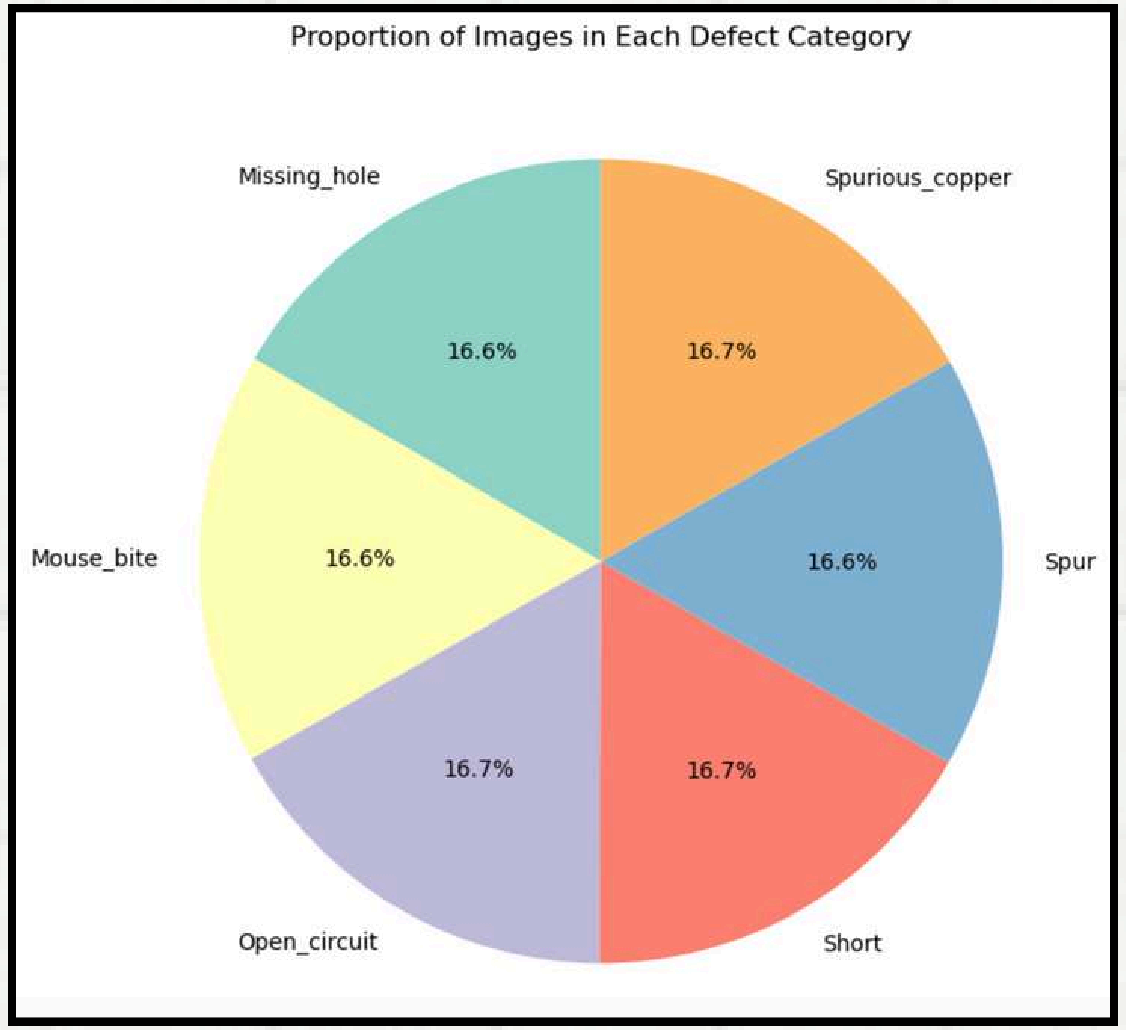
Title	Focus Area	Methodology	Key Findings	Gaps Identified/Comparison
Hidden Defect Detection in Additively Manufactured Parts	Detecting hidden defects in additively manufactured parts with complex internal features	Developed synthetic defect images from defect-free CT scans to train a V-DETR model	Achieved >90% accuracy in defect detection; capable of detecting unseen defects in the dataset	Limited to initial part designs; requires further improvements in generalizability across materials and designs
Wafer Map Pattern Recognition and Similarity Ranking	Improving efficiency and accuracy of WMFPR and WMSR systems in semiconductor manufacturing	Proposed rotation- and scale-invariant feature extraction techniques validated on the WM-811K dataset	Achieved 94.63% accuracy in WMFPR; efficient similarity ranking with a retrieval time of 2.5 seconds for 811,457 wafer maps	Further improvements needed in feature extraction and advanced machine learning for better performance and scalability
Active cluster annotation for wafer map pattern classification in semiconductor manufacturing	Reducing annotation costs in wafer map pattern classification	Grouped unlabeled wafer maps into clusters using k-means and applied CNN-based active learning with uncertainty estimation	Reduced labeling costs and improved classification accuracy with fewer iterations; enhanced consistency and mitigated noise	Trade-off between cluster purity and annotation efficiency; further improvements needed in clustering methods and mixed-pattern handling
Self-Supervised Learning for Wafer Bin Maps	Wafer bin map classification with reduced labeled data dependency	Developed a masked autoencoder (MAE) framework with patchMC encoder and multi-label fine-tuning	Achieved 96.7% classification precision; outperformed self-supervised and CNN-based methods in multi-GFA scenarios	Requires integration of object detection and few-shot learning for better handling of local features and weak signals

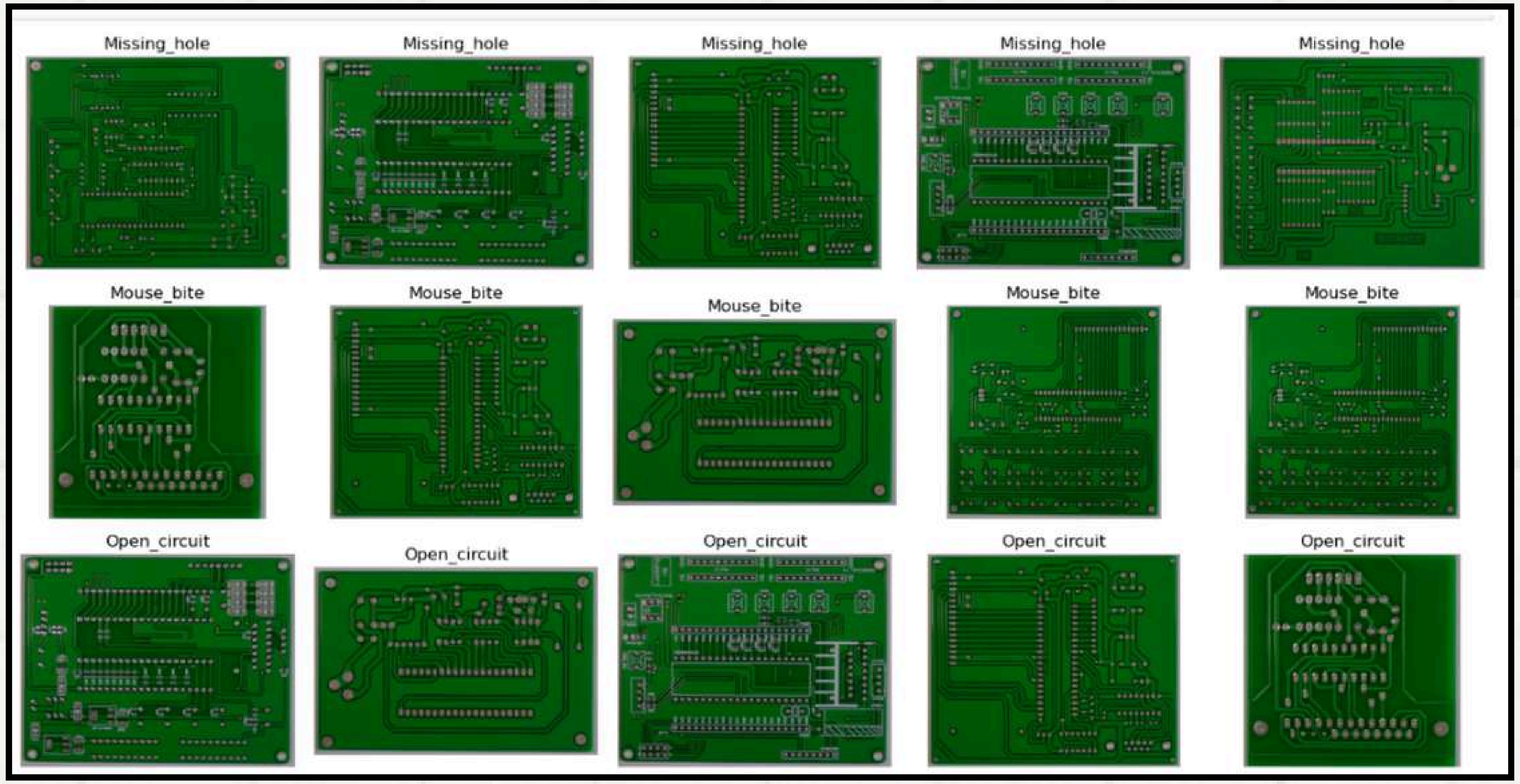
Technological Survey

Title	Technology/Technique Used	Problem Addressed	Methodology	Key results
PCB Defect Detection using YOLO	YOLO (CNN-based)	PCB defect detection in electronic devices	Trained YOLO/CNN model with 11,000 labeled images using 24 convolutional and 2 fully connected layers	Achieved 98.79% defect detection accuracy, reducing false detections and enhancing production efficiency.
Enhanced PCB Defect Detection with ACASEM	Attentive Context and Semantic Enhancement Module (ACASEM), Faster R-CNN, Cascade R-CNN	Detecting small and visually unrecognizable defects in PCBs with low accuracy or high computational cost	ACASEM fuses feature maps from various backbone layers enriched with contextual and semantic information	Improved accuracy for small and medium-sized defects, achieving better precision and recall on DeepPCB and Augmented PCB Defect datasets.
Enhancing Wafer Defect Detection Using Lightweight Machine Learning Approach	Enhanced YOLOv7, SPD-Conv, CBAM, SIoU Loss	Addressing missed detections, slow processing speeds, and small defect identification	Integrated SPD-Conv for feature extraction, CBAM for attention, and SIoU for bounding box accuracy	Achieved 92.23% accuracy, 94.1% recall, 92.5% mAP, and 136 FPS; outperformed YOLOv5 and Faster R-CNN.

Data Collection – PCB

Data	Source	Image Count	Size
Dataset 1	Open Lab on Human-Robot Interaction, Peking University	1386 Images	2 GB

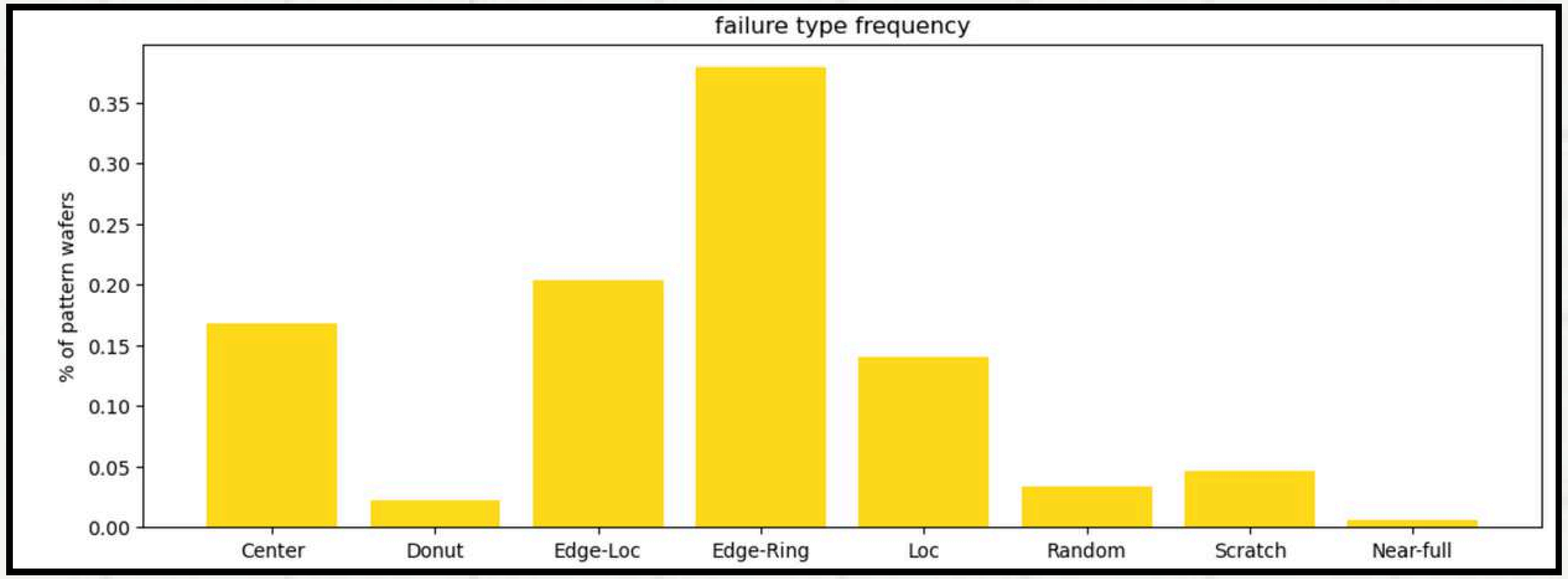


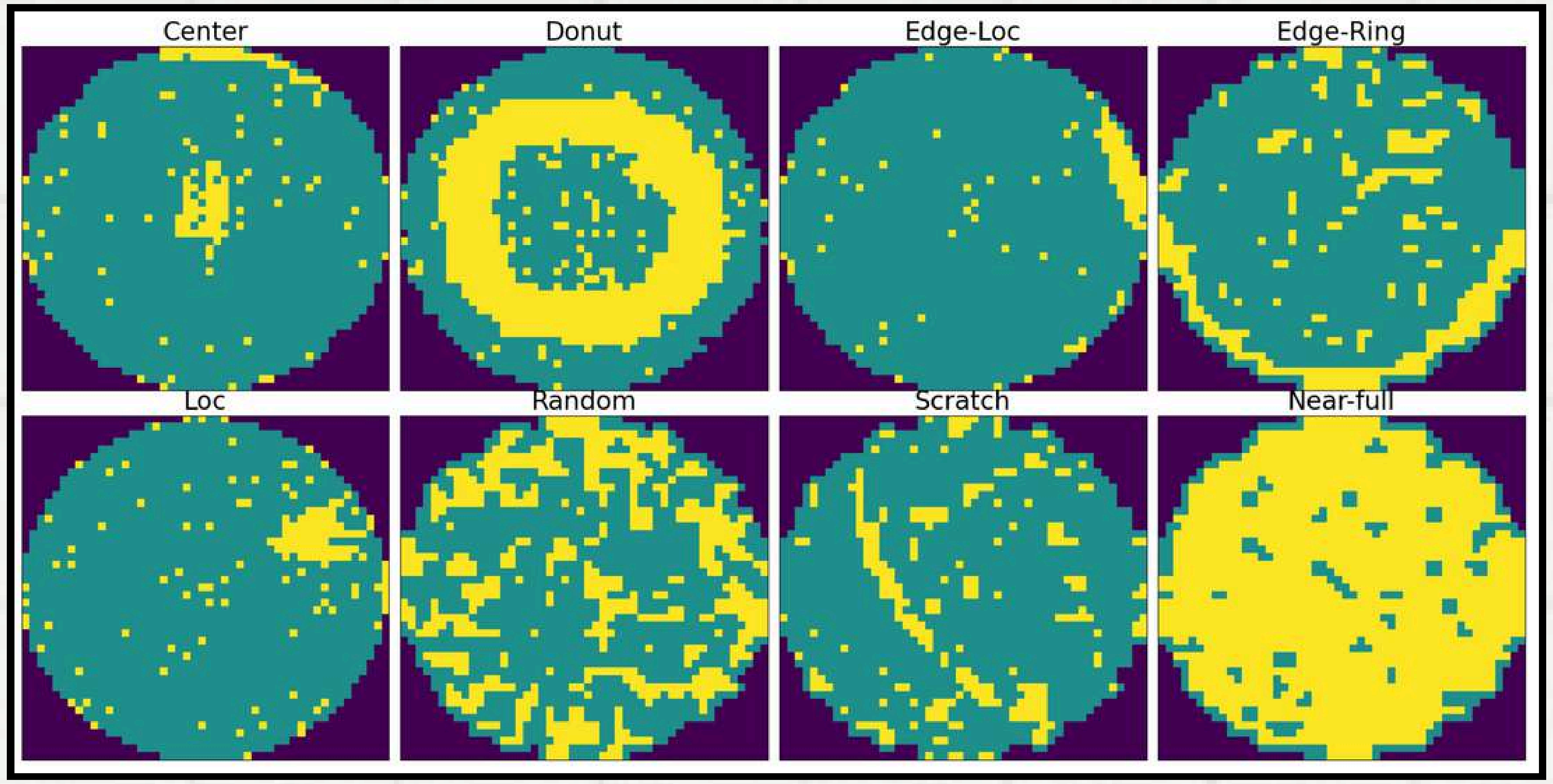


Data Collection – Wafer

Data	Source	Image Count	Size
Dataset 2	WM-811K wafer map	811457 rows	2.9GB

	waferMap	dieSize	lotName	waferIndex	trianTestLabel	failureType
0	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	1.0	[[Training]]	[[none]]
1	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	2.0	[[Training]]	[[none]]
2	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	3.0	[[Training]]	[[none]]
3	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	4.0	[[Training]]	[[none]]
4	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	5.0	[[Training]]	[[none]]





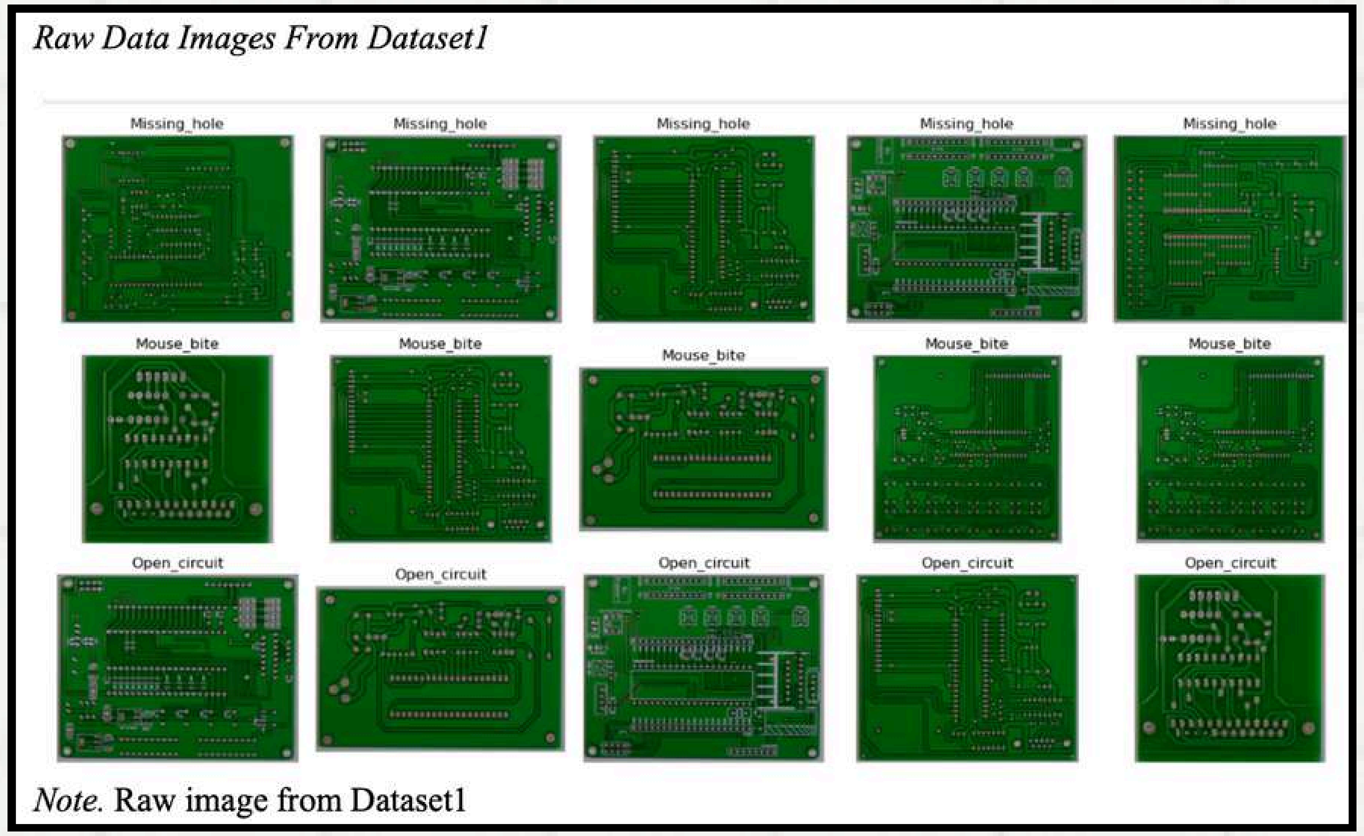
Data Process

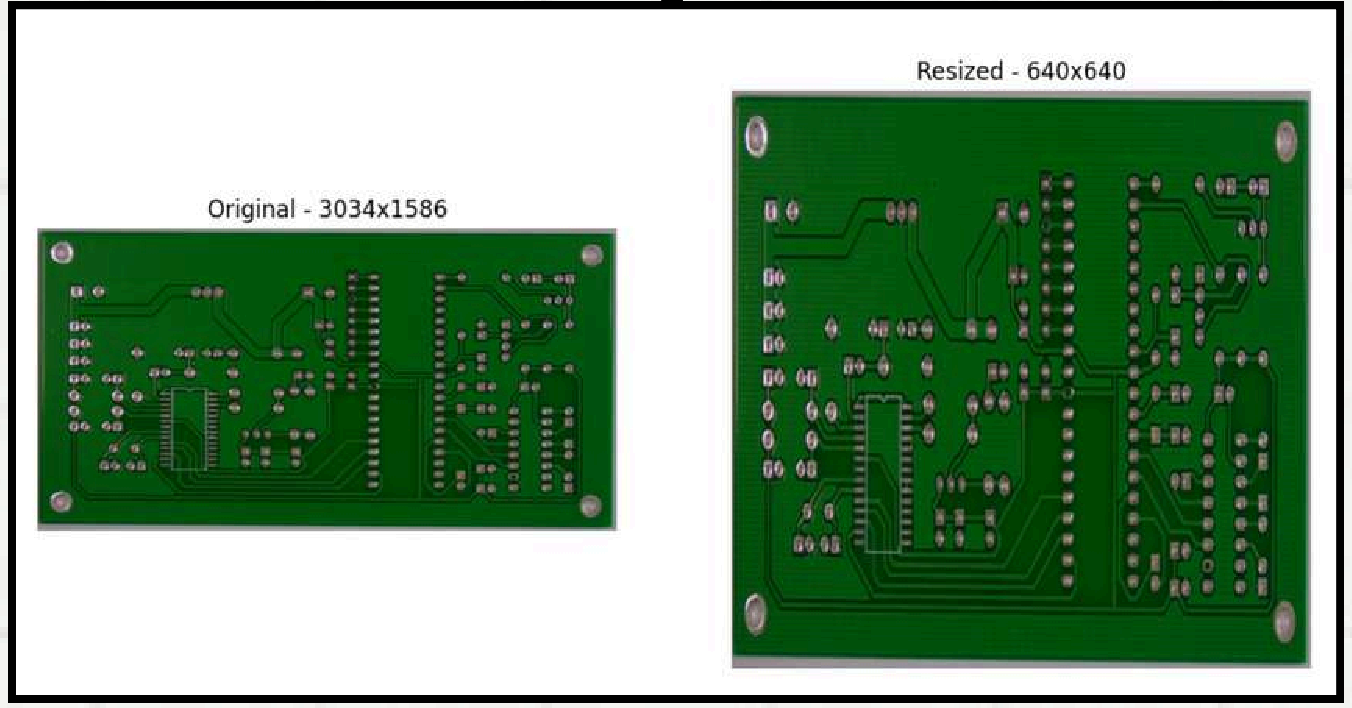
Data Collection	Data Pre-processing	Data transformation	Data Spliting
PCB data from Peking University	EDA, Cleaning, Labeling & Encoding Compression & Resize Normalization	Rotation Cutout Crop Enlarge	Stratified K-Fold Cross-Validation
Wafer Data	Maping Non-label removal Normalization of shape	Data augmentation Oversampling	K-Fold Cross-Validation

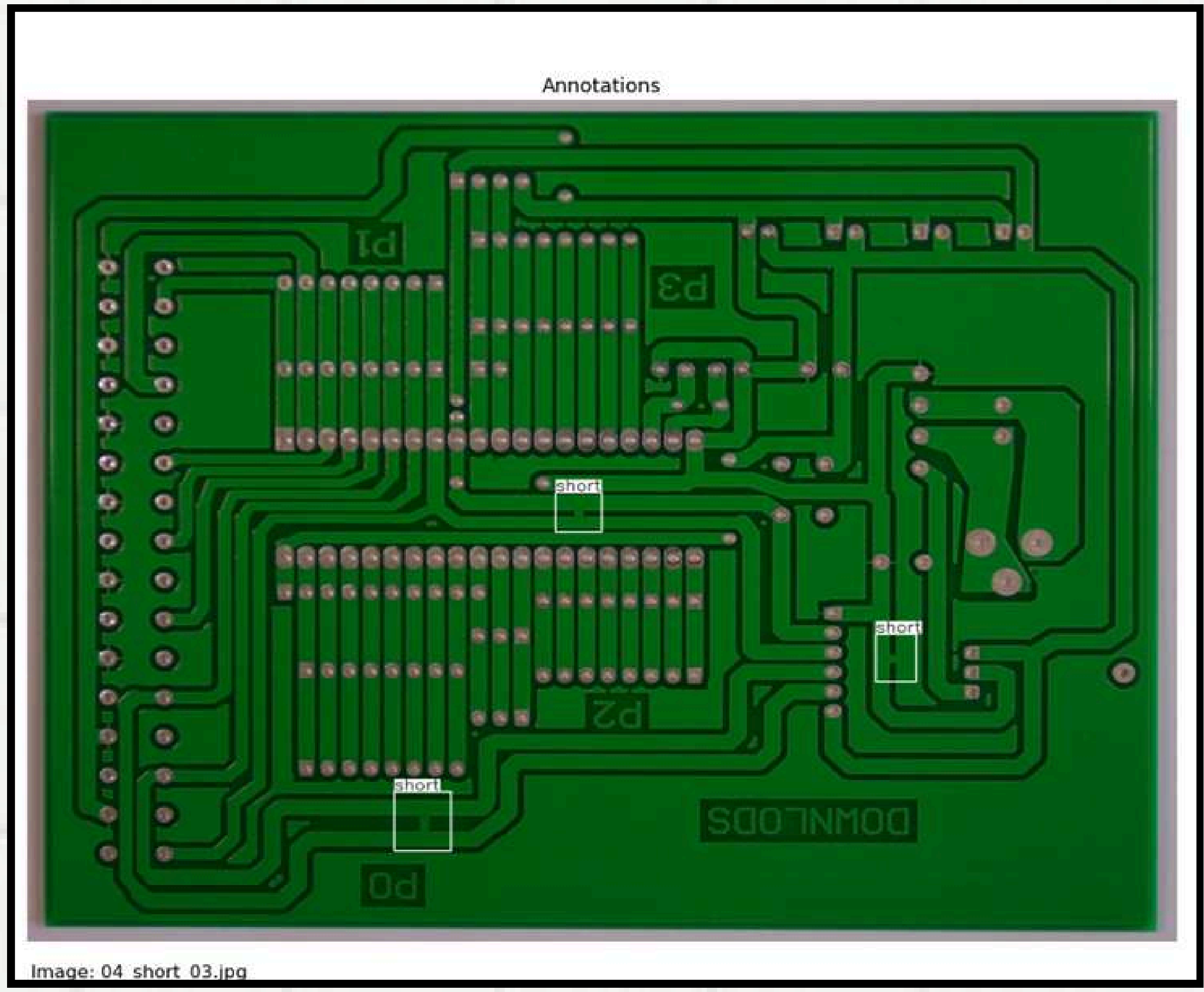
Pre – Processing Results

Dataset 1

1. Image Resizing
2. Labeling & Encoding
3. Compression
4. Normalization
5. Data Augmentation





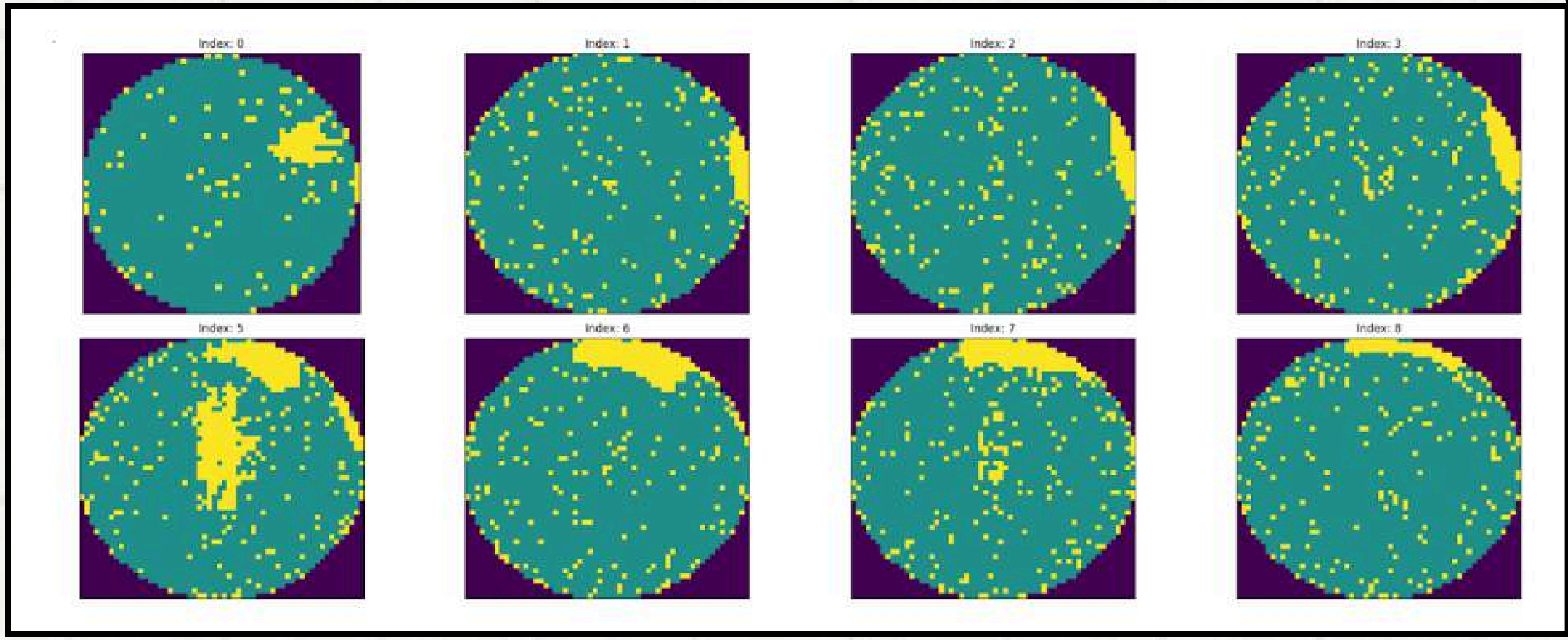


Pre – Processing Results

Dataset 2

1. Maping
2. Non-label removal
3. Normalization of shape
4. Data Augmentation
5. Oversampling

	waferMap	dieSize	lotName	trianTestLabel	failureType	waferMapDim	failureNum	trainTestNum
0	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	[[Training]]	[[none]]	(45, 48)	8	0
1	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	[[Training]]	[[none]]	(45, 48)	8	0
2	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	[[Training]]	[[none]]	(45, 48)	8	0
3	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	[[Training]]	[[none]]	(45, 48)	8	0
4	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...	1683.0	lot1	[[Training]]	[[none]]	(45, 48)	8	0

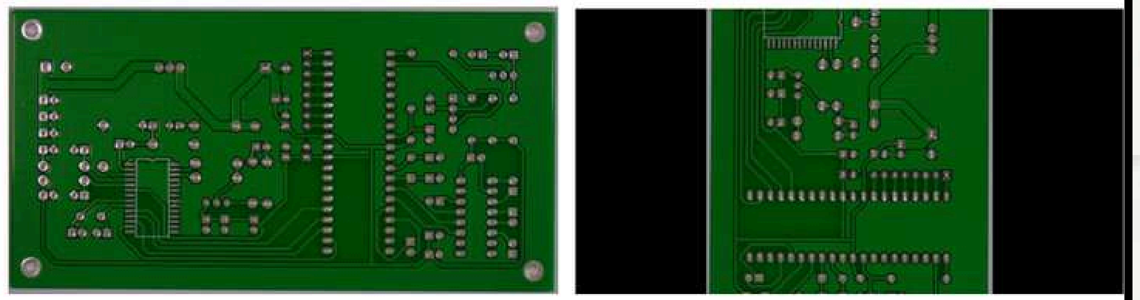


Data Transformation & Train – Test Split

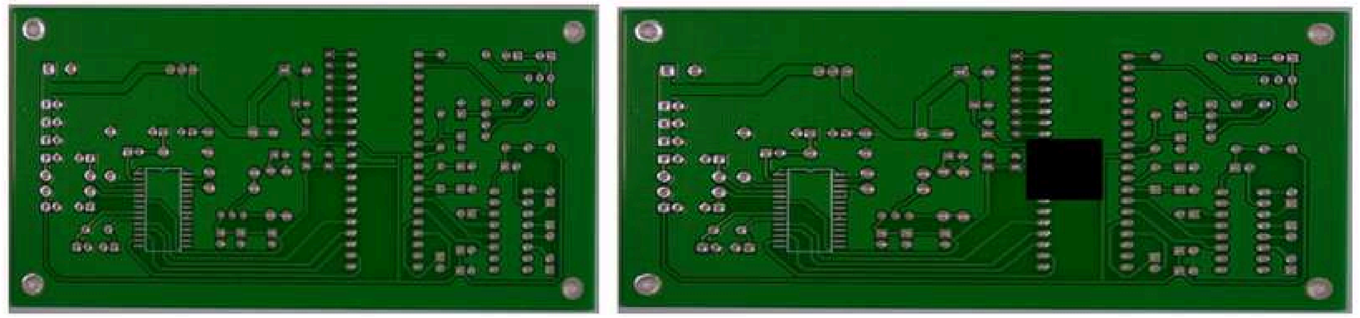
Dataset 1

1. Image Formation
2. Image Resizing
3. Crop
4. Oversampling
5. Stratified 5-fold

Data Transformation Rotation.

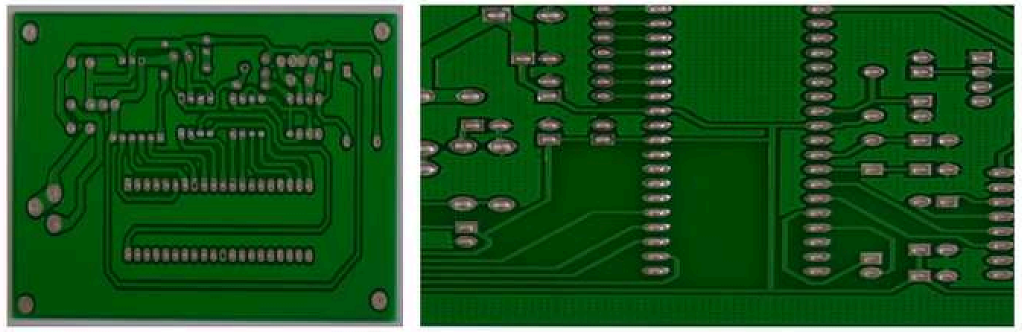


Note. Data transformation Rotation.



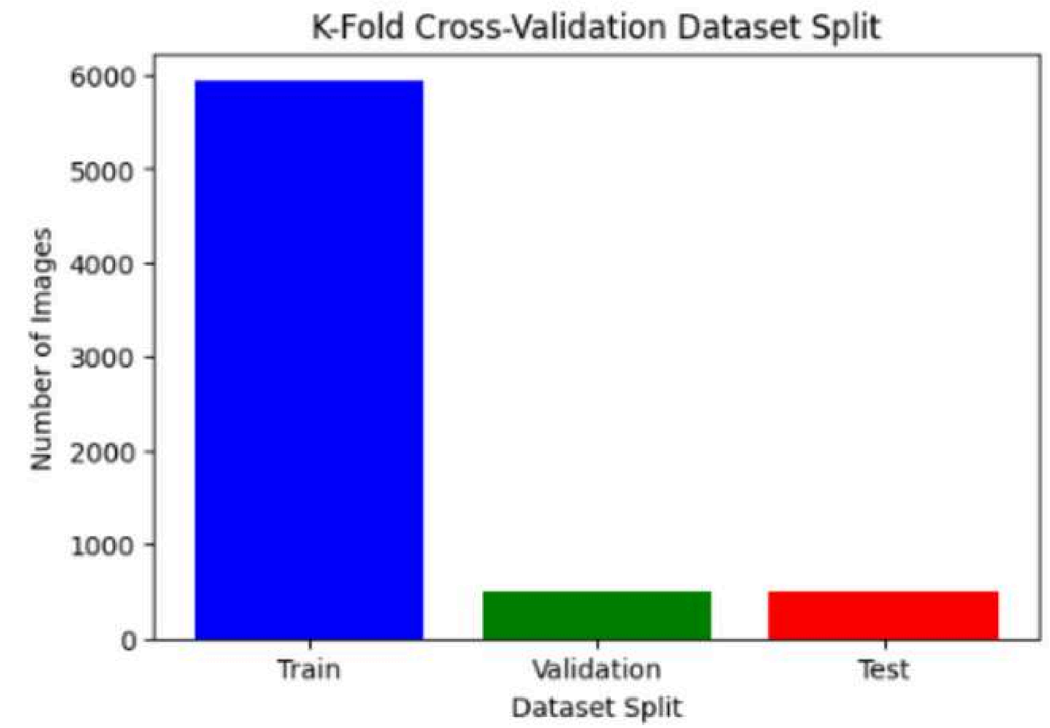
Note. Data transformation Cutout

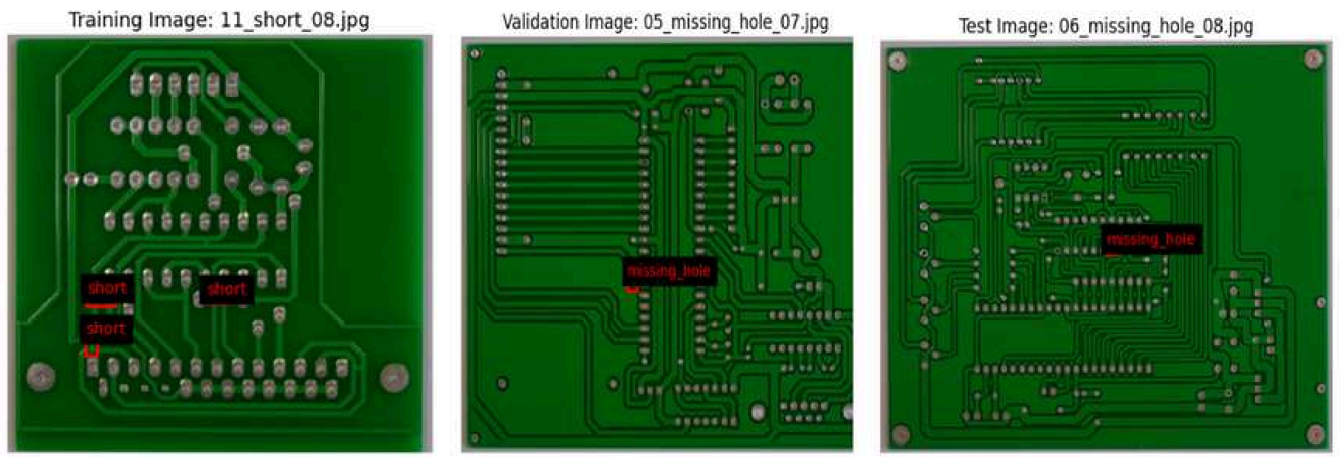
Data Transformation Cropped.



Note. Transformation Cropped.

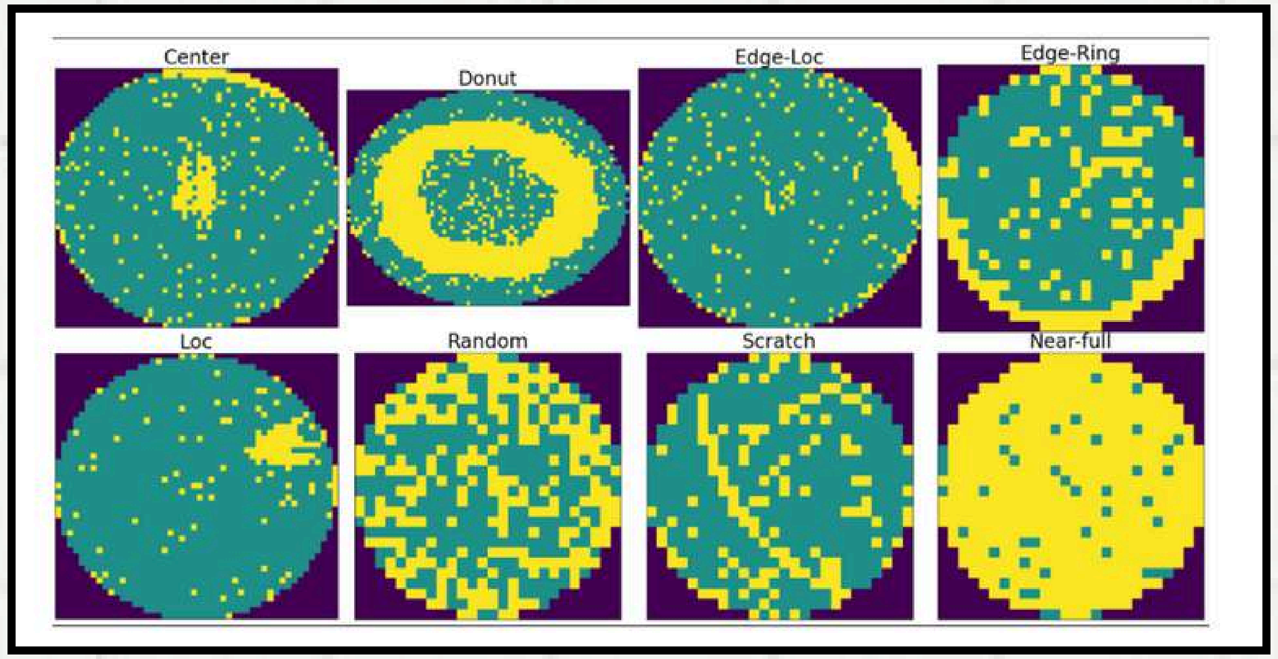
Training, Validation and Test Dataset split

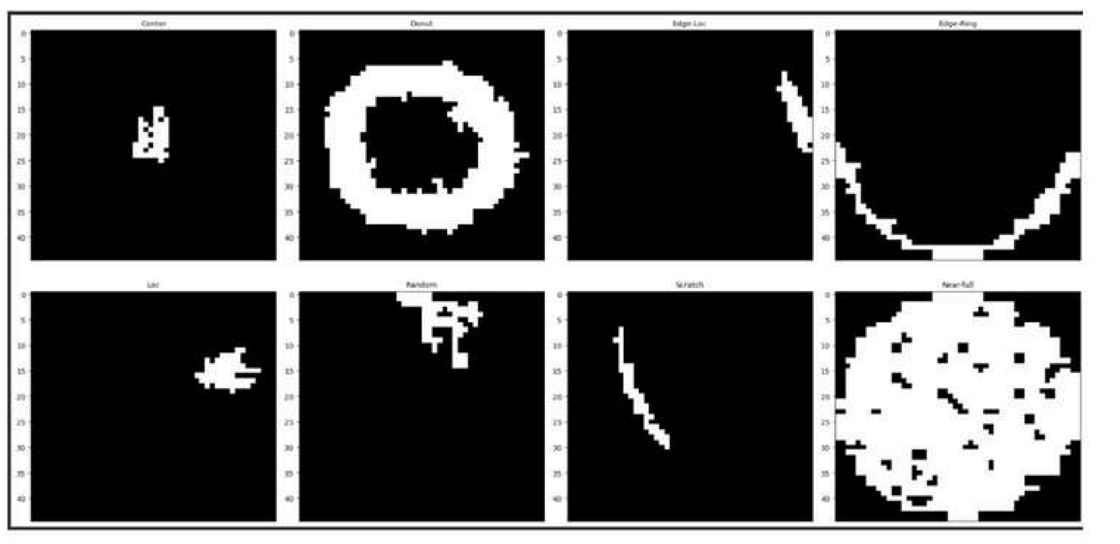




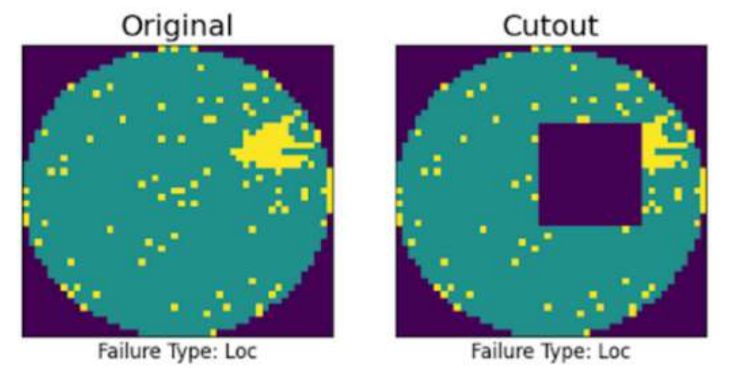
Data Transformation & Train – Test Split

Dataset 2

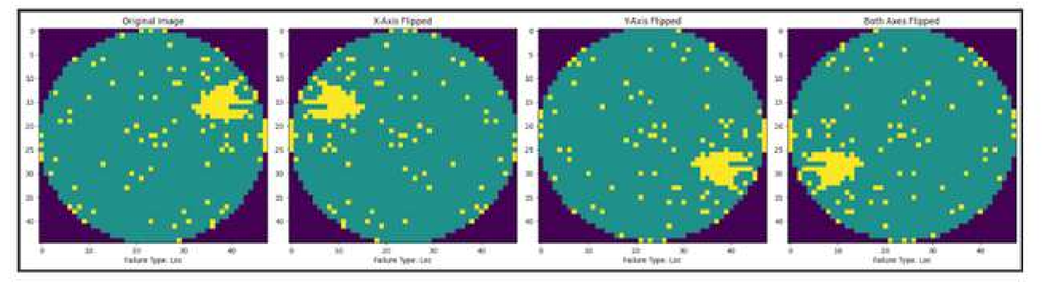




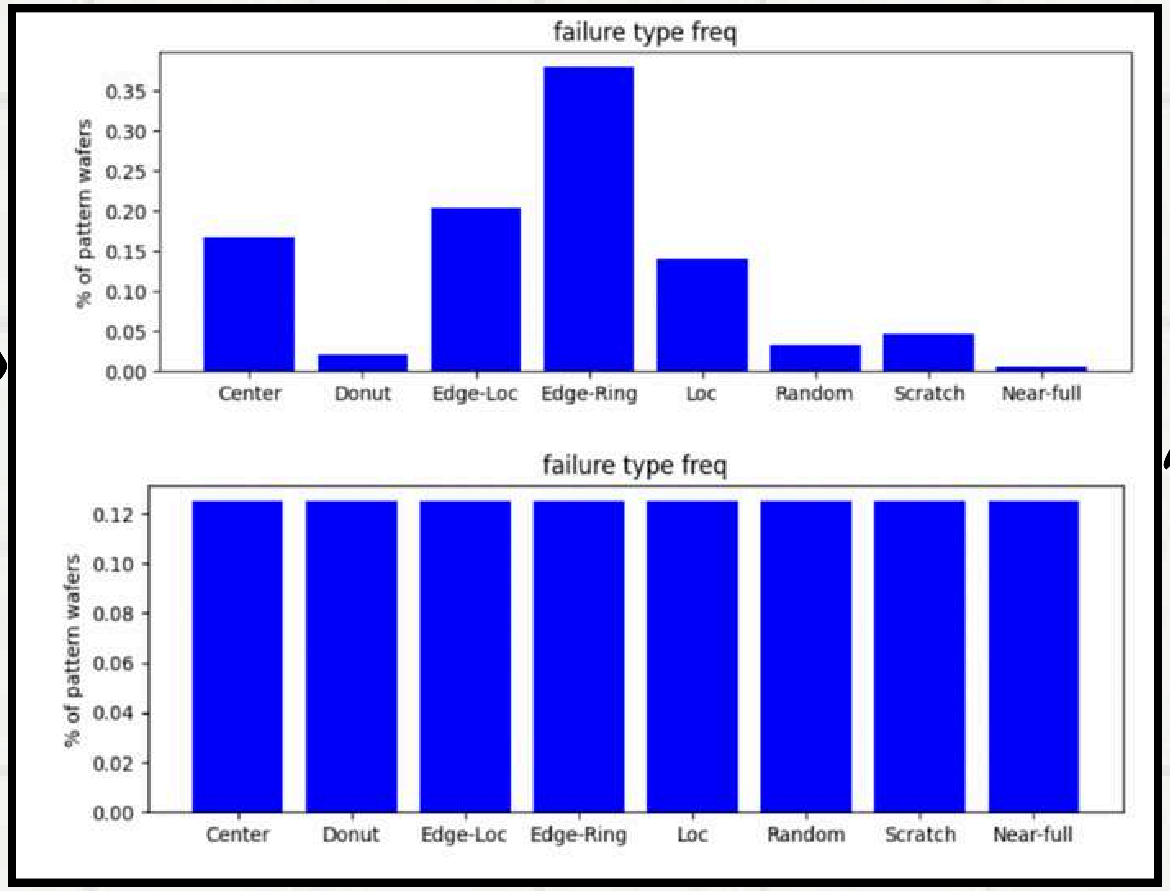
Data after feature extraction.

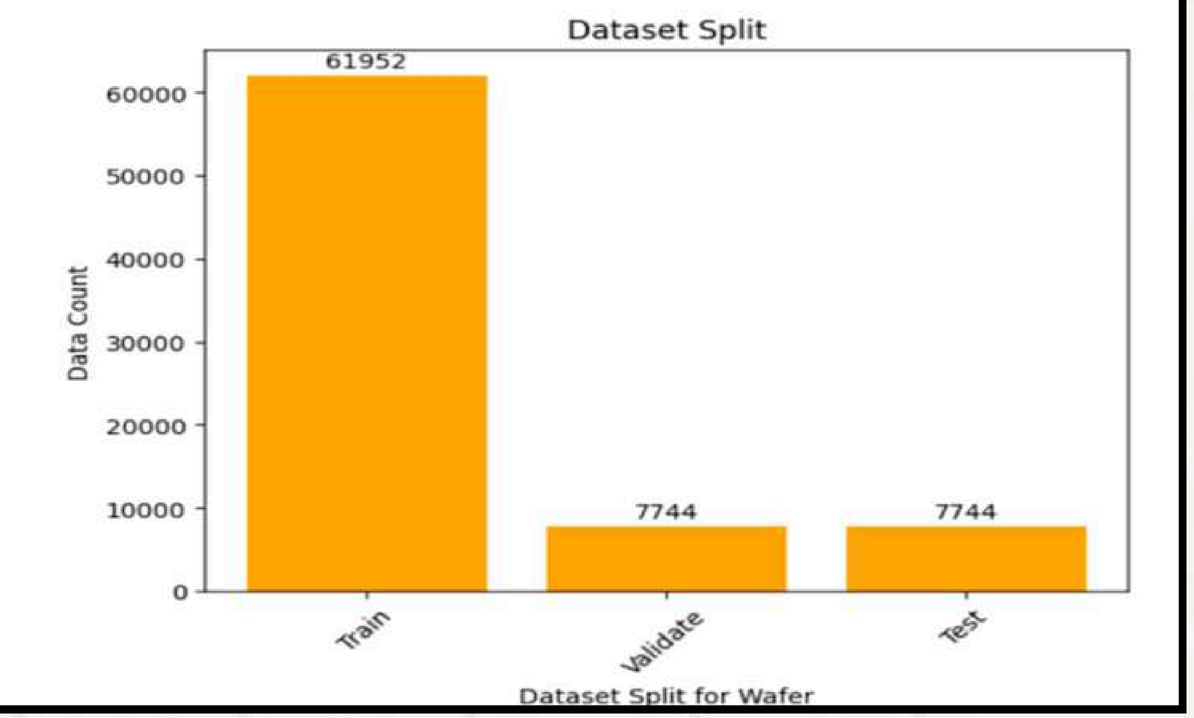


Data Transformation Cutout.



Data Transformation Rotation.



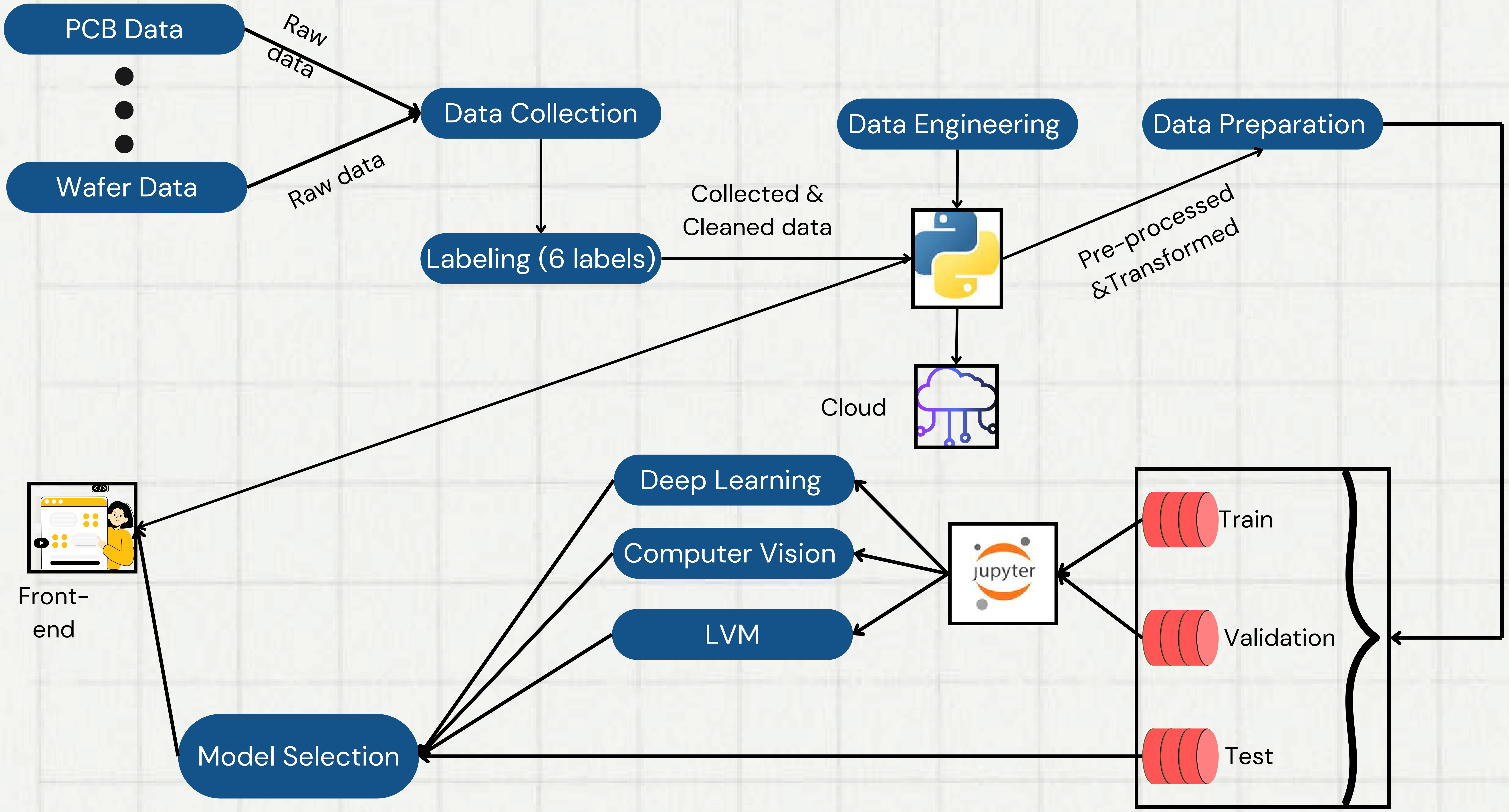


Training, Validation and Test Dataset split

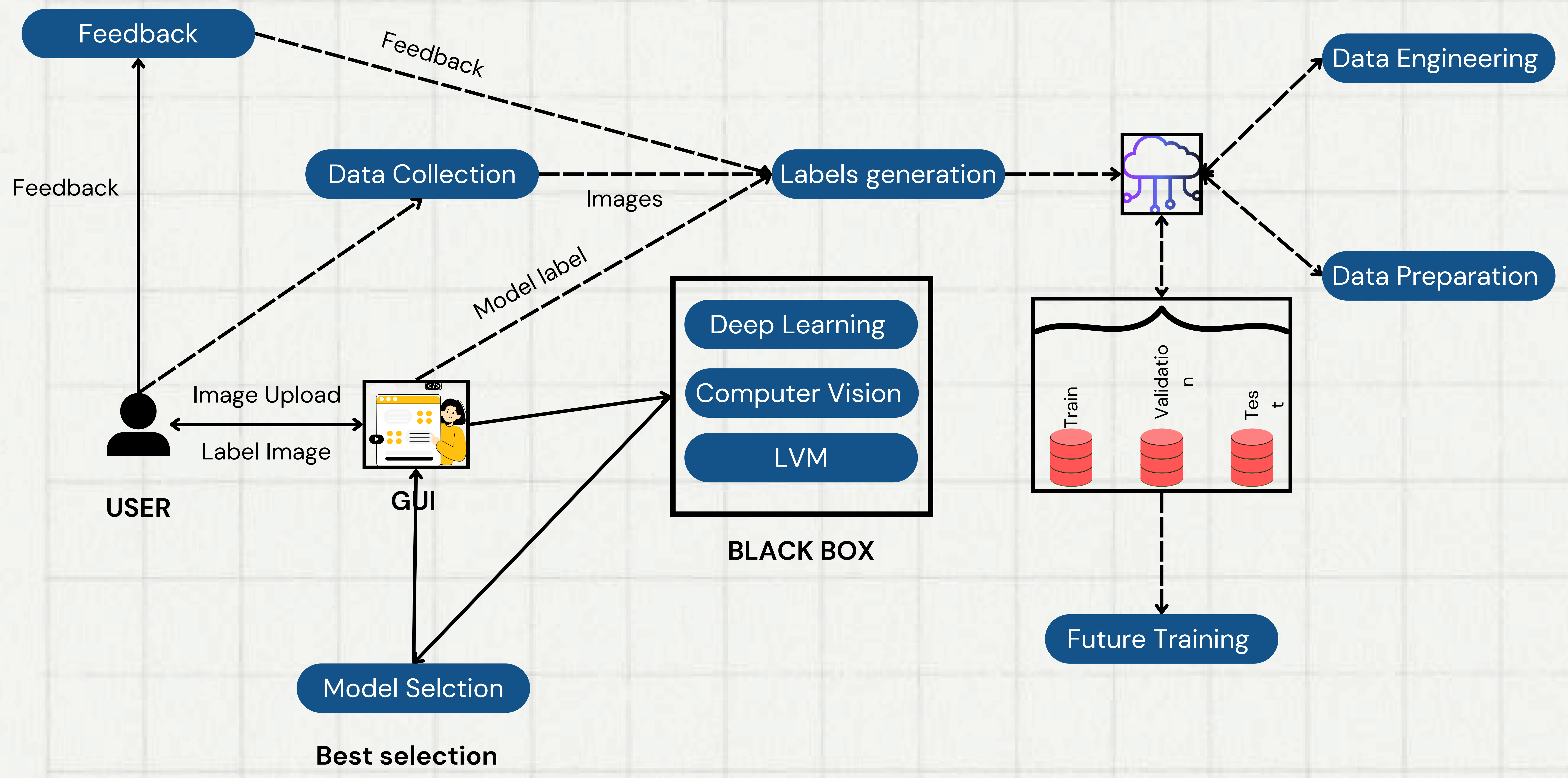
Data Statistics

	PCB Dataset	Wafer Dataset
Data Collection	1,386 images, 4608×3456 pixels	811457 rows
Data Pre-processing	1,386 images, 640 x 640 pixels	172950 rows, 45 x 48 shape
Data Transformation	5,544 images, 640 x 640 pixels	77440, 45 x 48 shape
Data Preparation	4,544 images, 640 x 640 pixels 500 images, 640 x 640 pixels 500 images, 640 x 640 pixels	61952 rows, 45 x 48 shape 7744 rows, 45 x 48 shape 7744 rows, 45 x 48 shape

System Architecture



User Data Flow



Machine Learning Modeling

Model Selection

- Business/Use-Case Alignment: Ensure the chosen model fits the specific defect detection or classification requirements.
- Data Constraints: Consider the volume, variety, and quality of available PCB/Wafer images.
- Computational Resources: Weigh single-stage models like YOLO vs. two-stage models like Faster R-CNN based on real-time needs and hardware limitations.

Model Improvement

- Hyperparameter Tuning: Adjust learning rates, batch sizes, and other parameters to boost accuracy and reduce overfitting.
- Data Augmentation: Apply rotations, flips, or noise injection to enhance model robustness against variability.
- Regularization/Optimization: Techniques such as dropout, weight decay, or advanced optimizers (Adam, SGD with momentum) to stabilize training.

Model innovation

- Architectural Advances: Experiment with novel frameworks (e.g., Vision Transformers, lightweight CNNs) suited to high-resolution defect inspection.
- Scalability & Edge Deployment: Investigate quantization or pruning to enable efficient inference in real-time industrial environments.

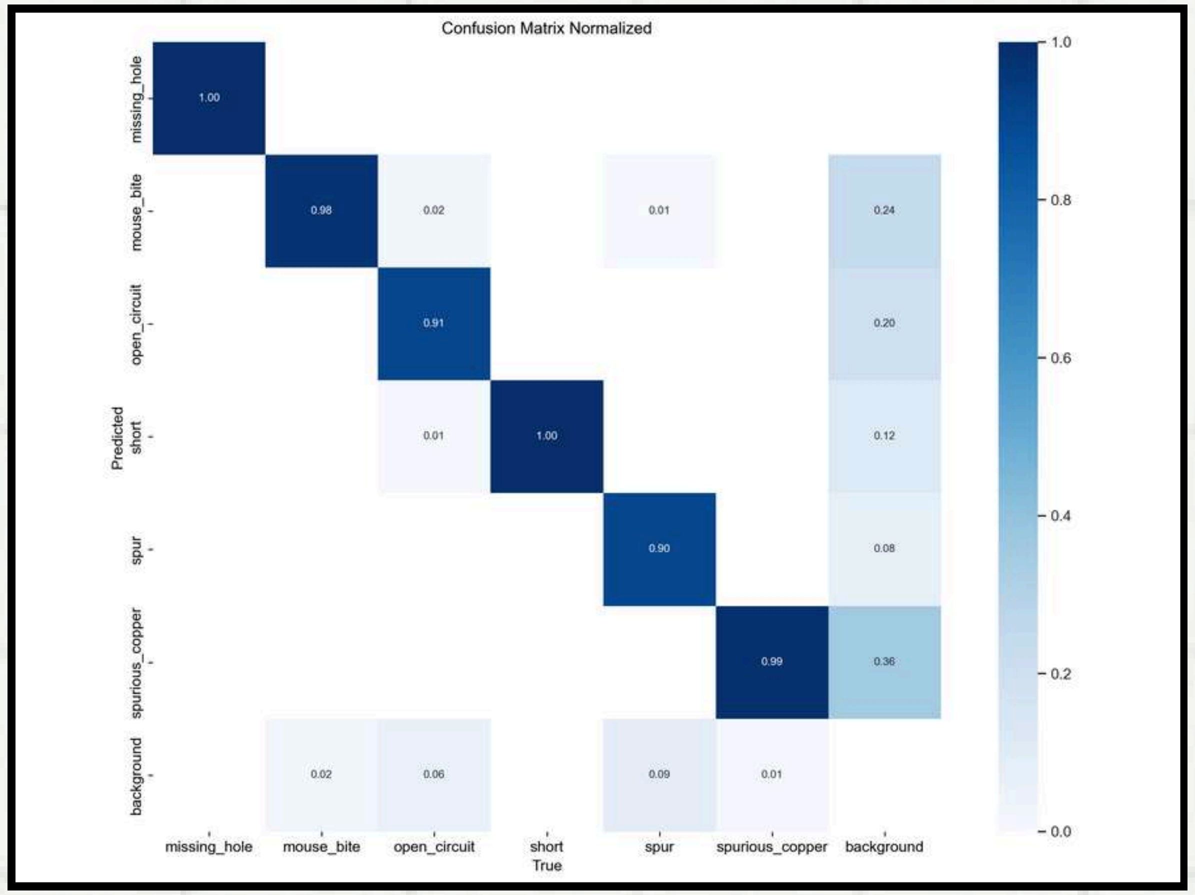
Machine Learning Models

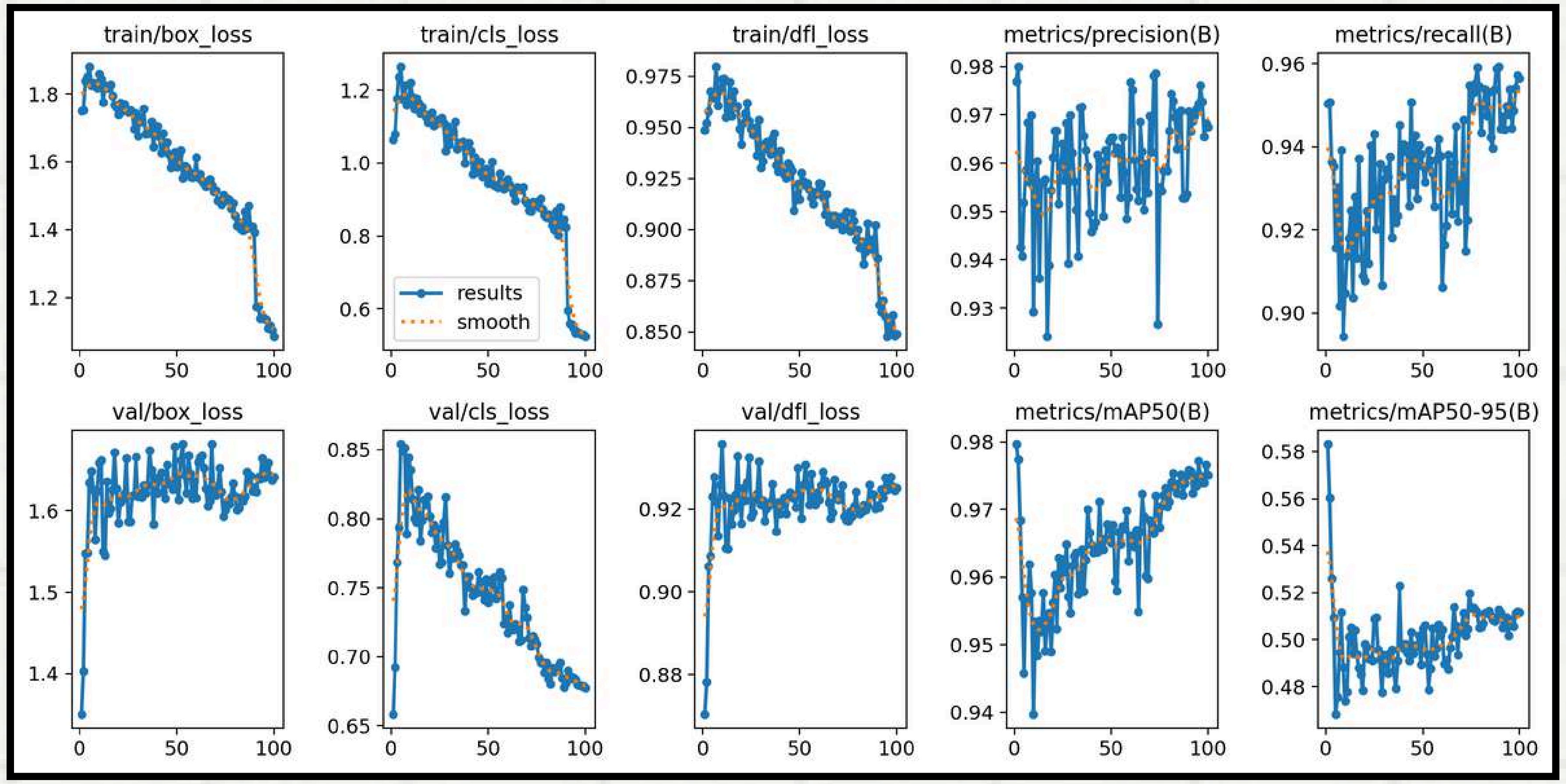
	Yolo	Faster R-CNN	CNN	RestNet	V-DETR
Flexibility	High, supports multiple tasks	High	Moderate	High, adaptable for multiple vision tasks	High, effective for transfer learning
Efficiency on Small Data	Moderate	Moderate	Low	Low	Moderate, better with transfer learning
Preprocessing Required	Minimal, mostly resizing	Minimal	Moderate, region proposals	High, patches and positional encoding	Minimal, typically resizing and normalization
Space Complexity	Moderate, optimized for real-time	Moderate	High, larger due to feature maps and proposals	High, transformer layers require more memory	Moderate to High, increases with depth
Weakness	May miss small objects	Struggles with complexity	Slow on inference	Requires high resources, prone to overfitting	Relatively slow inference with deeper variants

Machine Learning Results

YOLO

PCB Dataset





Model Performance Summary

- 100% accuracy for "missing_hole" and "short".
- Over 98% accuracy for "mouse_bite", "open_circuit", and "spurious_copper".
- Confusion Areas:
  - Notable misclassifications with the "background" class.

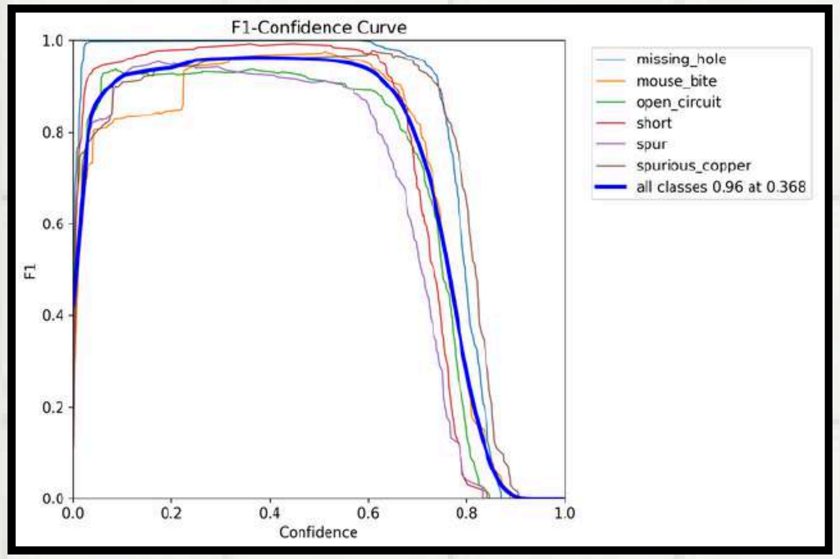
Training & Validation Insights

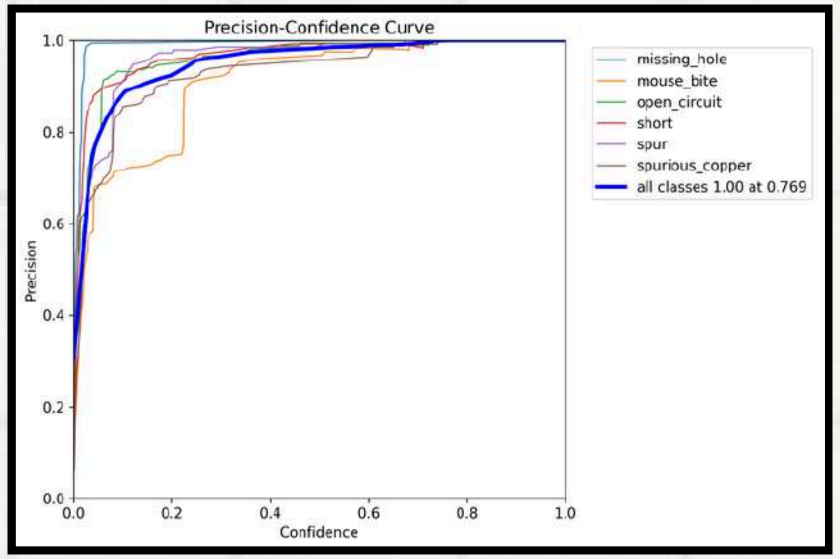
- All training losses (box, cls, dfl) show a consistent decline over epochs.
- Slight overfitting indicated by rising validation losses after early epochs.
- Model Learning Trends:
  - Gradual and stable increase in precision and recall metrics.
  - mAP@50 steadily improves, nearing 0.98, indicating strong bounding box accuracy.

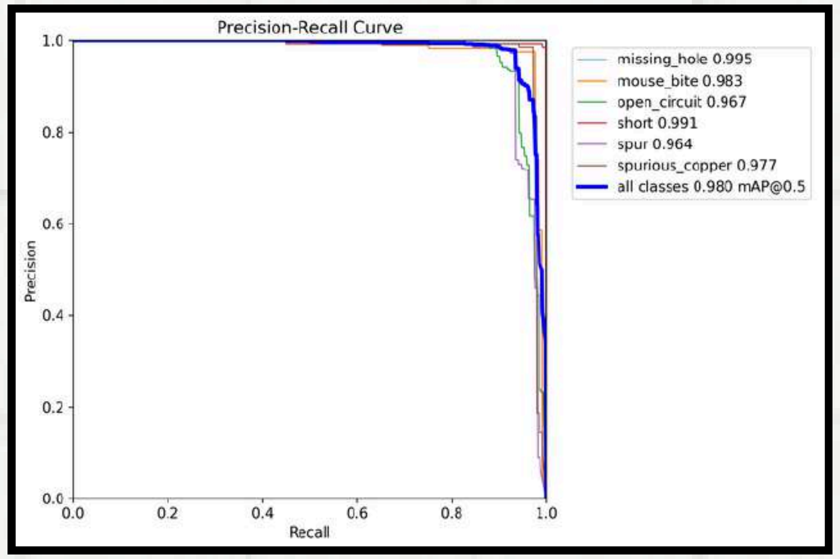
Machine Learning Results

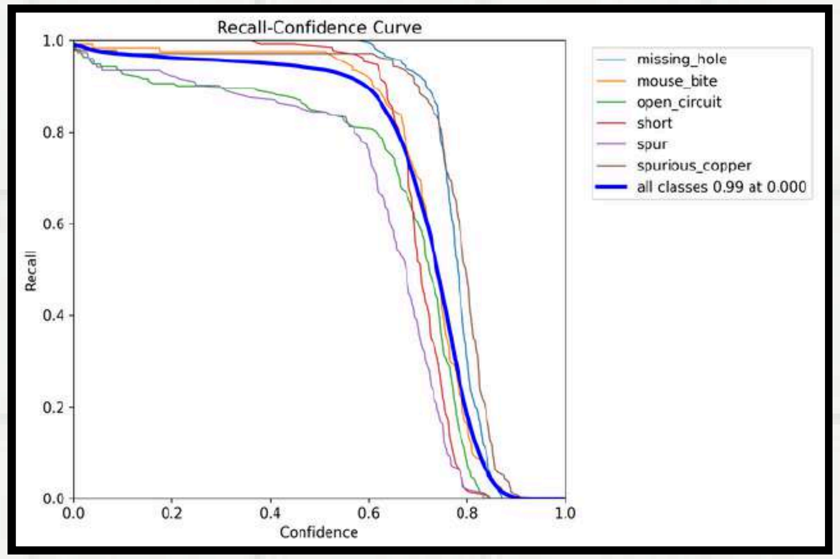
YOLO

PCB Dataset









High Overall Performance:

- Achieved F1-score of 0.96 at an optimal confidence threshold of 0.70.
- Precision peaks at 1.00 when confidence reaches 0.8, indicating extremely reliable predictions at higher confidence levels.

Trade-off Observations:

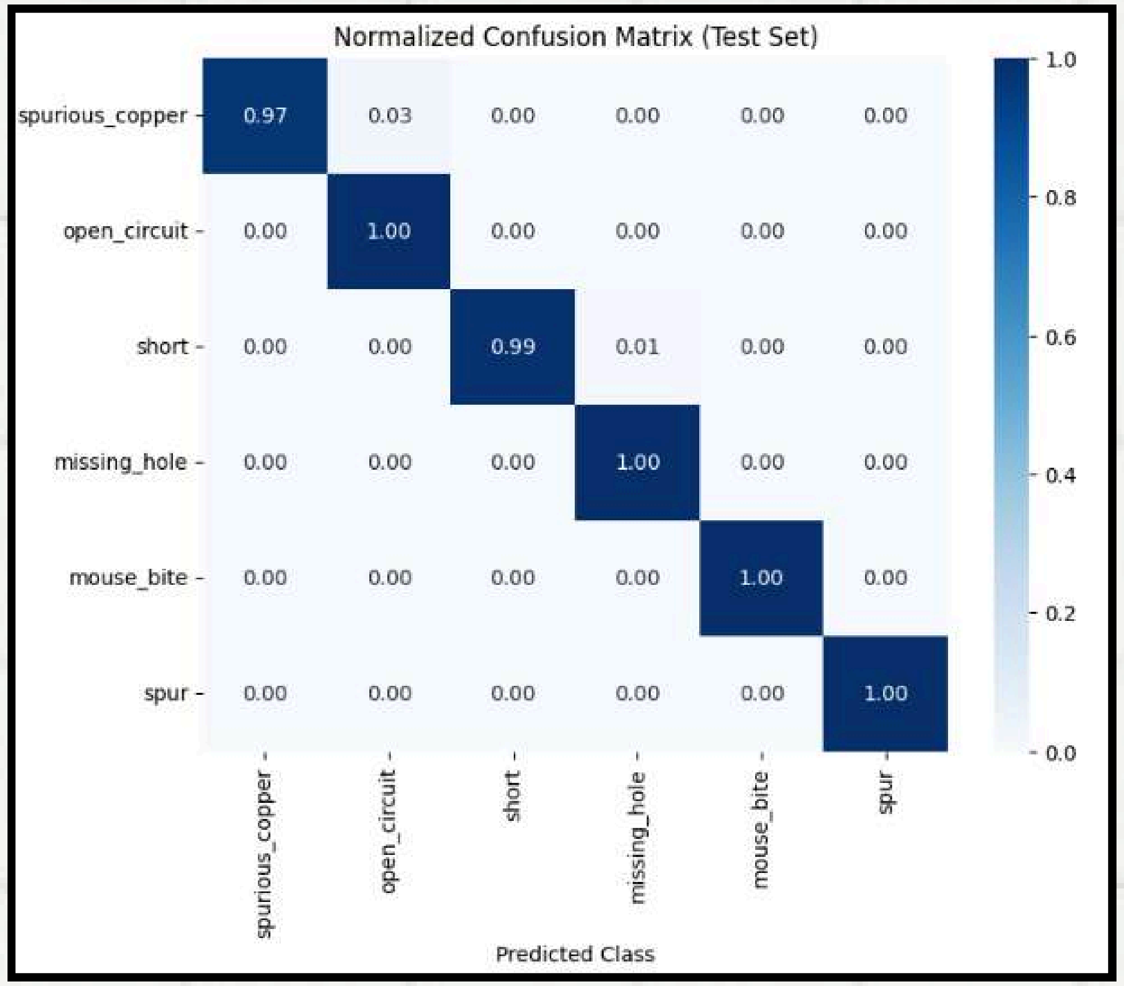
- Recall drops rapidly past 0.6 confidence, as seen in the Recall-Confidence Curve, signaling that overconfidence can suppress true positive detection.
- Precision-Recall Curve confirms mAP@0.5 = 0.980, reinforcing strong class-wise performance.

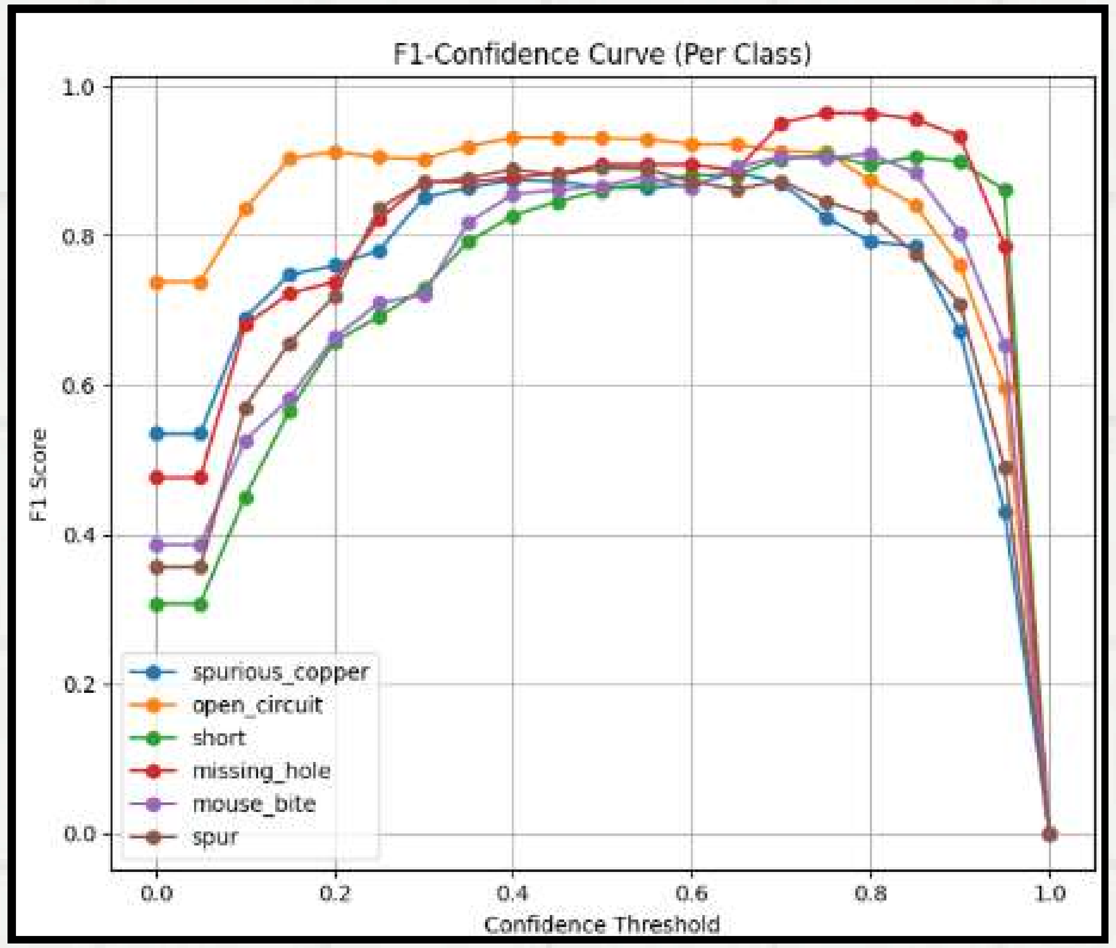
- For real-world deployment, set the operating threshold between 0.35–0.45 to balance both recall and precision optimally.
- Consider calibration or confidence smoothing for production models to reduce sharp drops in recall.

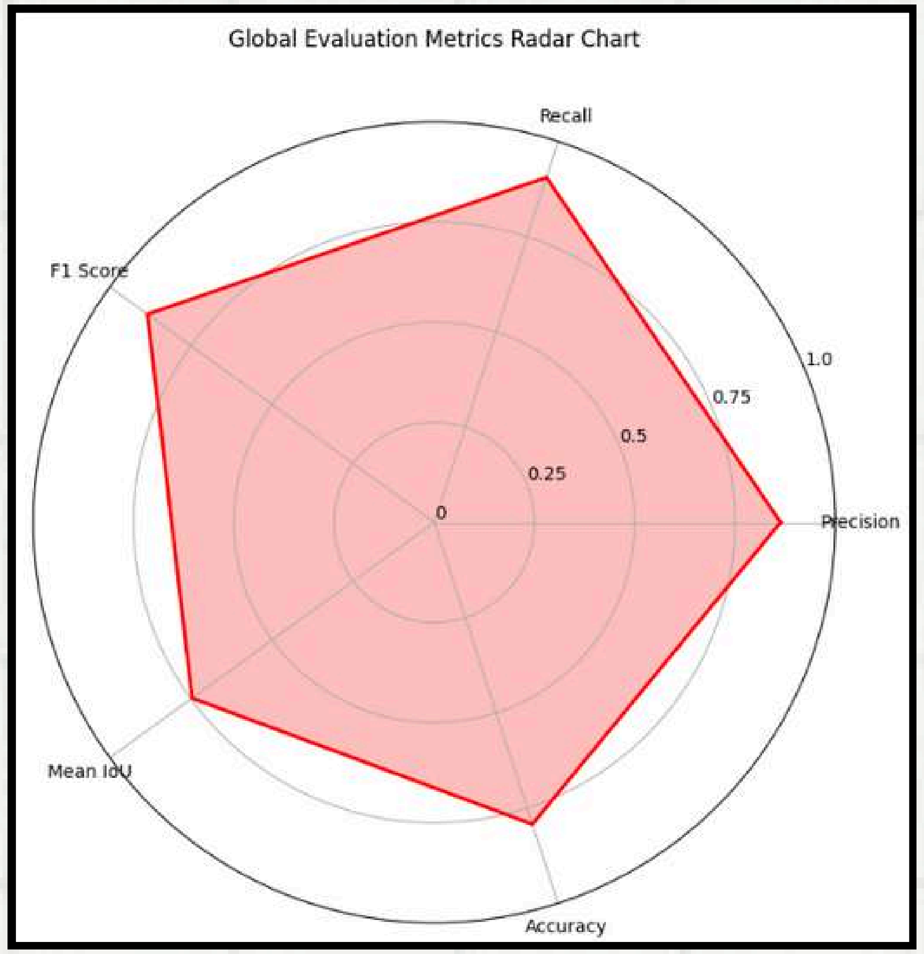
Machine Learning Results

Faster RCNN

PCB Dataset







High Accuracy: Entries along the diagonal indicate high classification accuracy for respective defect types, showing the model's precision.

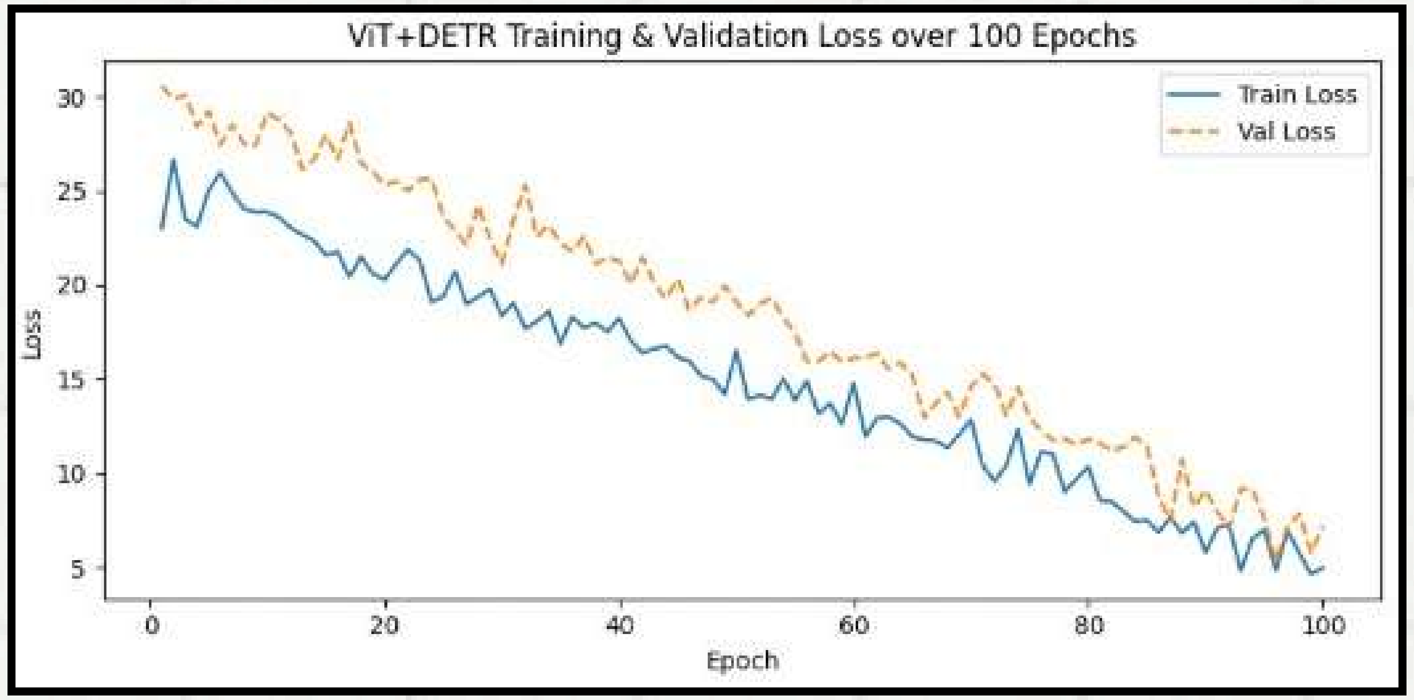
Different lines represent various defect types, highlighting which defects are easier or harder to detect accurately & Indicates model’s effectiveness at balancing false positives and false negatives across the confidence spectrum.

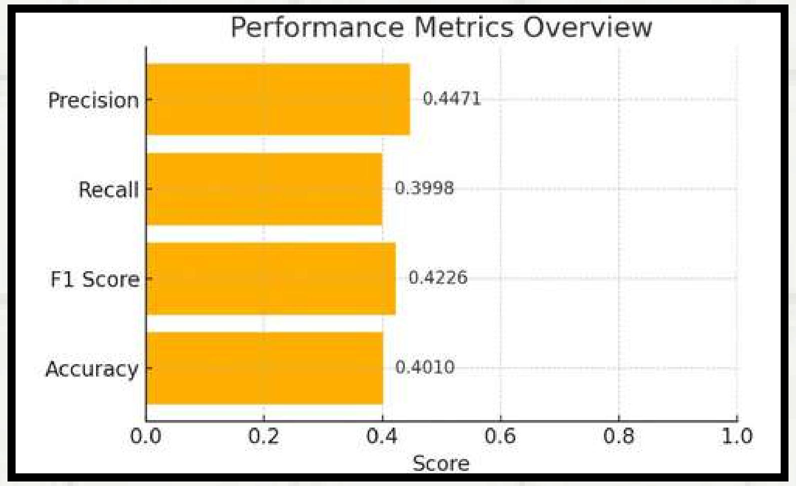
Showcases the model's performance with Precision at 0.75, Recall at 1.0, F1 Score around 0.9, Mean IoU at 0.25, and Accuracy at 0.5, providing a visual representation of its capabilities in various key areas of defect detection.

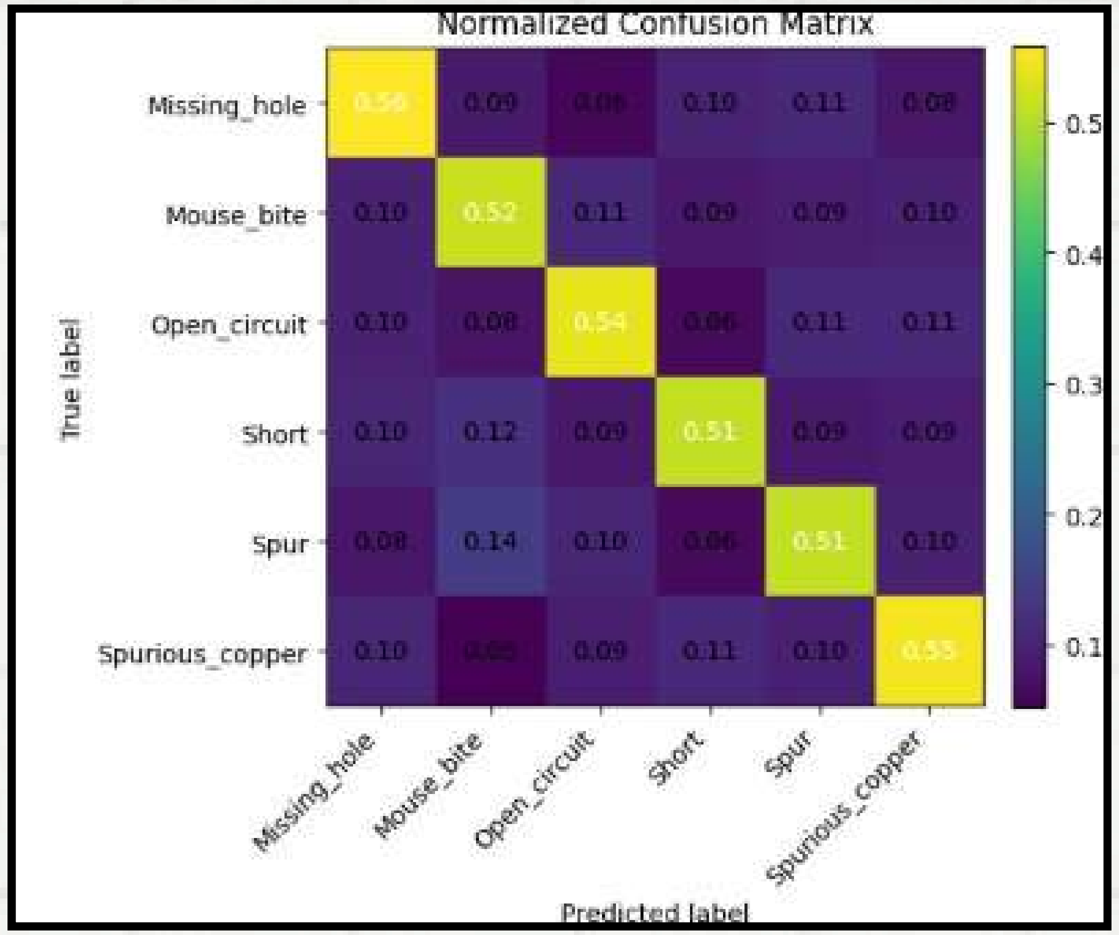
Machine Learning Results

V-DETR

PCB Dataset





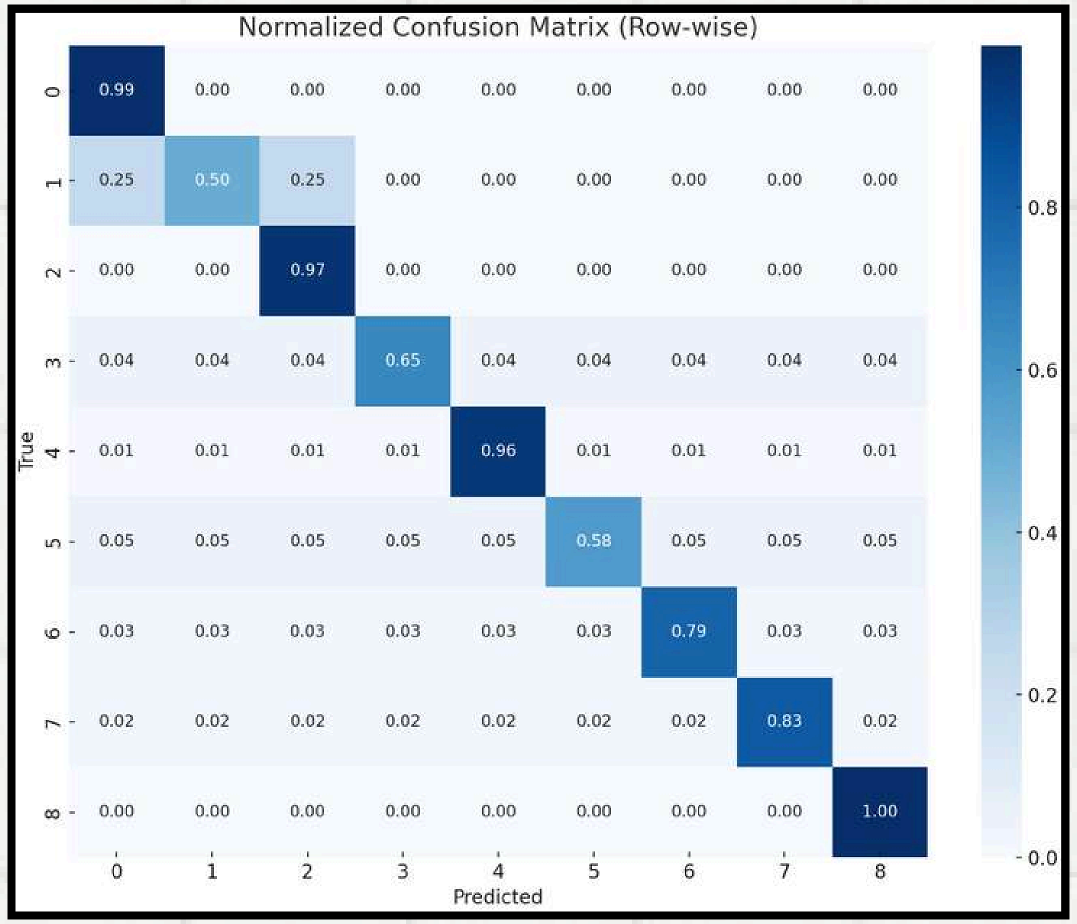


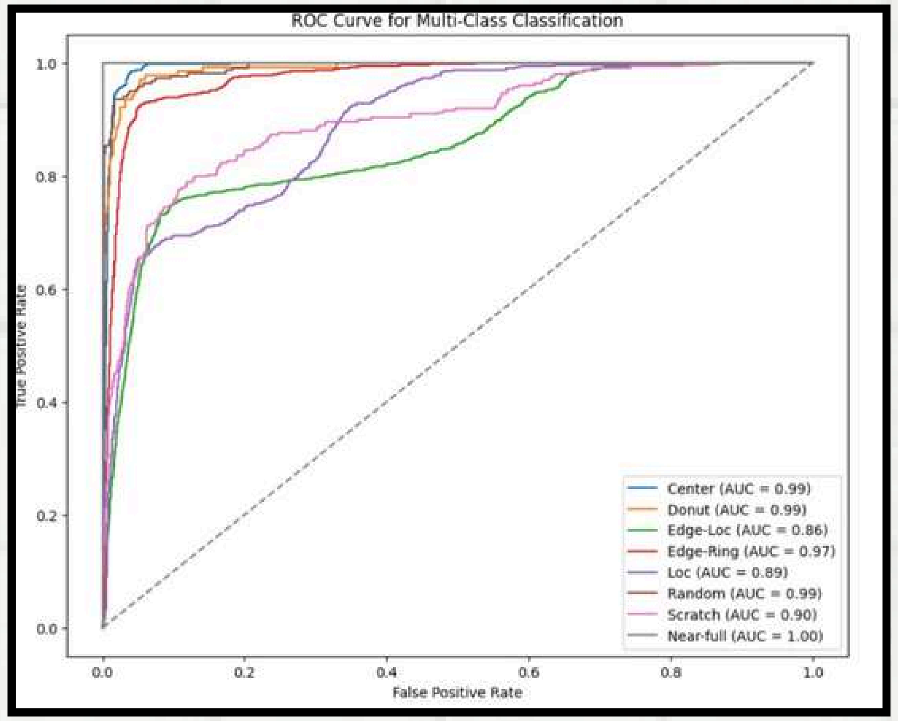
- Consistent Downward Trend: Both training and validation loss show a steady decrease, indicating effective learning and convergence over time.
- Well-Tuned Training: The curve pattern confirms proper use of techniques like early stopping or learning rate scheduling to stabilize training.

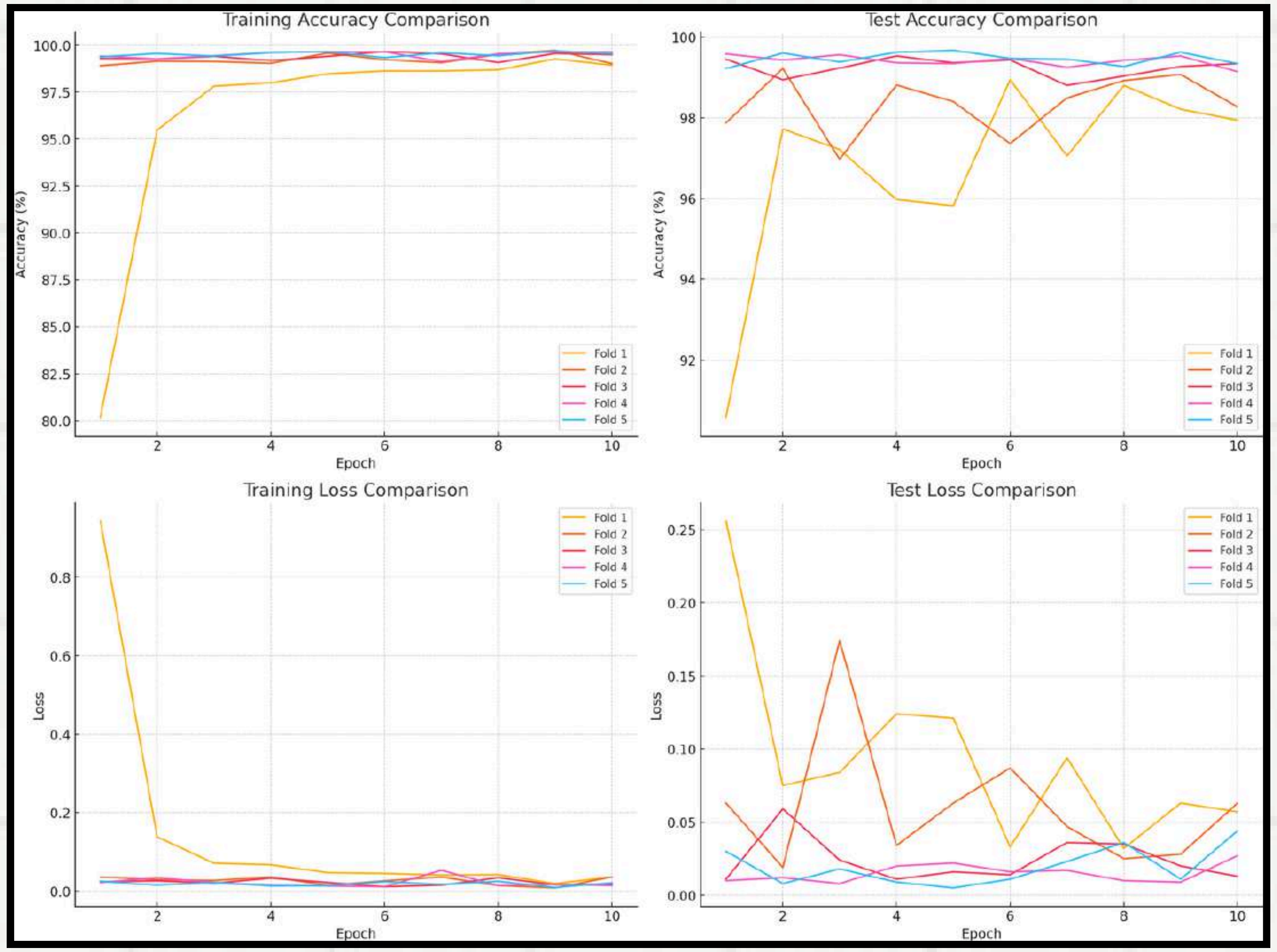
Machine Learning Results

Model CNN

Wafer Dataset







- Achieved near-perfect accuracy for certain classes (e.g., class 0 with 0.99 and class 8 with 1.00).
- Strong performance across all classes with low misclassification rates in most cases.
- The model has the worst performance for the Edge-Loc class.

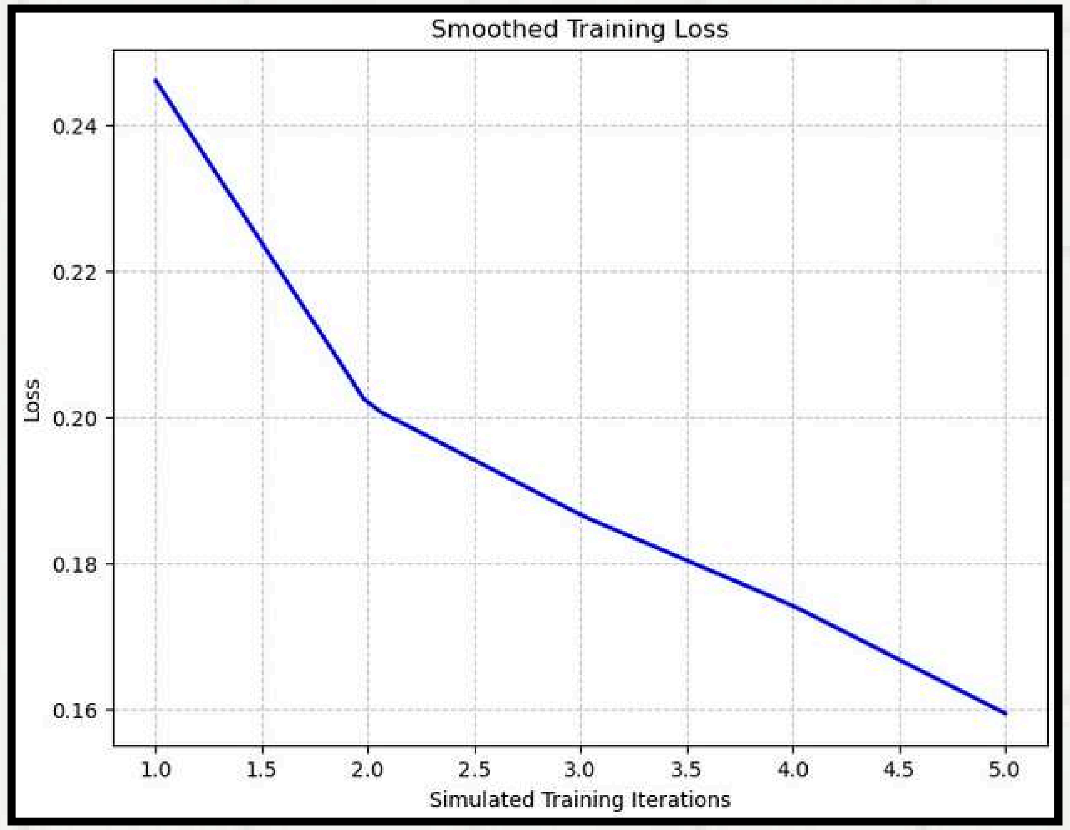
Confusion Matrix Observations:

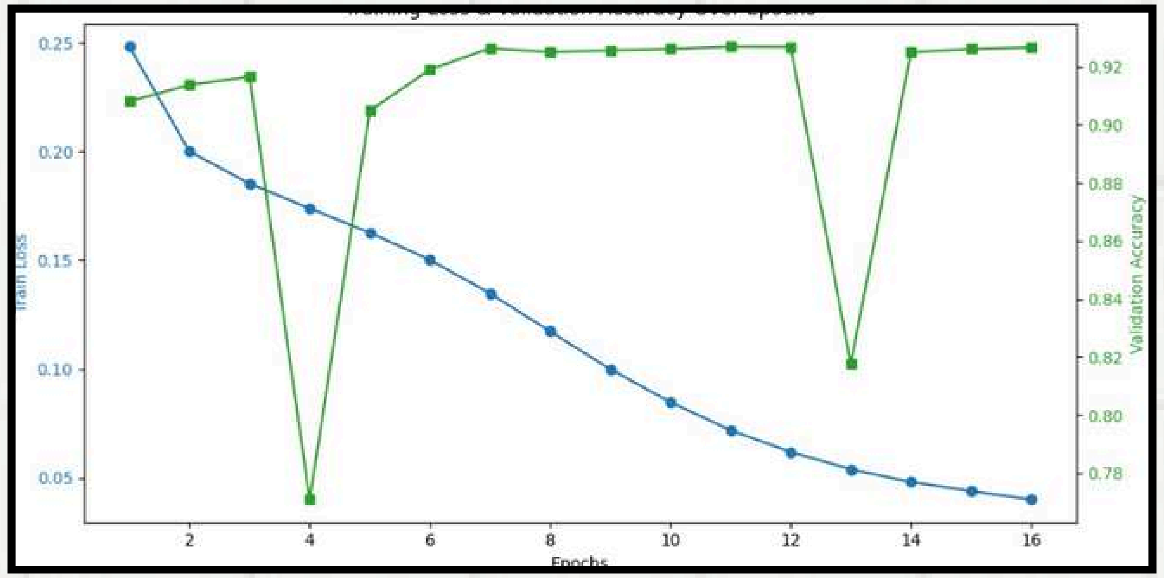
- Class 1 (row index 1) showed confusion with class 0 and 2 (each 25%), suggesting a need for improved feature separation.
- Class 3 had more dispersed misclassifications, indicating annotation or representation issues.

Machine Learning Results

ResNet

Wafer Dataset



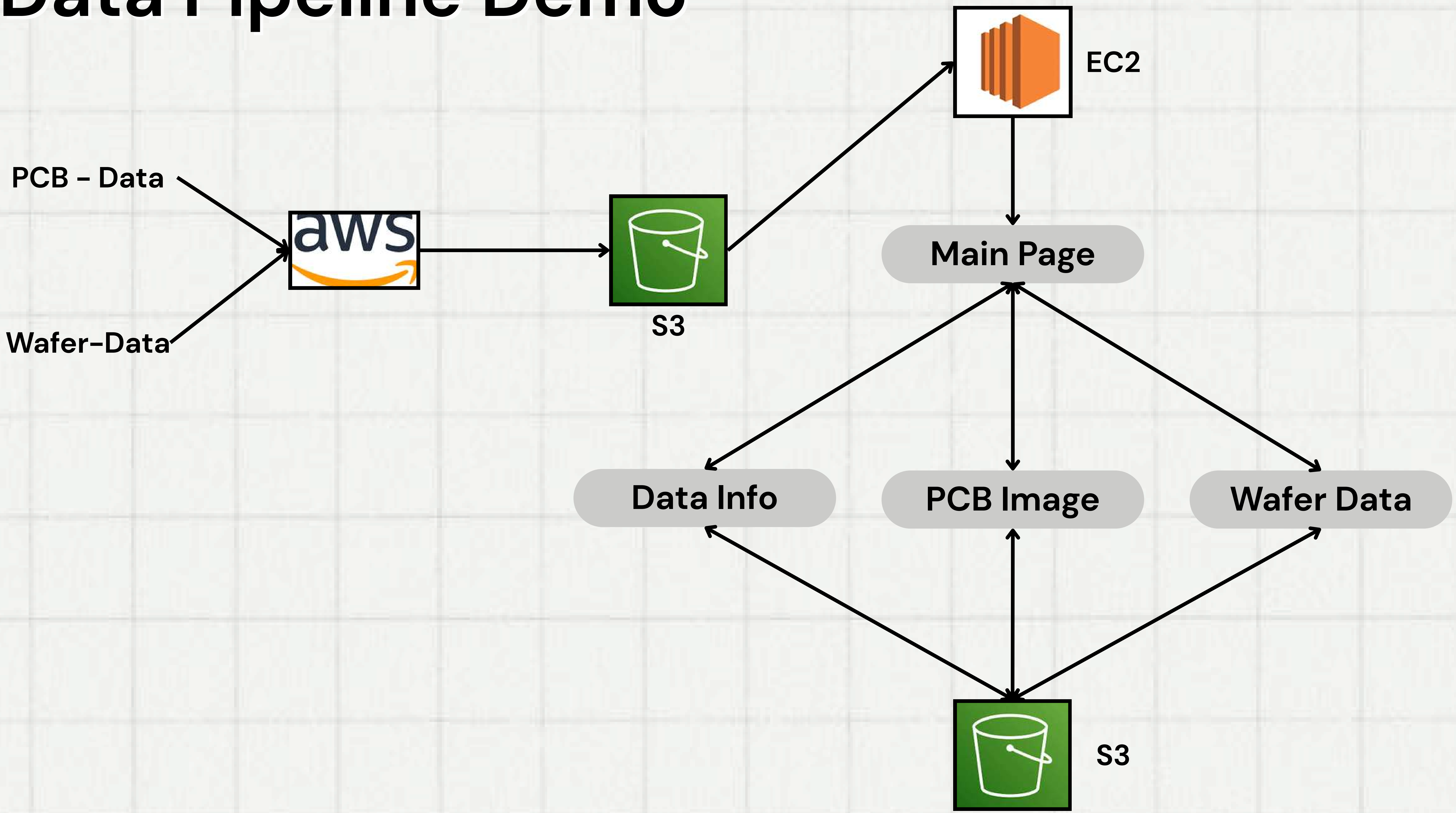


- Training loss shows a smooth, consistent downward trend.
- Validation accuracy dips notably around epochs 4 and 13, but quickly recovers, indicating temporary fluctuation rather than overfitting.
- Final validation accuracy stabilizes above 0.92, showing the model generalizes well on unseen data.
- Smoothed loss trend confirms controlled learning without overfitting, backed by validation recovery.

Machine Learning Results Comparision

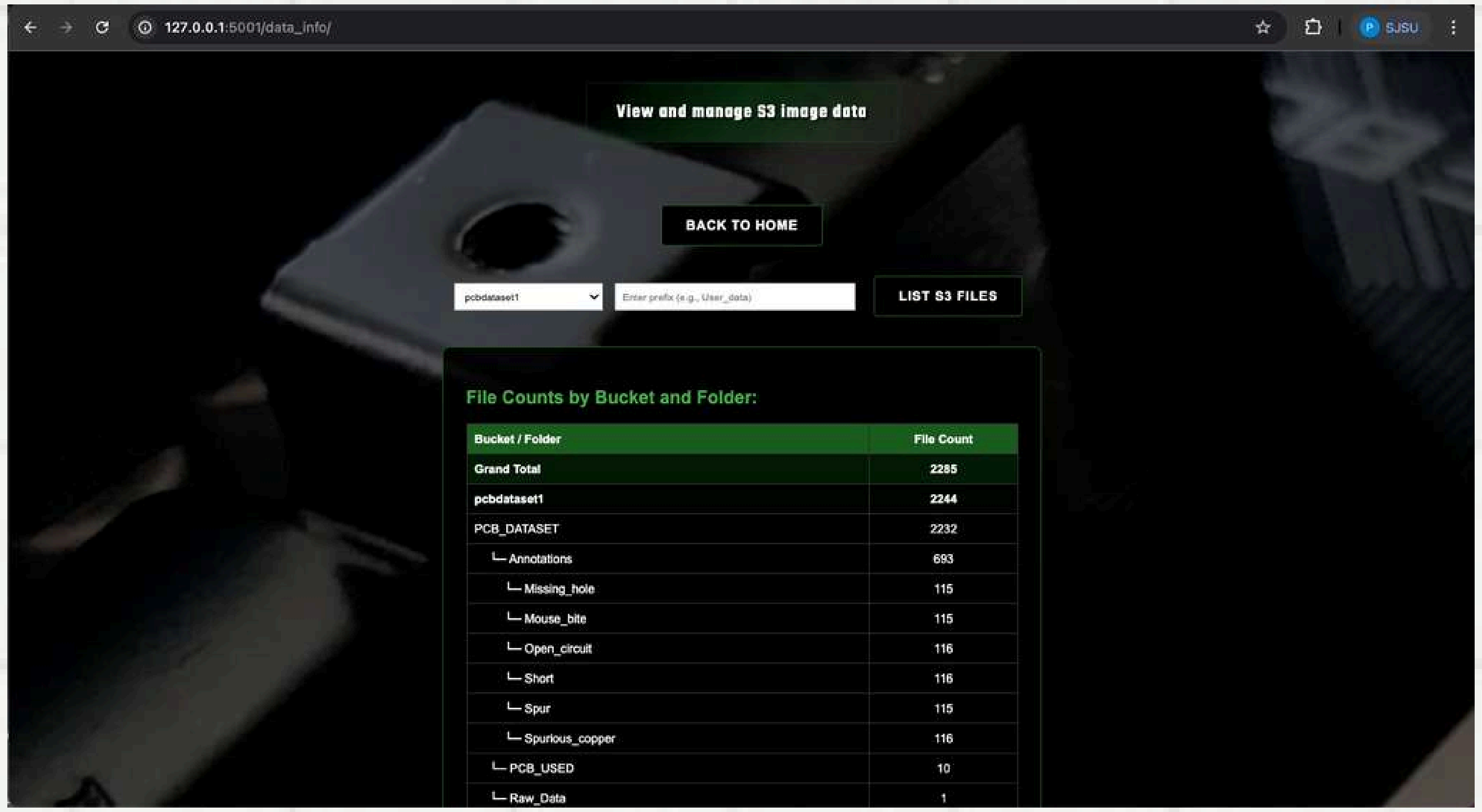
	Yolov8	Faster-RCNN	CNN	ResNet	V-DETR
Dataset	PCB Dataset	PCB Dataset	Wafer Dataset	Wafer Dataset	PCB Dataset
Accuracy	98%	87%	95%	92.66%	40%
Recall	93%	90%	90%	93%	40%
Precision	95%	86%	98%	92%	44%
Inference Time	1 sec	2 sec	Moderate	3 sec	High
Format	Object Localization & Classification	Object Localization & Classification	Classification	Classification	Object Localization & Classification

Data Pipeline Demo



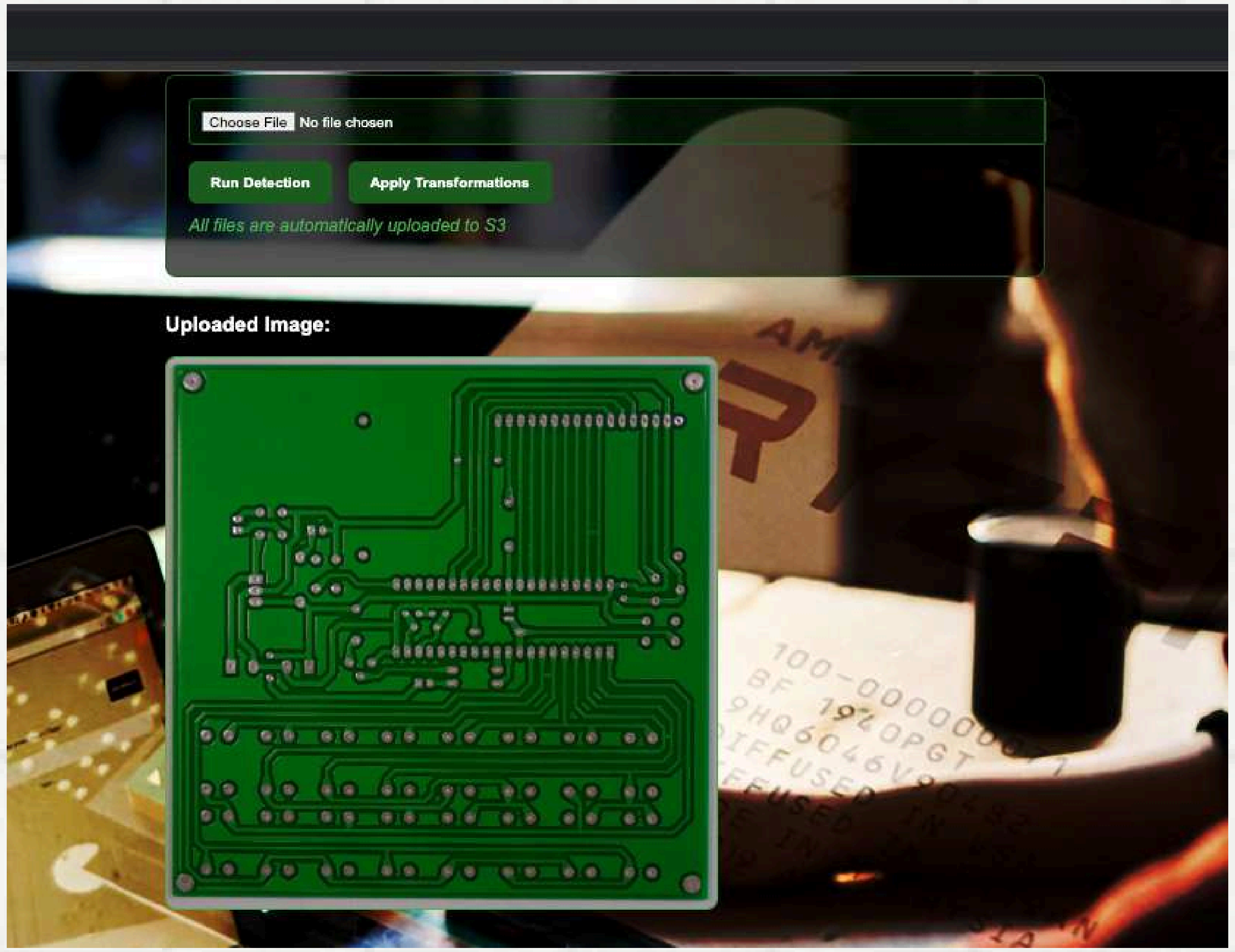
Web Interface

Welcome Page

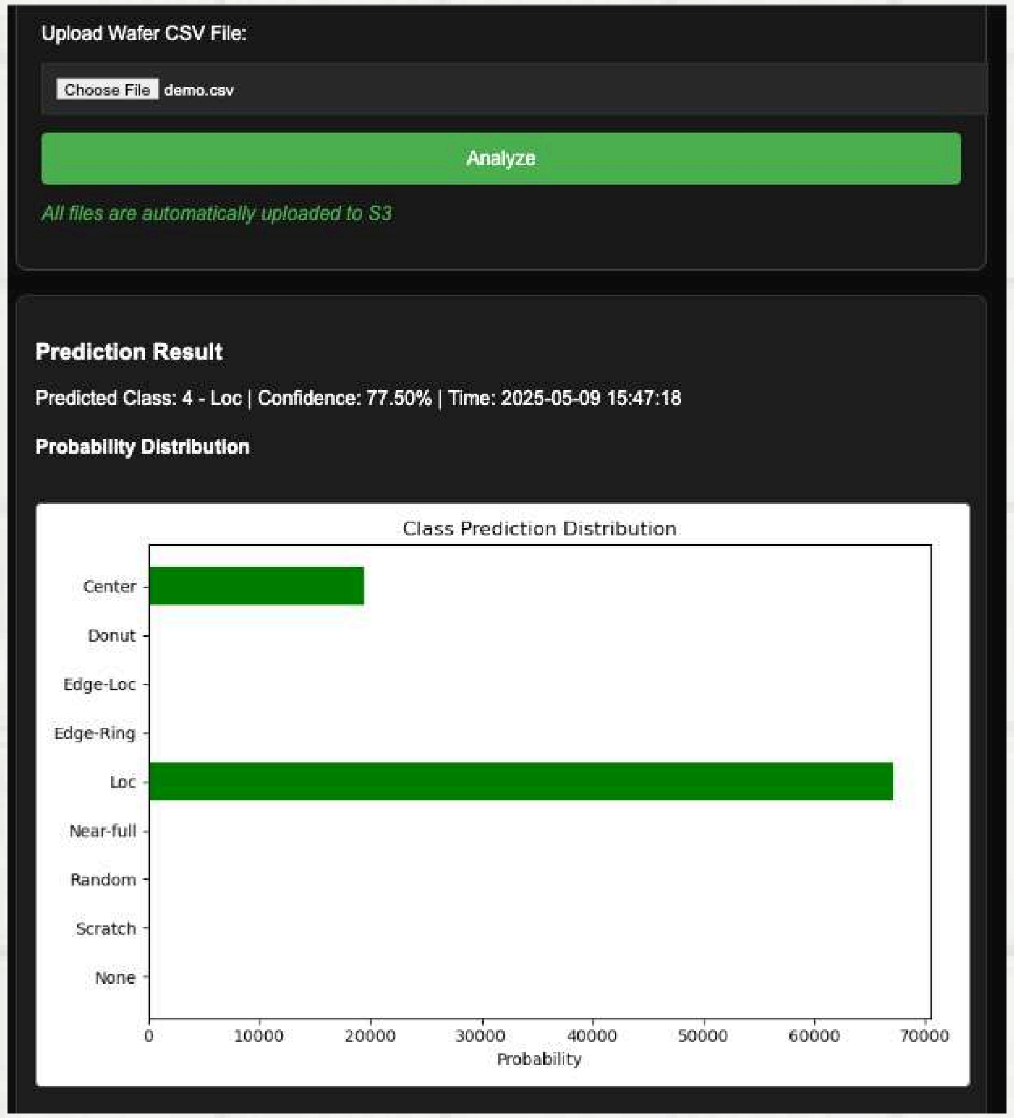


Data Information and logs

Web Interface



Web Interface PCB Defect Analysis



Wafer Detection Web Interface

Web Interface

Feedback Statistics:

Positive Feedback (Yes)
3

Negative Feedback (No)
4

Feedback by Defect Type:

Defect Type	Yes	No	Satisfaction Rate
Loc	2	4	33.3%
Edge-Loc	1	0	100.0%

Recent Feedback:

Timestamp	Feedback	Defect Type	Image
2025-05-09 15:49:28	Yes	Edge-Loc	resize_x_5.csv
2025-05-09 14:46:57	Yes	Loc	demo.csv
2025-05-09 13:20:40	No	Loc	demo.csv
2025-05-09 13:20:25	No	Loc	demo.csv
2025-05-09 13:20:10	No	Loc	demo.csv

Feedback Analytsis

Conclusion

- Developed a multi-model AI system using YOLOv8, CNN, ResNet, Faster R-CNN, and V-DETR for PCB and wafer defect detection.
- Integrated and preprocessed complex wafer (LSWMD) and PCB (PKU) datasets into a unified processing pipeline.
- Designed an interactive Streamlit and Flask UI for real-time defect visualization, model feedback, and analytics.
- Enabled cloud deployment via AWS (EC2, S3) for scalable, remote-access inspection systems.
- Incorporated human-in-the-loop learning for continuous improvement and adaptability to new defect types.

Thank you!