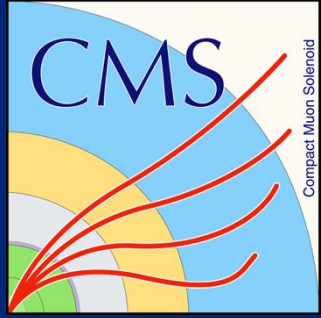


中国科学院高能物理研究所

*Institute of High Energy Physics, Chinese Academy of Sciences*



# Anomaly Detection for Defect Identification in CMS HGCAL Assembly at IHEP

Suyun Huang

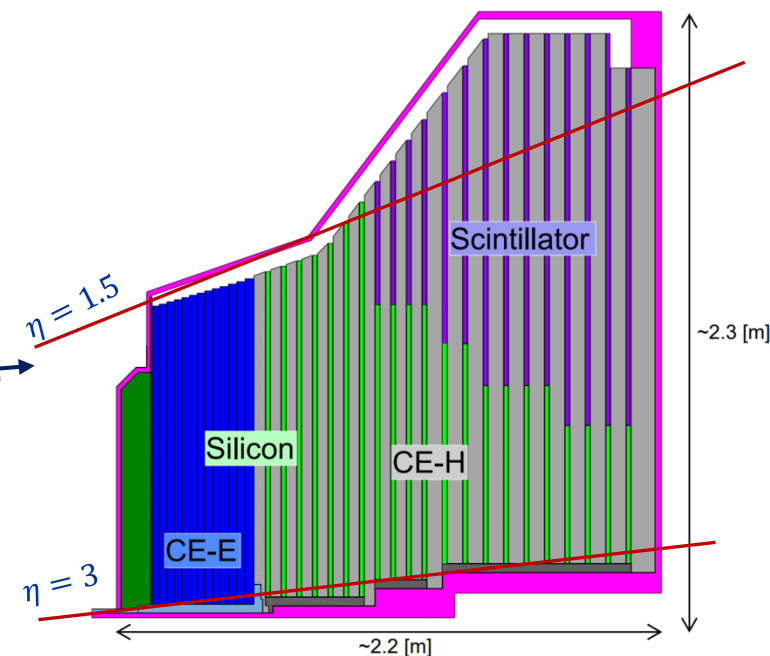
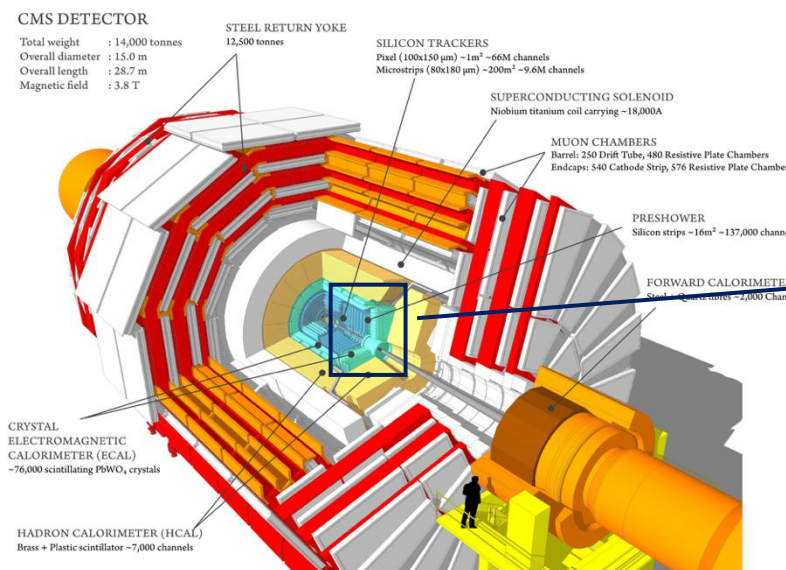
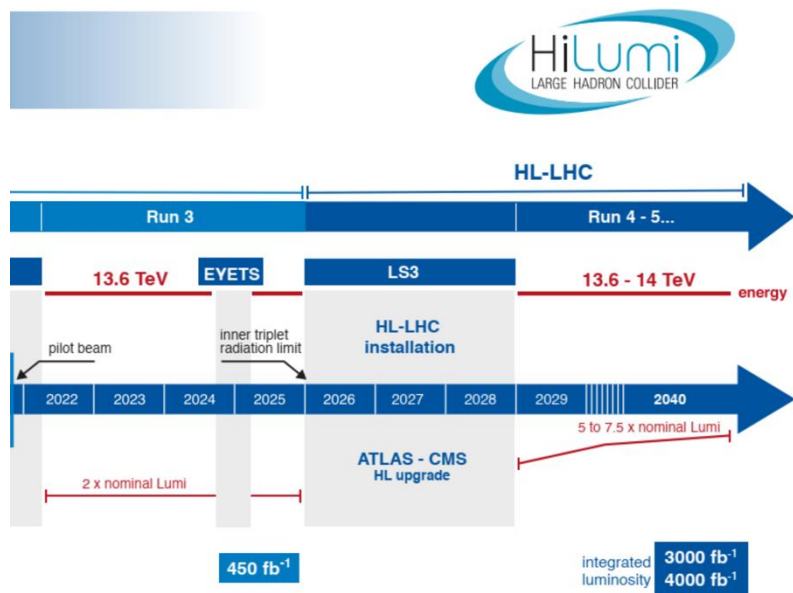
2026.4.15

# What is HGCal

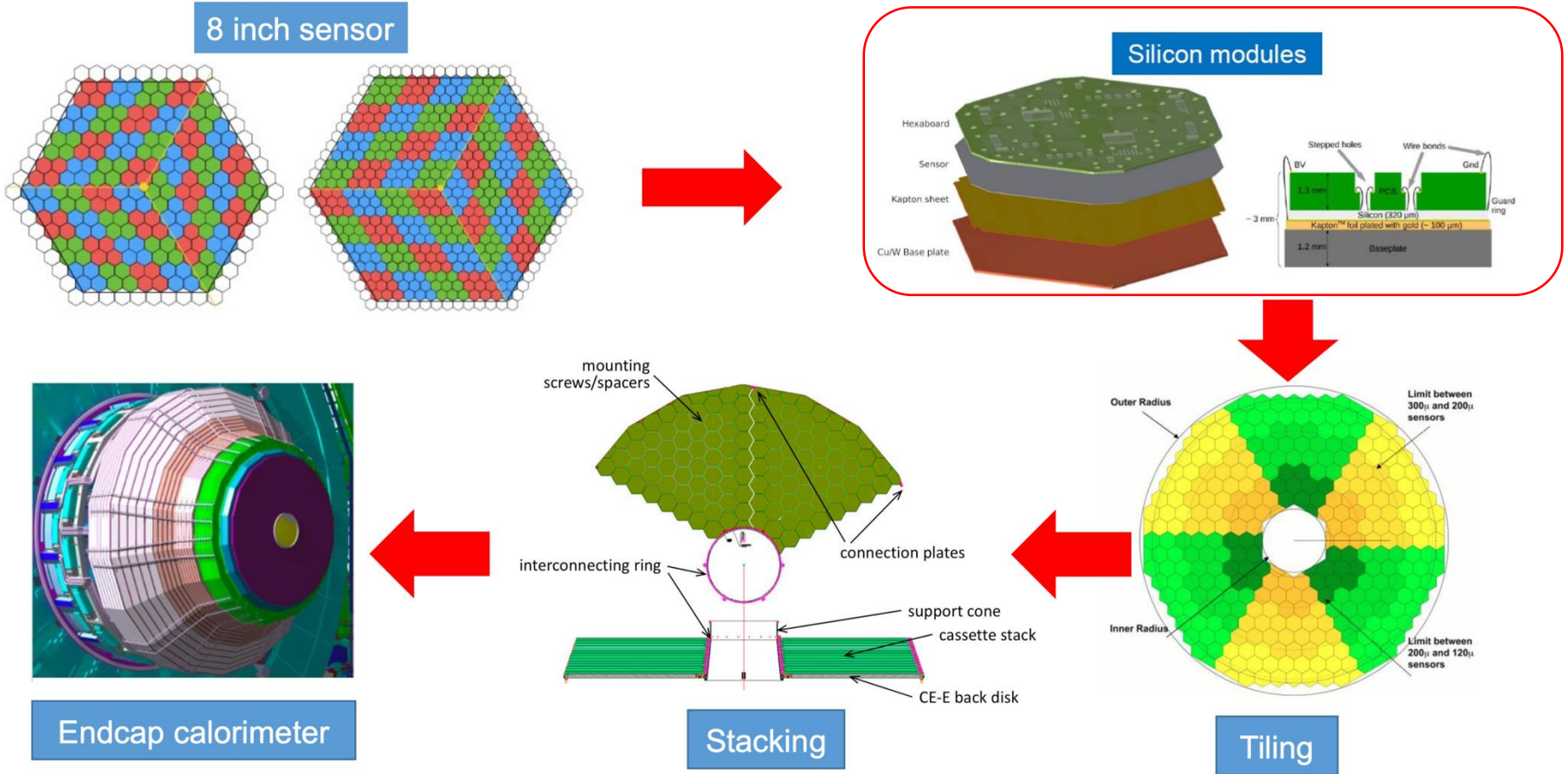
HL-LHC will operate at  $5 \times 10^{34} \text{ cm}^{-2} \text{ s}^{-1}$  with pileup of  $\sim 140$  events.

Existing CMS endcap calorimeters cannot cope with the expected radiation or pileup.

**HGCal (High Granularity Calorimeter)** is the new **endcap calorimeter** of the CMS experiment at the HL-LHC

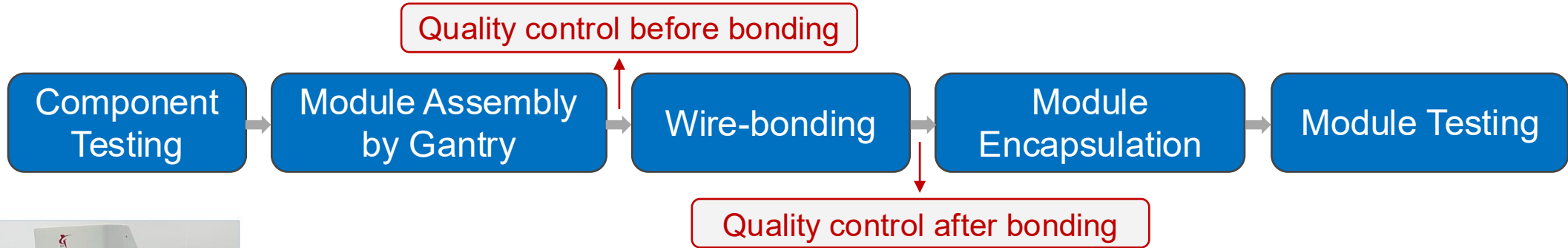


# HGCal Assembly process



IHEP是六个模块组装中心之一，已经组装了数百块模块，速度可以达到16块/天

# Why image recognition matters in HGCal assembly

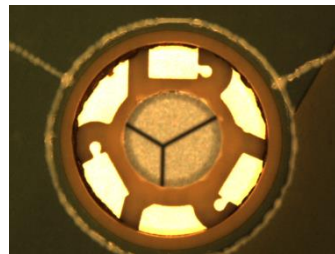


## Before bonding :

Glue leakage or contaminants in bonding holes → reduces bonding success rate

## After bonding:

Wire defects (breakage, bending, missing) → affects electrical reliability



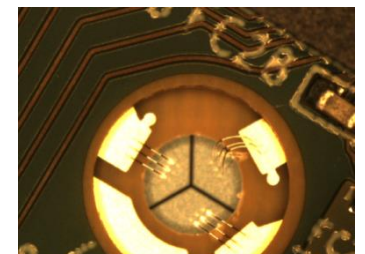
normal



glue leakage



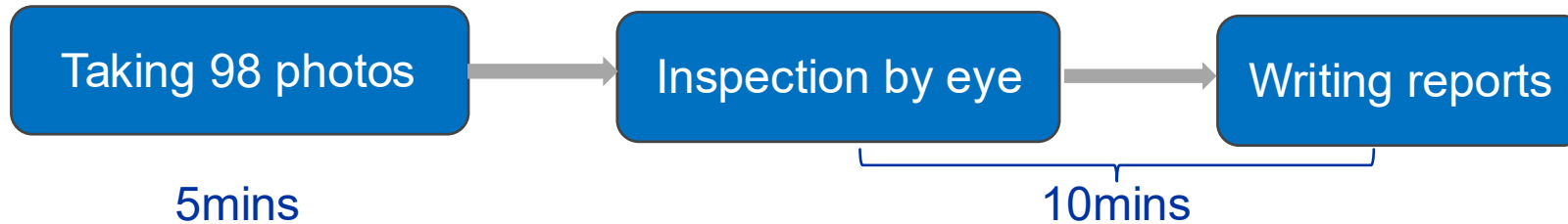
normal



bad wire

# Why image recognition matters in HGCal assembly

Traditional approach:

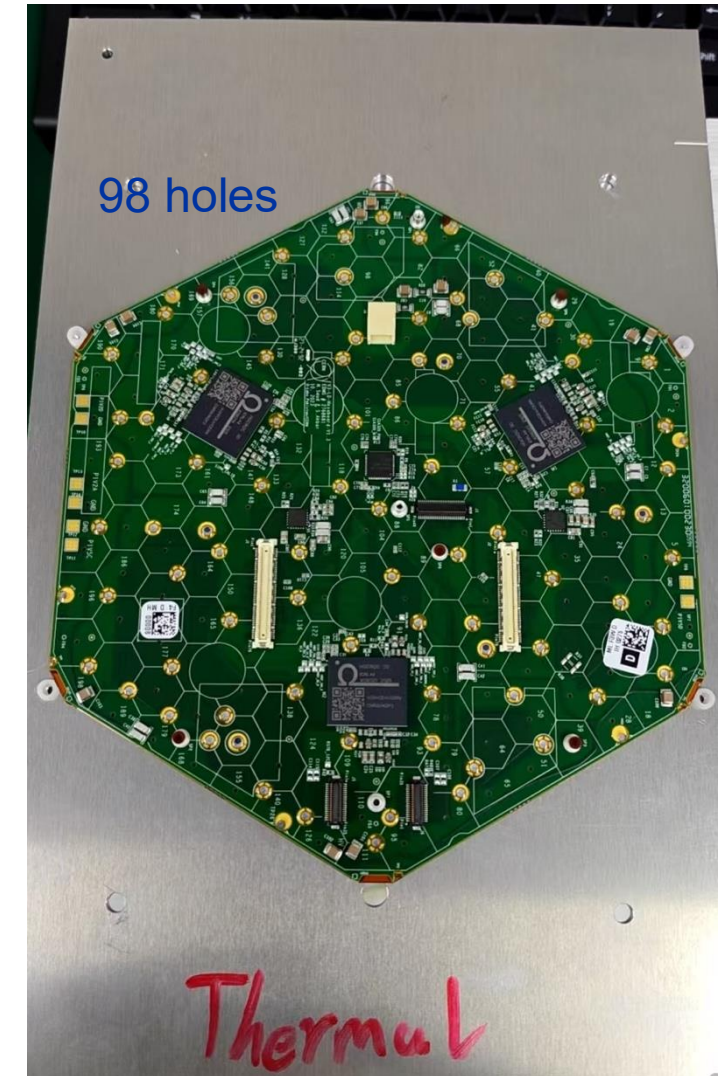


manual inspection: Inefficient, subjective, inconsistent

Goal:



Develop automated image recognition systems using machine learning

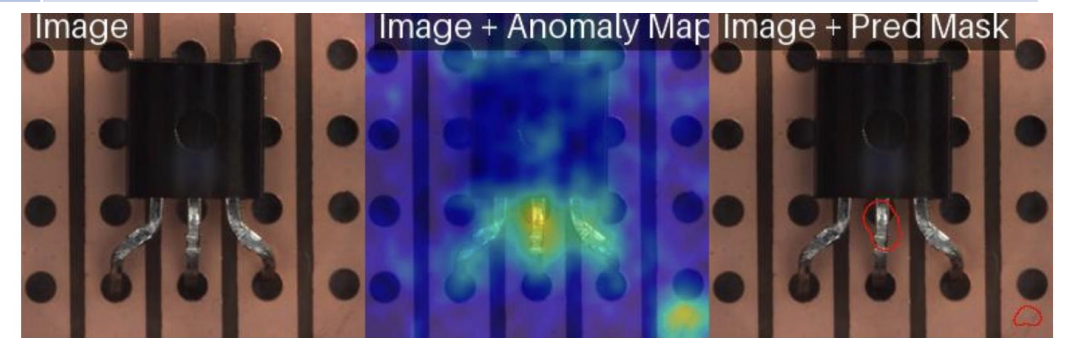


# Method roadmap: YOLO and PatchCore

## YOLO (You Only Look Once)

## PatchCore (Anomalib)

|                  |   |                                       |
|------------------|---|---------------------------------------|
| Type :           | <b>Supervised</b> object detection      | <b>Unsupervised</b> anomaly detection |
| Core principle : | One-pass prediction of location + class | Learns normal feature distribution    |
| Deployment :     | Fast, easy                              | Slower, more complex                  |
| Data :           | Requires <b>labeled</b> defect samples  | Requires only <b>normal</b> samples   |
| use case :       | Known, common defect types              | Rare, unknown, defects                |



- Before bonding: (1) YOLO (2) YOLO-seg+PatchCore
- After bonding: (1) YOLO (2) PatchCore (3) YOLO+PatchCore

# Before bonding: YOLO

**Label tool:** labellmg v1.8.6 (YOLO format)

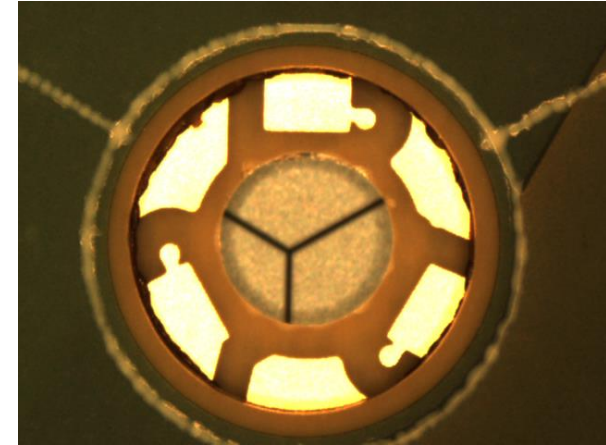
**Class:** 1 (“glue”)

**Total images:** 10,948 (no data augmentation)

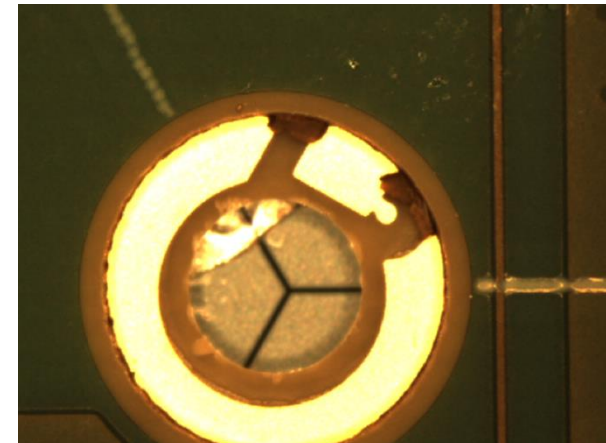
**Image Source:** Taken by OGP smartscope

## Dataset:

| class  | Train | Val  | Test |
|--------|-------|------|------|
| glue   | 308   | 38   | 39   |
| normal | 7727  | 969  | 958  |
| total  | 8035  | 1007 | 997  |

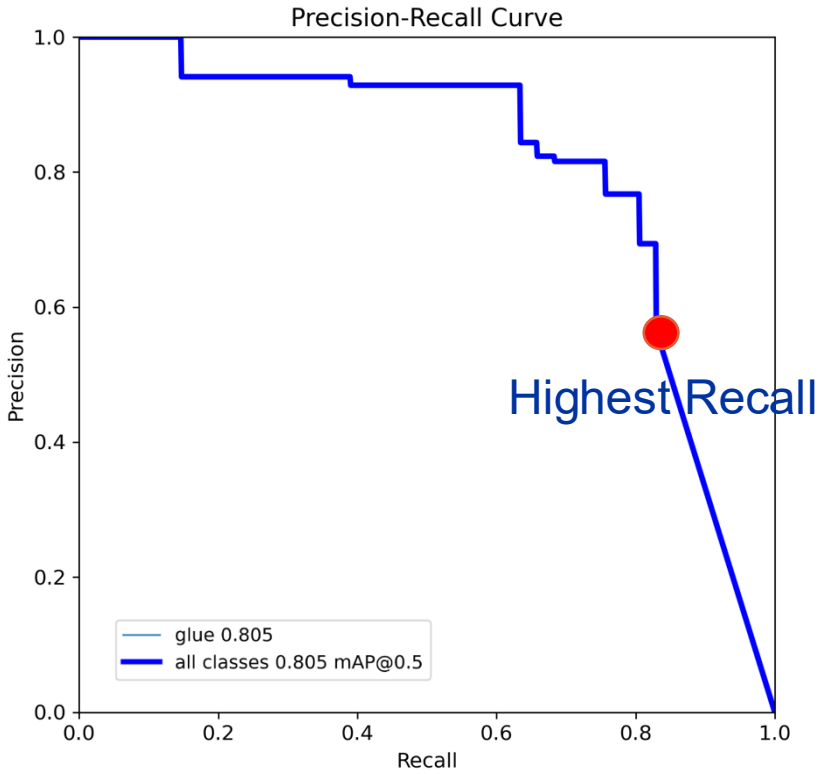


normal

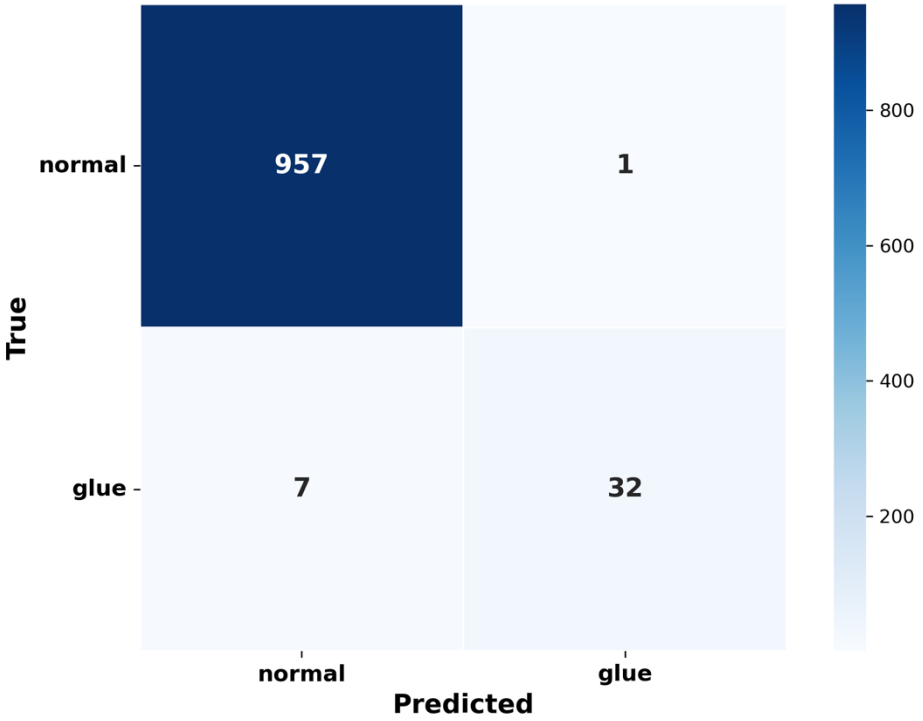


glue leakage

# Before bonding: YOLO Performance



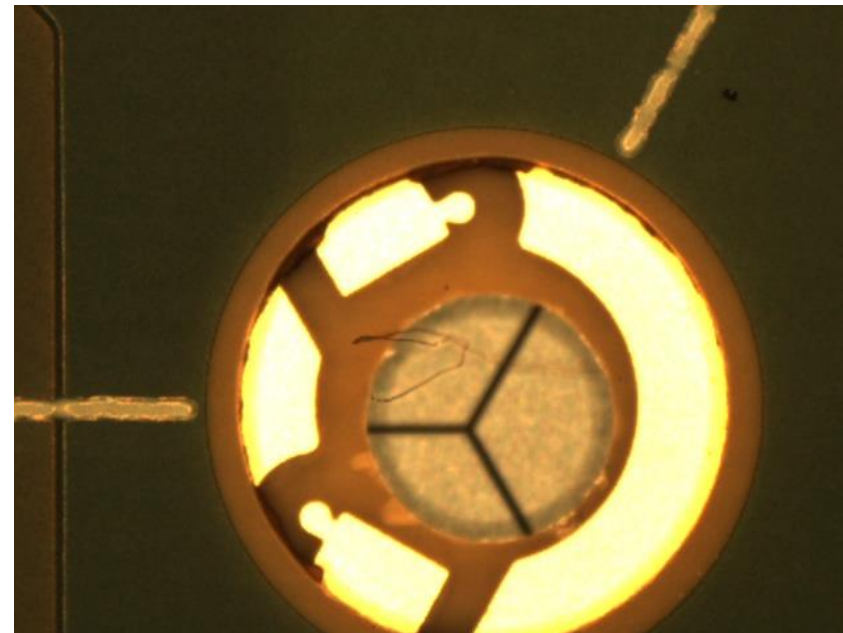
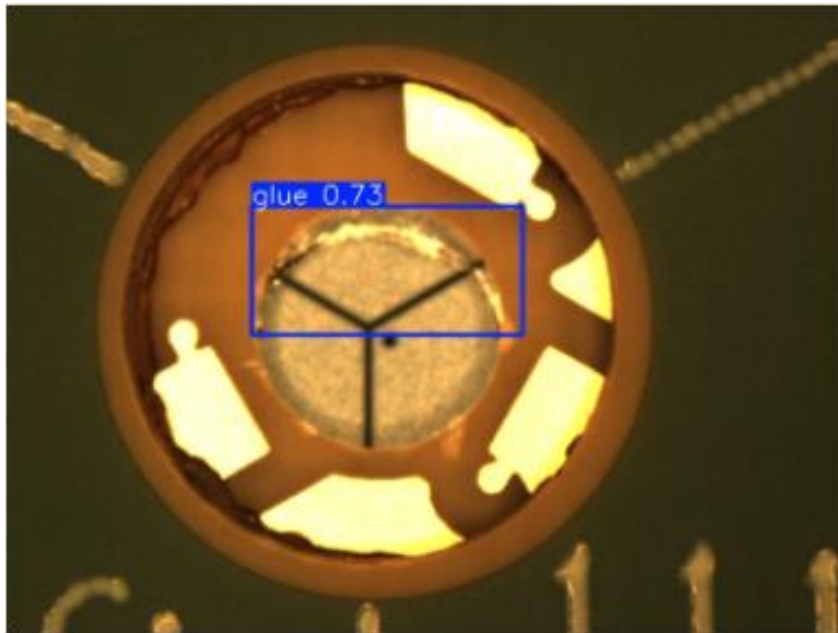
Conf=0.1, IOU=0.5  
Test Set Confusion Matrix



Precision : 0.9474    Recall : 0.7500



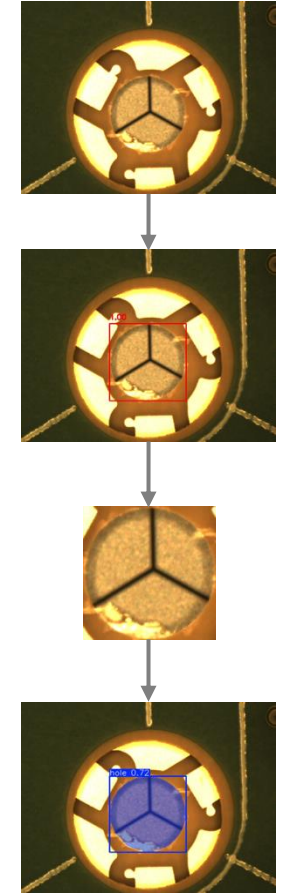
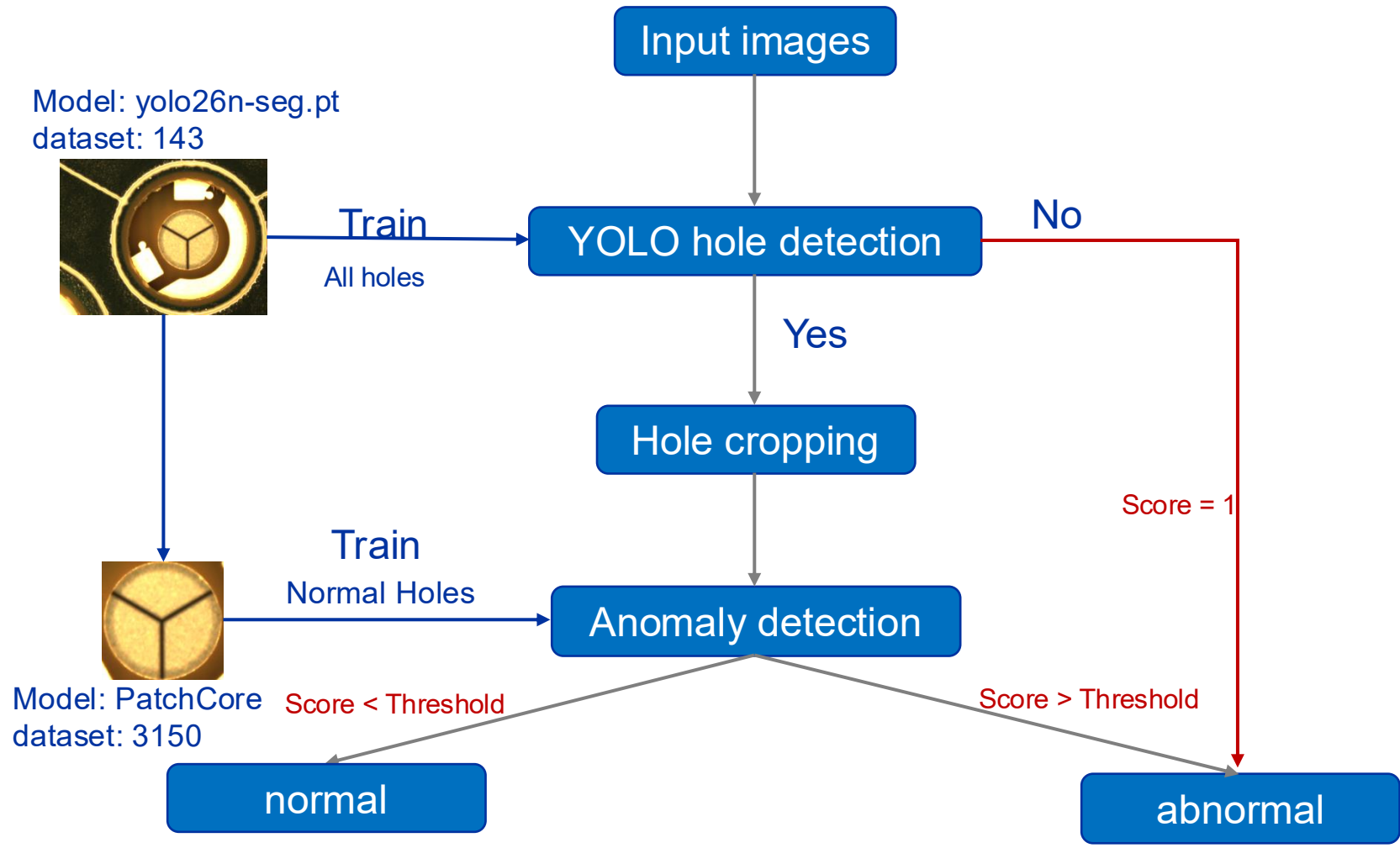
# Before bonding: YOLO Performance



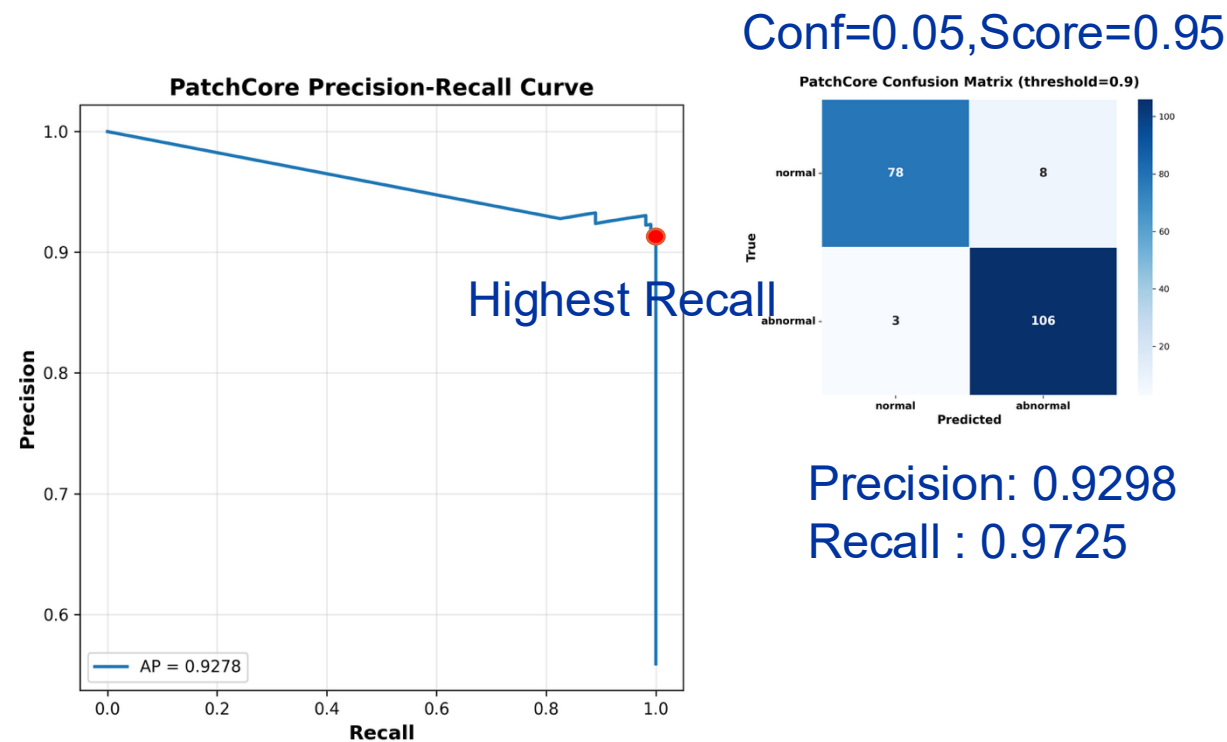
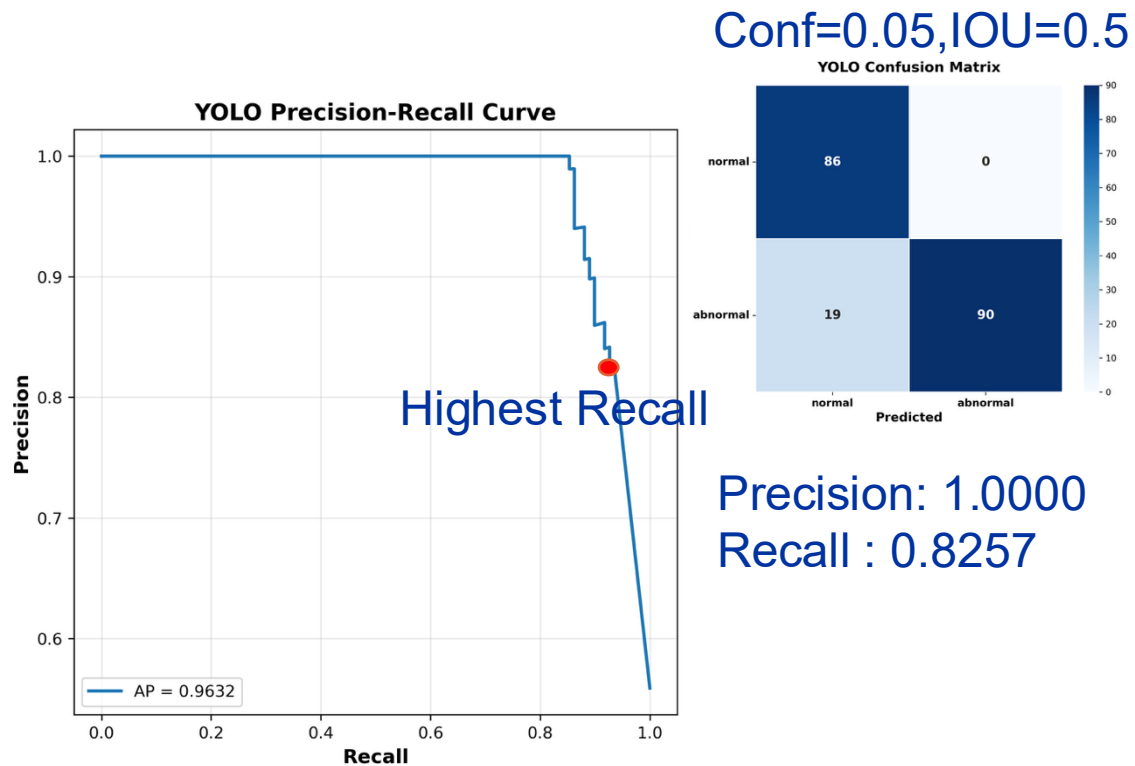
Ignore pollutants that are not in the training labels !

# Before bonding: YOLO-seg+ PatchCore

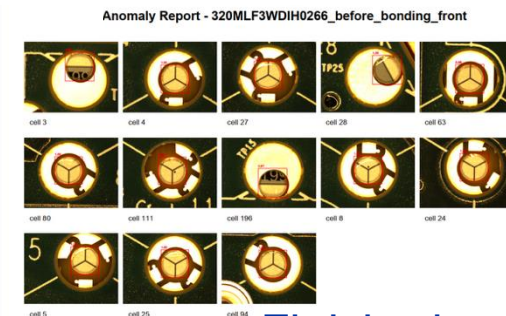
The YOLO model ignores pollutants not in the training set → YOLO + Unsupervised training method.



# Before bonding: YOLO vs. YOLO + PatchCore



- YOLO is cleaner for known glue patterns.
- PatchCore is more sensitive to unseen or subtle pollutants.
- The hybrid workflow is therefore the safer screening choice.

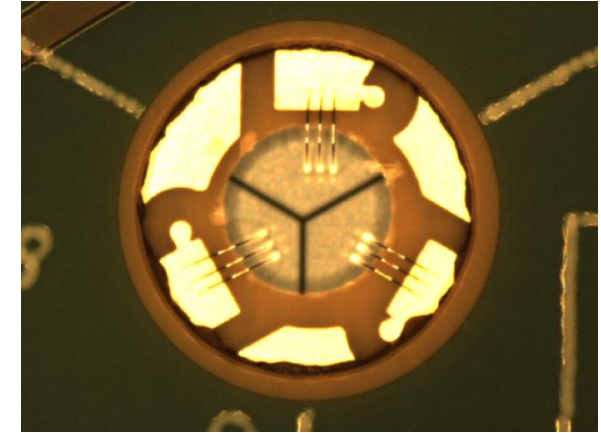


Finished end-to-end pipeline

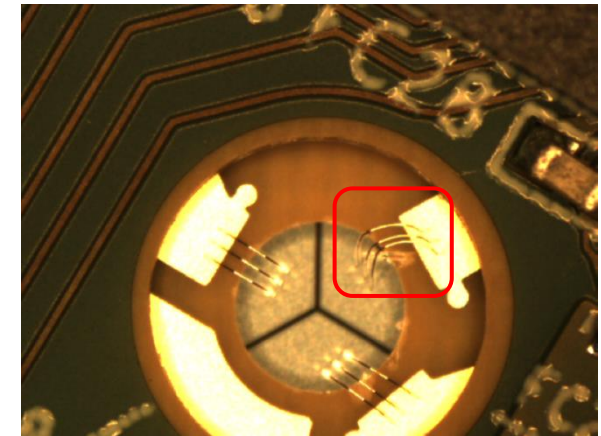
# After bonding: a harder problem

## Key challenges :

- Wires are **thin and occupy very small regions**
- Defects are **subtle and hard to distinguish**
- Need high labeling cost
- Severe **data imbalance** : Only ~100 abnormal in ~20,000 images



normal



Bad wire

# Attempt 1-YOLOv8 training

**Label tool:** labellmg v1.8.6 (YOLO format)

**Class:** 1 (“wire”)

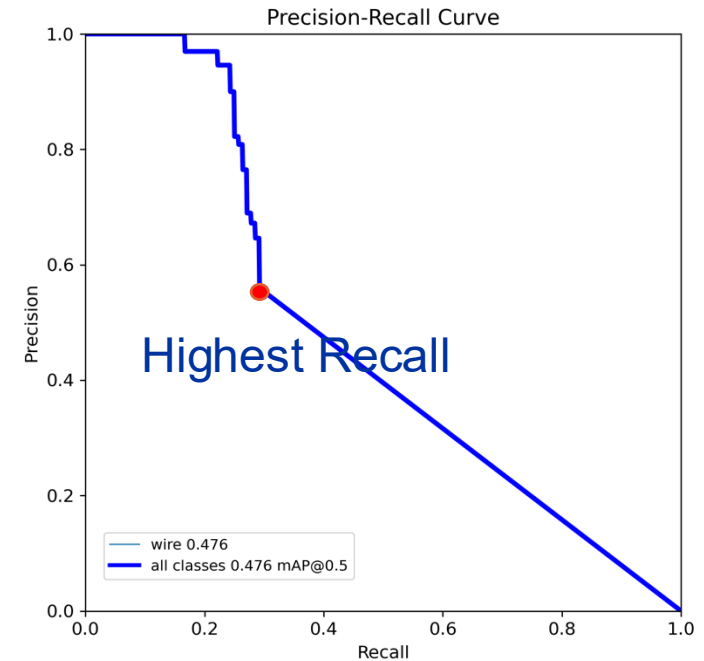
**abnormal:** 67+67\*7 (data augmentation)

**normal:** 1000 (To avoid training bias)

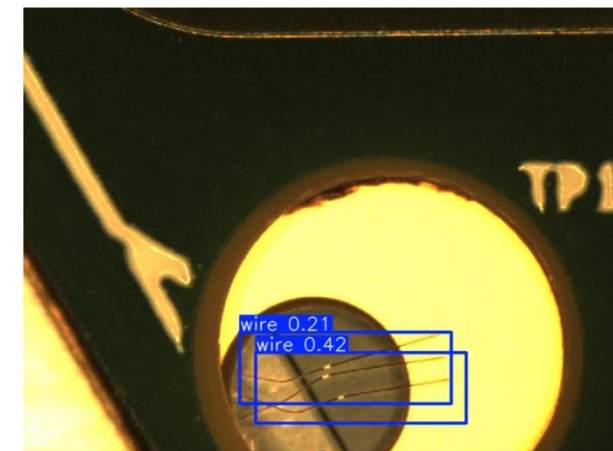
**Image Source:** Taken by OGP smartscope

## Dataset:

| class  | Train | Val | Test |
|--------|-------|-----|------|
| wire   | 424   | 56  | 56   |
| normal | 800   | 100 | 100  |
| total  | 1224  | 156 | 156  |



low recall on bad wires



# Attempt 2-Patchcore training

**Dataset:** 5000 normal images

## Results:

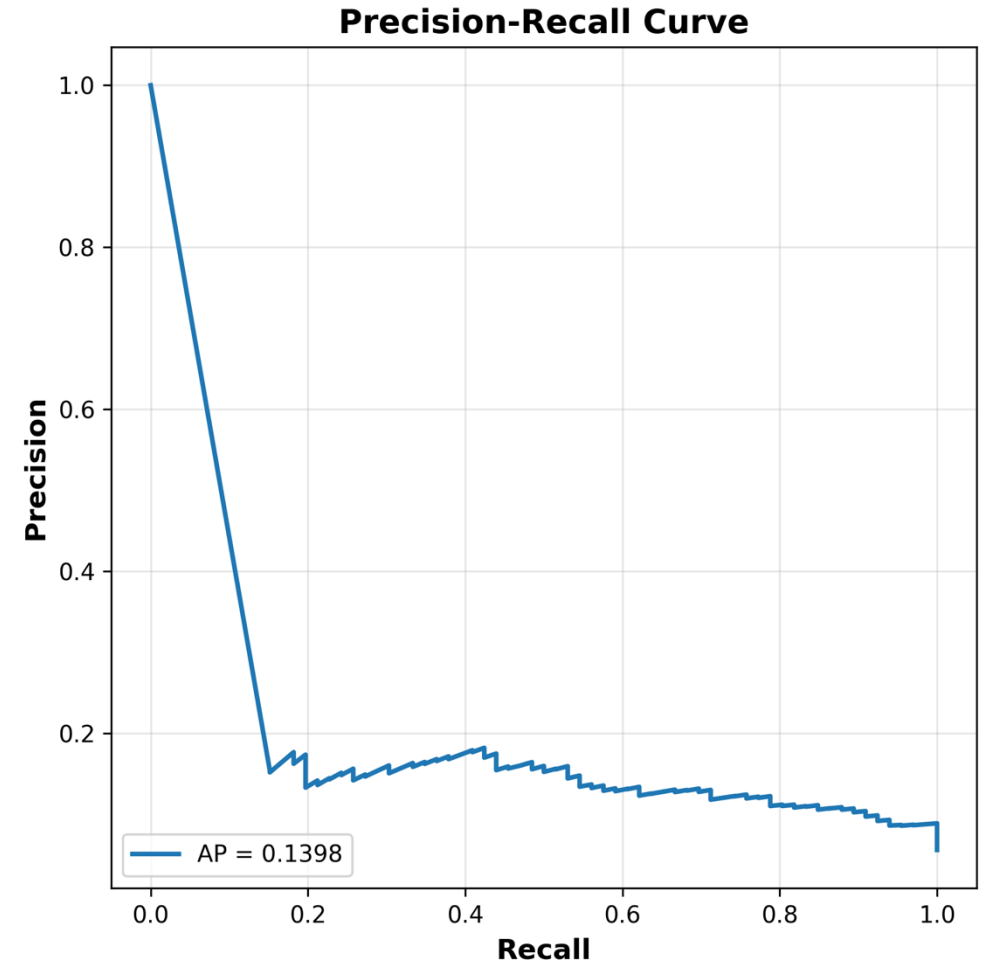
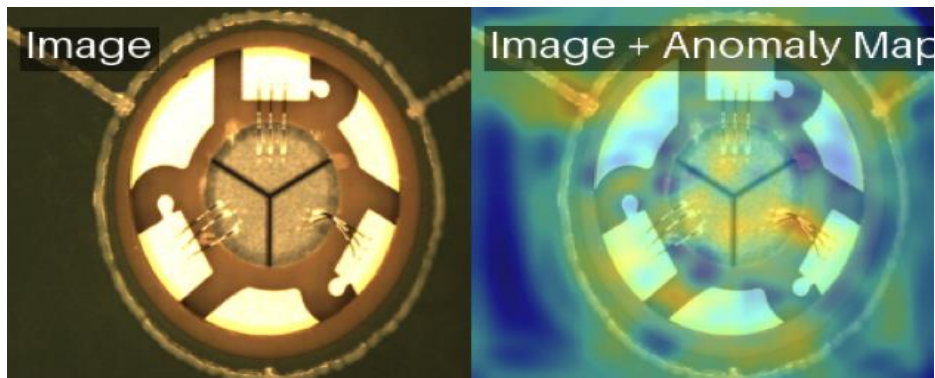
many normal images classified as abnormal

Scores overlap a lot

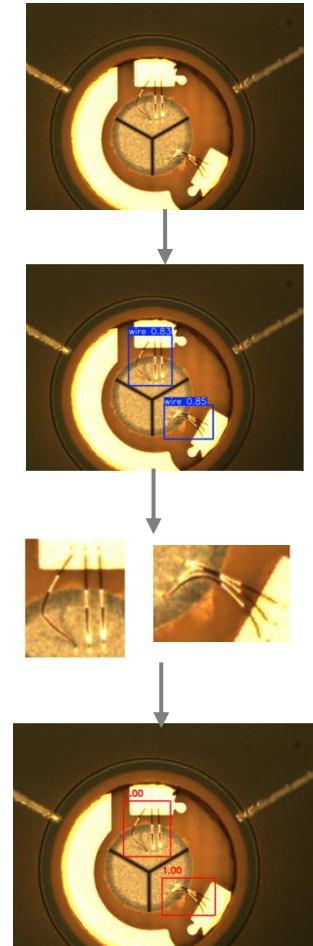
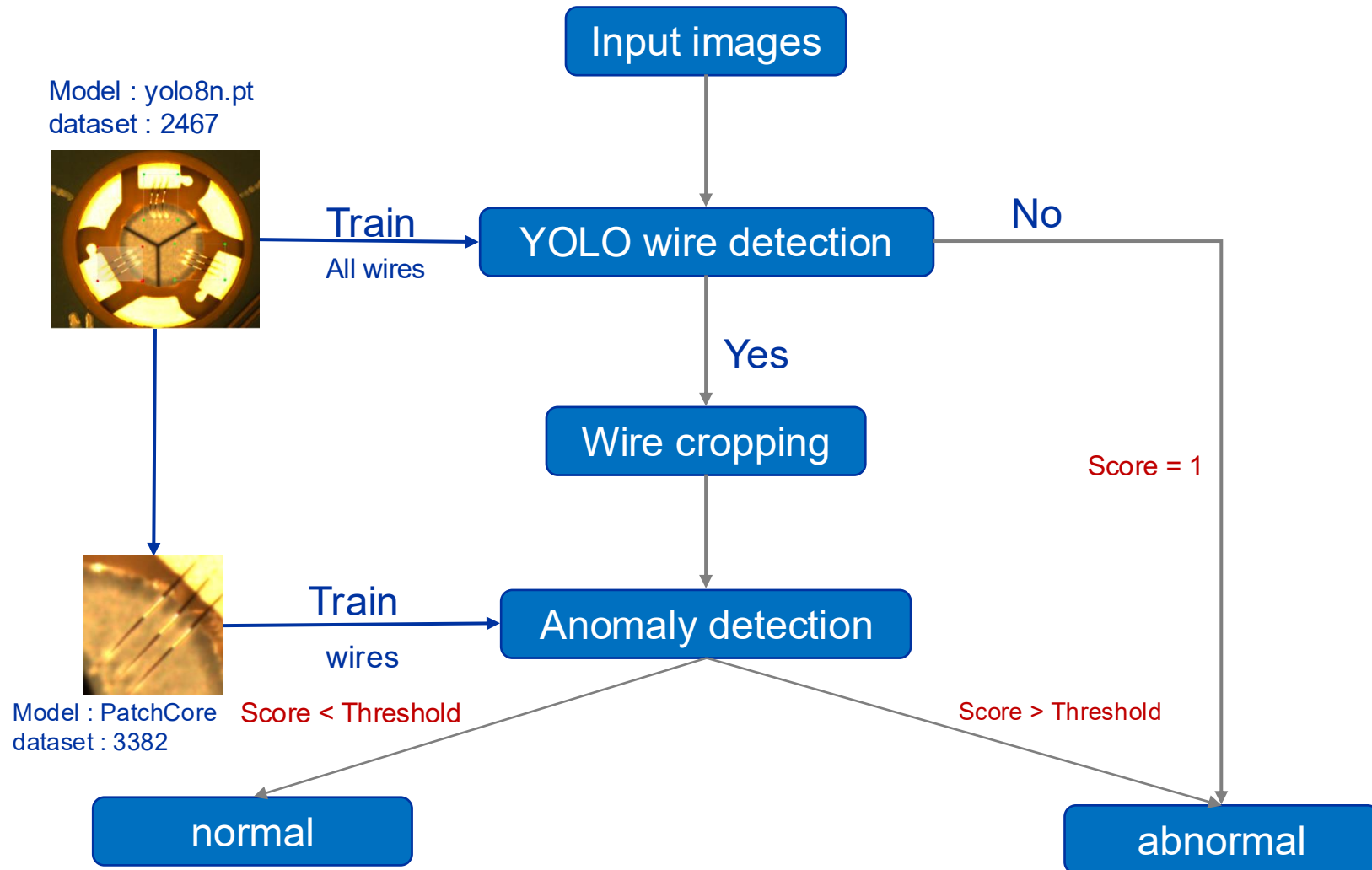
## Analysis:

Background variation  $\gg$  wire feature variation

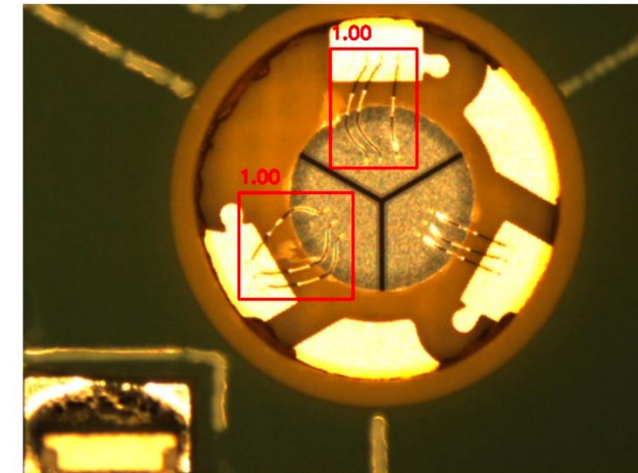
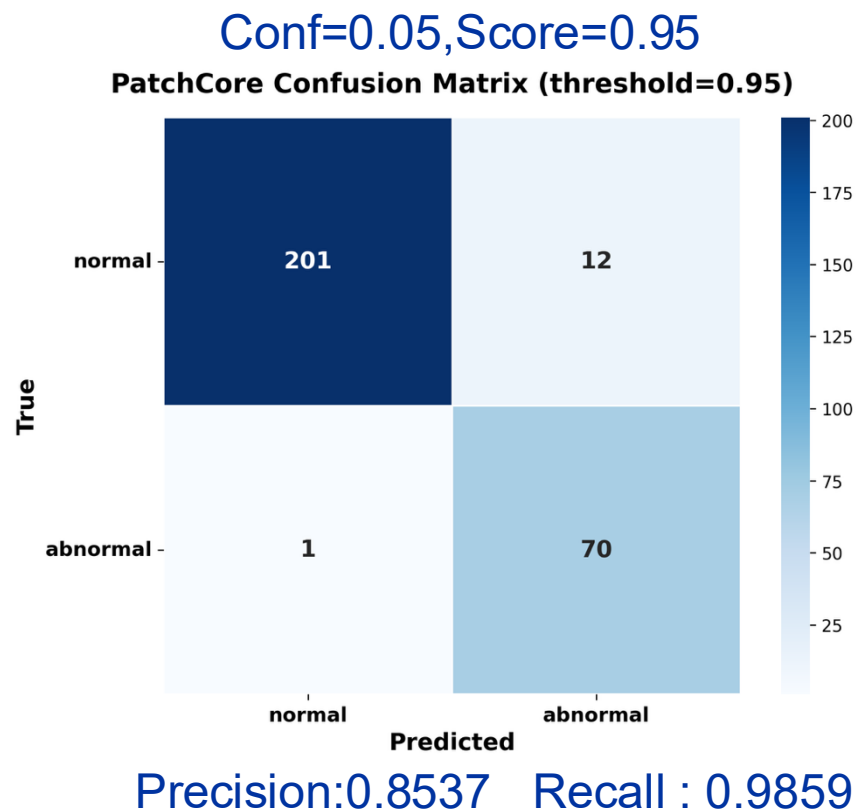
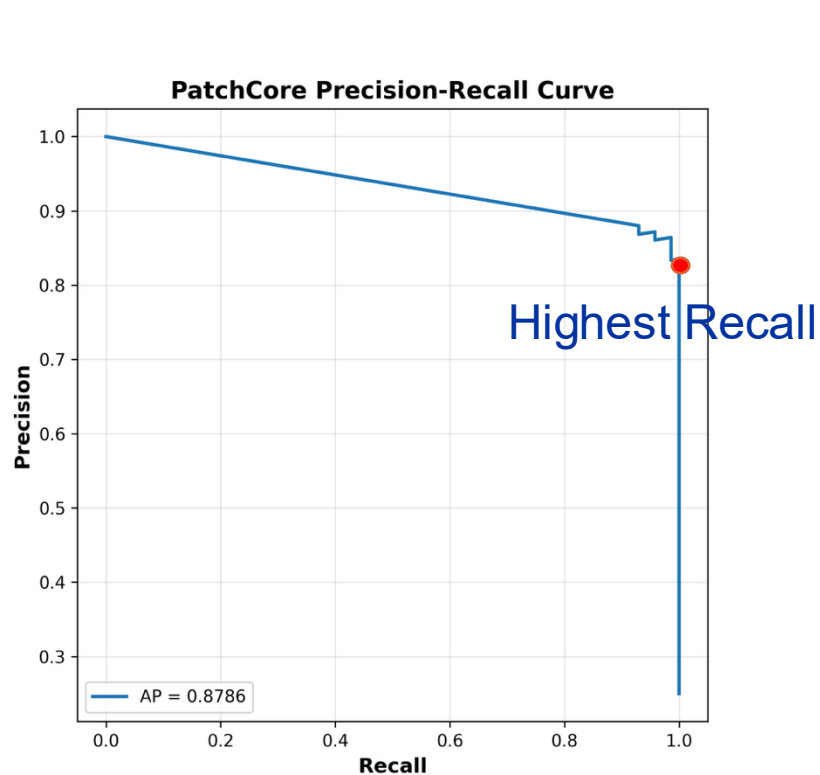
Model learns both background and wire itself



# After bonding: YOLO + PatchCore



# After bonding: current model status



Results are evaluated on the same production batch

# After bonding: current model status



The model becomes nearly ineffective when evaluated on a different batch due to lighting changes  
Not a low recall issue, precision collapses

# Summary

- YOLO + PatchCore combines precise region localization with label-efficient anomaly detection, enabling accurate and scalable quality inspection even with scarce defect data.

- Before bonding:

YOLO + PatchCore model has already been deployed to the production line.

- After bonding :

Performance drops significantly under different batches (lighting variations).

- *Future outlook:*

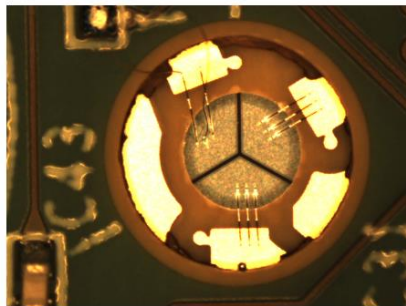
*Improve the after bonding model to handle batch variations.*

*Enable real-time recognition directly on OGP.*

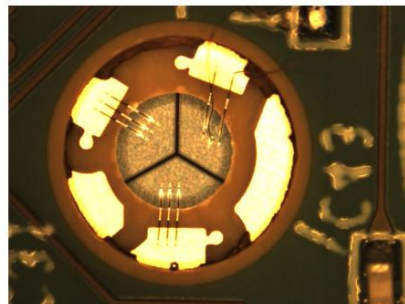


**THANKS**

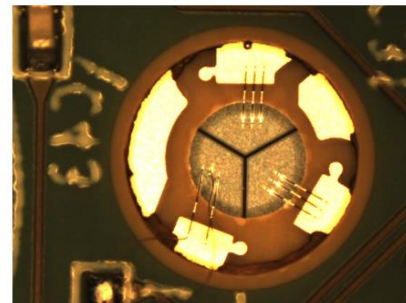
# Augmentation



original



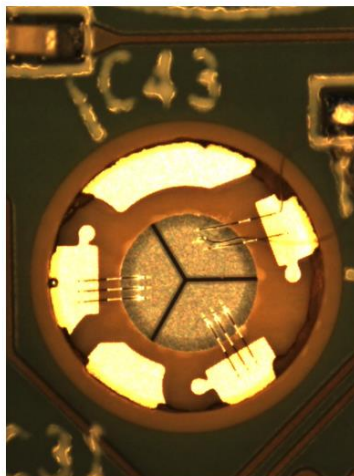
vertical flip



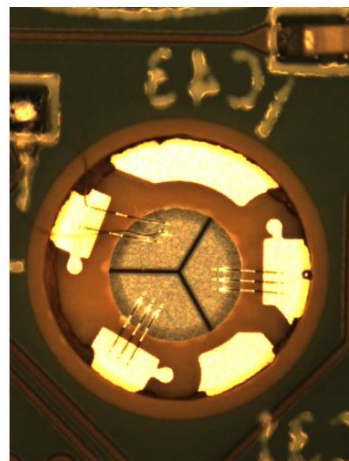
horizontal flip



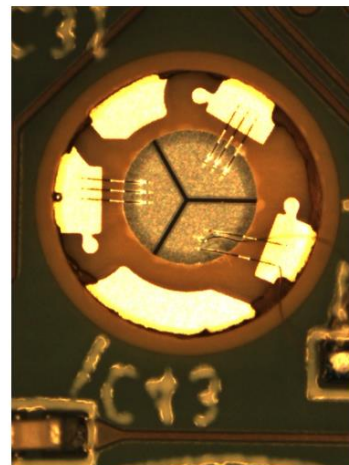
vertical+horizontal



90° rotation



90° +vertical



90° +horizontal



90° +vertical+horizontal