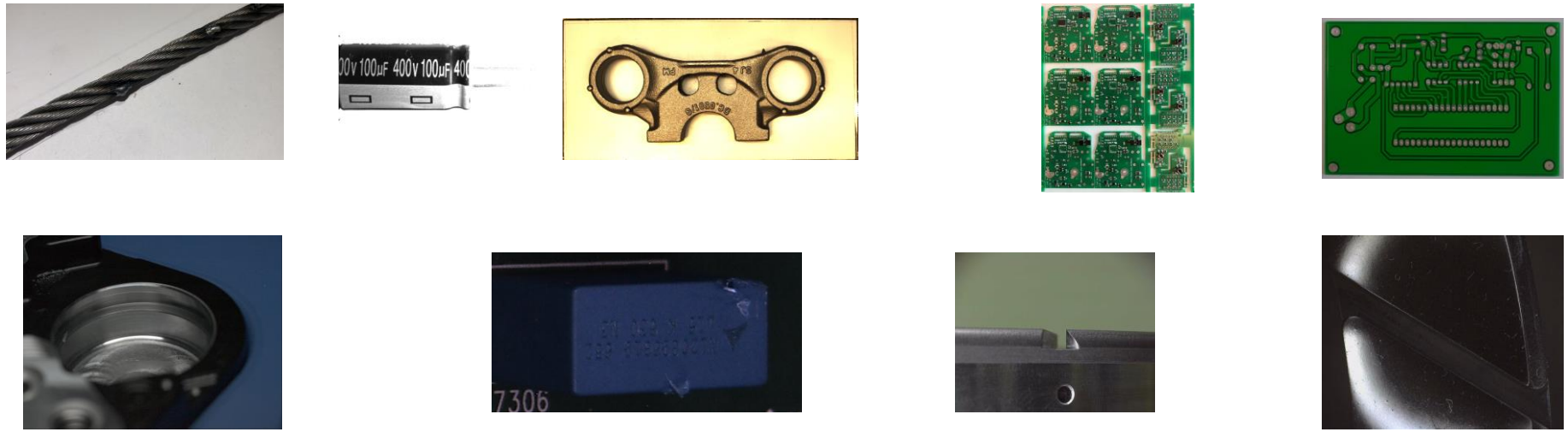
# Leveraging Noisy Boundaries for Semi-Supervised Industrial Defect Detection

Yue Wang[1], Qianyu Zhou[1], Ke Fan[1], Ran Yi[1], Wenbing Zhu[2], Lizhuang Ma[1]

[1]Shanghai Jiao Tong University, Shanghai, China; [2]Rongcheer

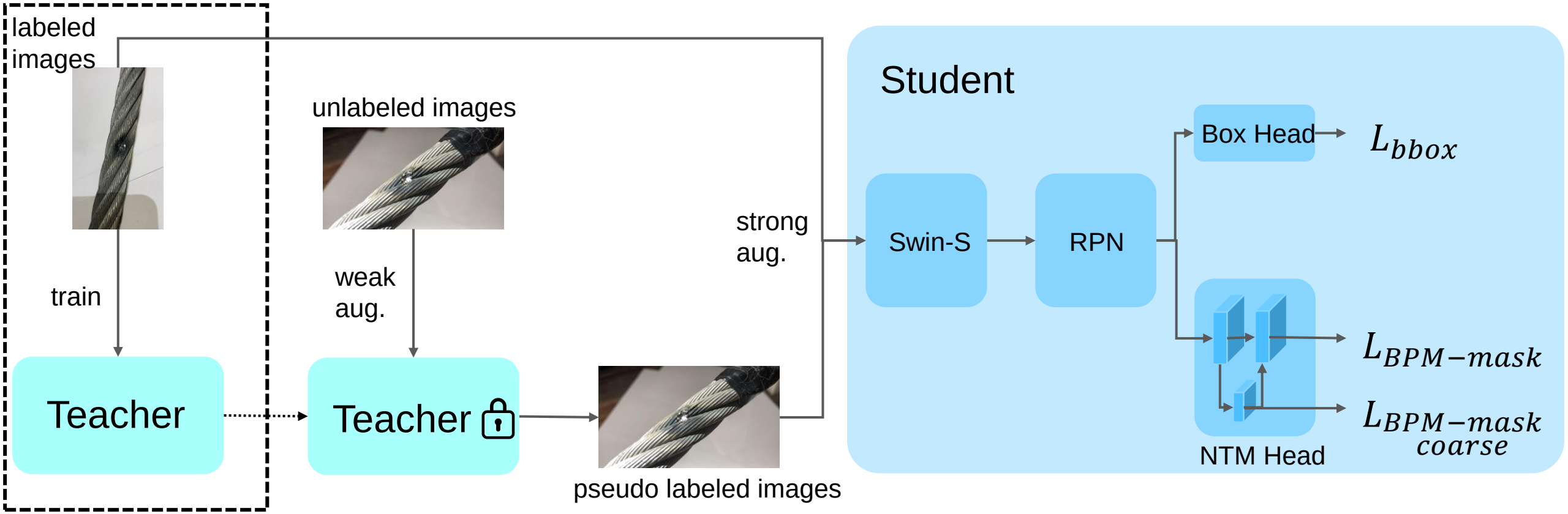
# Overview

- The increasing demand for high-quality and efficient manufacturing processes necessitates advanced defect detection techniques in industrial inspection systems.
- However, current instance segmentation methods rely heavily on pixel-level labeled images, which are expensive and time-consuming to acquire. This constraint restricts the scale of datasets, limiting the performance of these methods.



# Pipeline

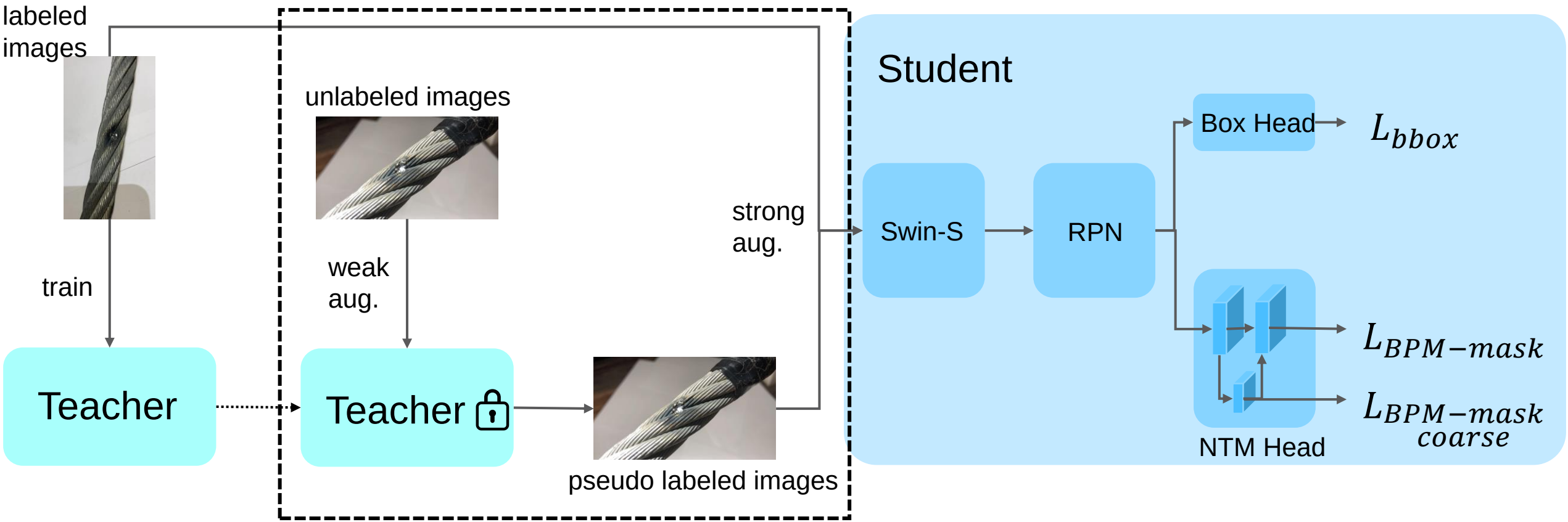
- Our semi-supervised learning pipeline contains 3 steps.
  1. We first train the teacher model in a supervised manner, and the teacher model's performance will further influence the student model's performance.



# Pipeline

- Our semi-supervised learning pipeline contains 3 steps.

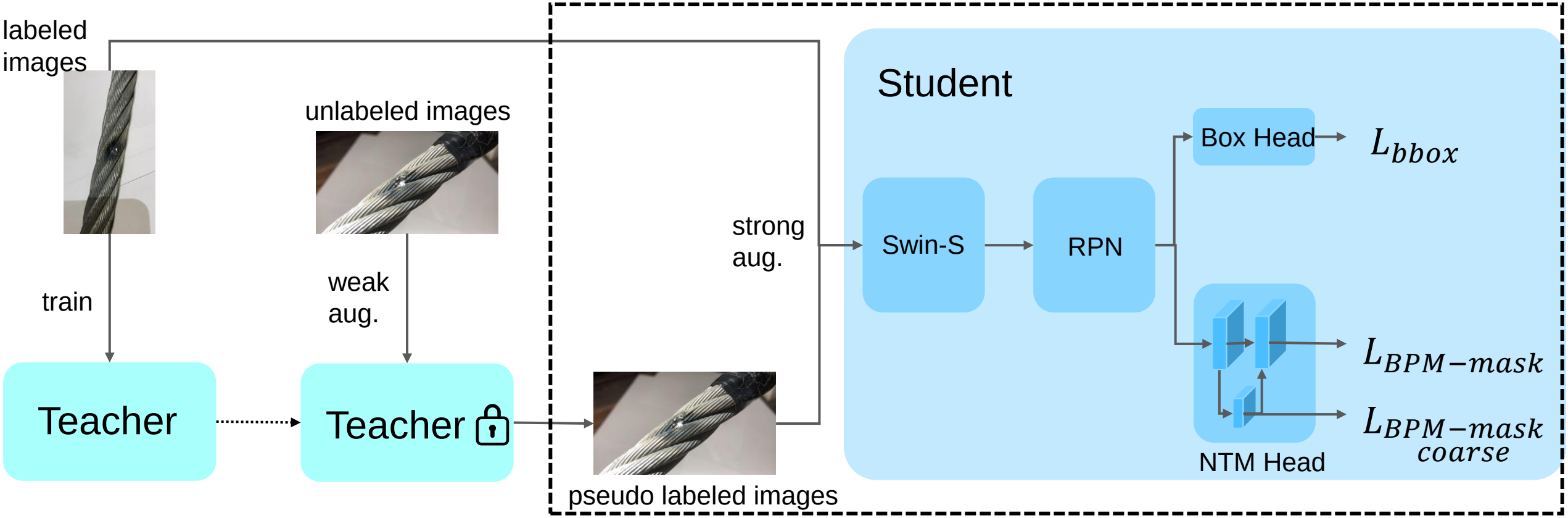
  2. We use the teacher model to generate the pseudo label and filter most noise to get clearer labels.



# Pipeline

- Our semi-supervised learning pipeline contains 3 steps.

  3. We train the student model with the ground truth and pseudo label, NTM and BPM are used to resist the noise in pseudo labels.



# Overview

- **Noise-Tolerant Mask Head (NTM)** uses a low-resolution branch to get a small size and low-resolution feature, which has less noise and cleaner message. After fusing the low-resolution feature to the original branch, the NTM can resist the noise of the pseudo labels
- **Boundary-Preserving Map (BPM) i**s a Laplace operation of the sigmoid function output. In the BPM, the pixel, that does not lie on the boundaries but is close to boundaries, has the highest weight. The above pixel has clean messages and is quite useful for semi-supervised learning

# Architecture Analysis

- The teacher model is quite important for pseudo-label generation.
- We explore different backbone, detector, and pre-train strategies in this section, the result is illustrated below:

| Detector | Backbone | Pretrain | Cable | Casting |
|---|---|---|---|---|
| Mask R-CNN | ResNet-50 | ImageNet1K | 0.014 | 0.050 |
| Mask R-CNN | ResNet-50 | ImageNet1K + MS COCO | 0.211 | 0.099 |
| Mask R-CNN | ResNet-101 | ImageNet1K + MS COCO | 0.211 | 0.096 |
| SOLOv2 | ResNet-50 | ImageNet1K + MS COCO | 0.205 | 0.091 |
| SOLOv2 | ResNet-101 | ImageNet1K + MS COCO | 0.188 | 0.109 |
| Mask R-CNN | Swin-S | ImageNet1K + MS COCO | **0.228** | **0.118** |

# All Categories Results

- We finally test our model on the inference dataset, and the results are illustrated below:

| Method | dataset | Cable | Casting | Cylinder | Groove | Lens | PCB_2 | Screw |
|---|---|---|---|---|---|---|---|---|
| | | Capacitor | Console | Electronic | Hemisphere | PCB_1 | Ring | Wood |
| sup | train | 0.223/0.302 | 0.119/0.220 | 0.455/0.561 | 0.110/0.246 | 0.119/0.196 | 0.707/0.736 | 0.437/0.512 |
| | | 0.244/0.363 | 0.033/0.083 | 0.141/0.280 | 0.204/0.344 | 0.145/0.165 | 0.156/0.219 | 0.248/0.314 |
| semi | train + inference | 0.198/0.348 | 0.076/0.218 | 0.455/0.561 | 0.134/0.311 | 0.183/0.274 | 0.808/0.837 | 0.427/0.510 |
| | | 0.324/0.469 | 0.043/0.121 | 0.178/0.316 | 0.258/0.407 | 0.293/0.373 | 0.117/0.234 | 0.256/0.331 |
| semi + scale | train + inference | 0.340/0.455 | 0.156/0.256 | 0.499/0.614 | 0.178/0.354 | 0.254/0.384 | 0.818/0.838 | 0.427/0.510 |
| | | 0.398/0.545 | 0.091/0.182 | 0.178/0.316 | 0.278/0.440 | 0.427/0.510 | 0.131/0.206 | 0.251/0.350 |
| sup + scale | train + val | 0.333/0.414 | 0.180/0.266 | 0.442/0.533 | 0.197/0.338 | **0.305/0.380** | 0.742/0.769 | 0.451/0.535 |
| | | 0.356/0.498 | 0.090/0.186 | 0.238/0.376 | 0.319/0.460 | 0.590/0.661 | **0.159**/0.226 | 0.273/0.353 |
| semi + scale | train + val + inference | **0.342/0.461** | **0.182/0.287** | **0.532/0.639** | **0.211/0.360** | **0.305/0.380** | **0.840/0.865** | **0.476/0.539** |
| | | **0.434/0.589** | **0.133/0.220** | **0.265/0.396** | **0.319/0.492** | **0.650/0.715** | **0.159/0.230** | **0.307/0.398** |

# All Categories Results

- We finally test our model on the inference dataset, and the results are illustrated below:

| Method | dataset | $ave.mAP$ | $ave.mAR^{maxDets=100}$ | mean |
|---|---|---|---|---|
| sup | train | 0.238 | 0.324 | 0.281 |
| semi | train + inference | 0.268 | 0.379 | 0.323 |
| semi + scale | train + inference | 0.328 | <u>0.437</u> | <u>0.383</u> |
| sup + scale | train + val | 0.334 | 0.428 | 0.381 |
| semi + scale | train + val + inference | **0.368** | **0.469** | **0.419** |

- **we can conclude that:**
  - the semi-supervise strategy has a significant influence on improving the instance segmentation performance.
  - the scale influence is as important as the semi supervised strategy.

# Conclusion

- We addressed the challenge of data-efficient defect detection in an industrial inspection by employing semi-supervised instance segmentation.
- Our method leverages both limited labeled data and abundant unlabeled data to improve the performance of instance segmentation for defect detection.

# THANKS!