SCHOOL OF
ENGINEERING AND
DIGITAL SCIENCES

NAZARBAYEV
UNIVERSITY

Fabric Defect Detection

Muhammad Husnain Rasool

Introduction

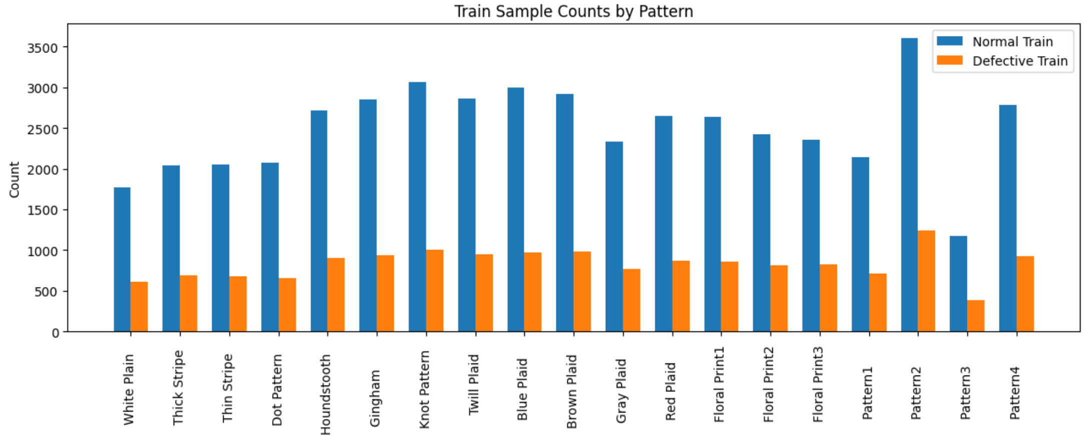
- Issue in Textile Manufacturing:
  - Fabric defects directly affect product quality, leading to significant economic losses.
- Current Quality Control Limitations:
  - Manual inspection is labor-intensive, error-prone, and costly.
- Need for Automation:
  - An automated defect detection system can reduce human error and cut the cost.
  - The system must be adaptable and capable of rapid deployment to detect a wide range of defects.

Literature Review

Model Category	Techniques / Examples	Strengths	Limitations
Traditional Statistical/Structural	• Histogram statistics • Co-occurrence matrices • Mathematical morphology	• Straightforward and computationally efficient • Effective for regular, simple fabric patterns	• Relies on hand-crafted features • Poor performance on complex or irregular textures
Deep Learning (Supervised)	• Convolutional Neural Networks (e.g., VGG, U-Net)	• End-to-end learning of complex features • High accuracy and precise localization through segmentation (e.g., U-Net's skip connections)	• Requires large, annotated datasets • Computationally intensive
Unsupervised / One-Class Learning	• Convolutional Auto-Encoders (CAE) • One-Class SVM	• Trains only on normal (defect-free) samples • Detects anomalies based on reconstruction errors or deviation from learned normal patterns	• May miss subtle defects • Performance depends on the quality of the learned representation

Dataset: ZJU-Leaper

- **Scale & Diversity:** Nearly 100,000 high-resolution images across 19 fabric types.
- **Detailed Annotations**: The dataset provides detailed annotations for the images.
- **Task Settings**: The dataset includes 5 different task settings, progressing from only normal samples to many defective samples with precise annotations.
- **Real-World Relevance:** Designed to mimic the challenges of actual textile production environments, making it ideal for developing adaptable, efficient defect detection systems.



Proposed Approaches

- **Convolutional Autoencoder (CAE) Anomaly Detection:**
  - An unsupervised reconstruction model trained on defect-free samples to highlight deviations as potential defects, enabling detection with no defect labels.
- **UNet-Based Segmentation:**
  - A fully supervised, pixel-wise segmentation model that produces fine-grained defect masks, ideal when precise annotations are available.
- **YOLOv8 Object Detection:**
  - A fast, single-stage detector that outputs bounding boxes and confidence scores for defects, balancing annotation effort and inference speed.

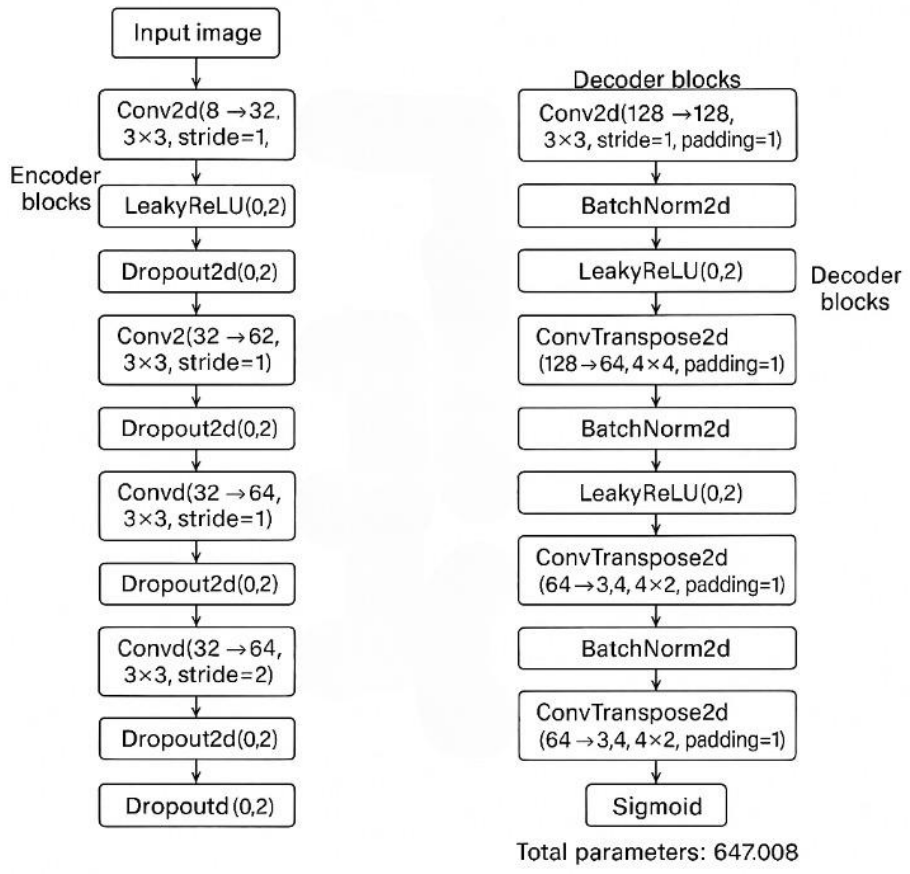
Convolutional Autoencoder (CAE)

- **Goal:** Detect novel defects using only normal samples
- **Architecture:** Deep encoder & symmetric decoder

Training:

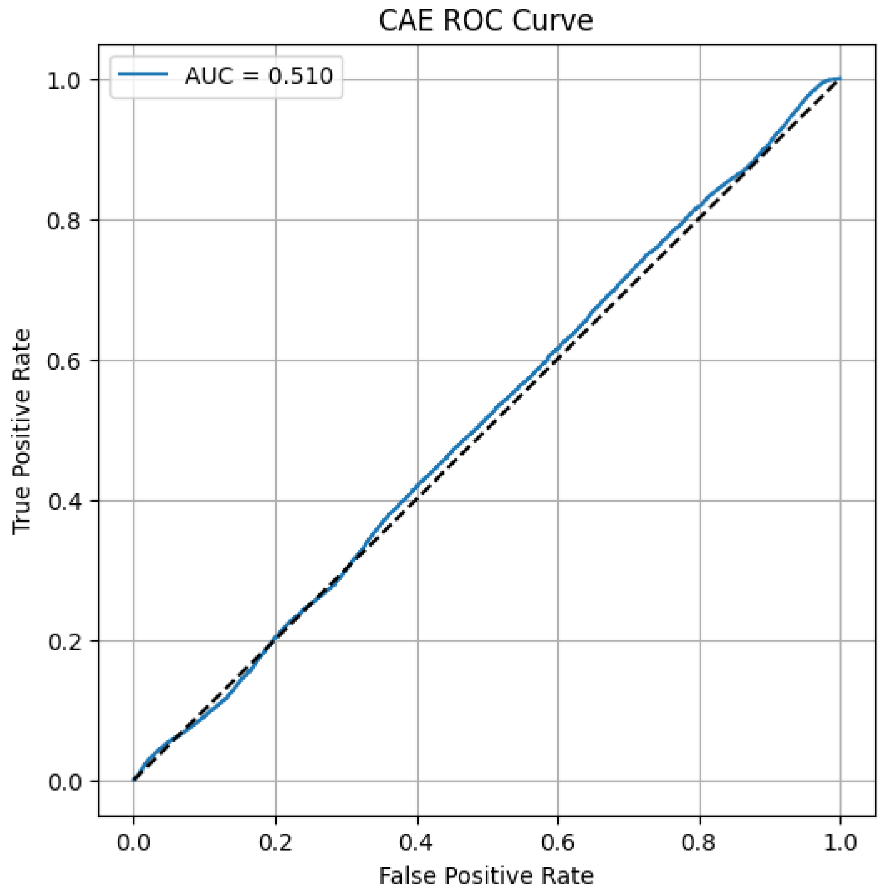
- Loss = MSE reconstruction + 0.5 × (1 – SSIM)
- Optimizer: Adam + cosine-annealing LR, 30 epochs

- **Detection:** Reconstruction error heatmap → defect regions



CAE-Results

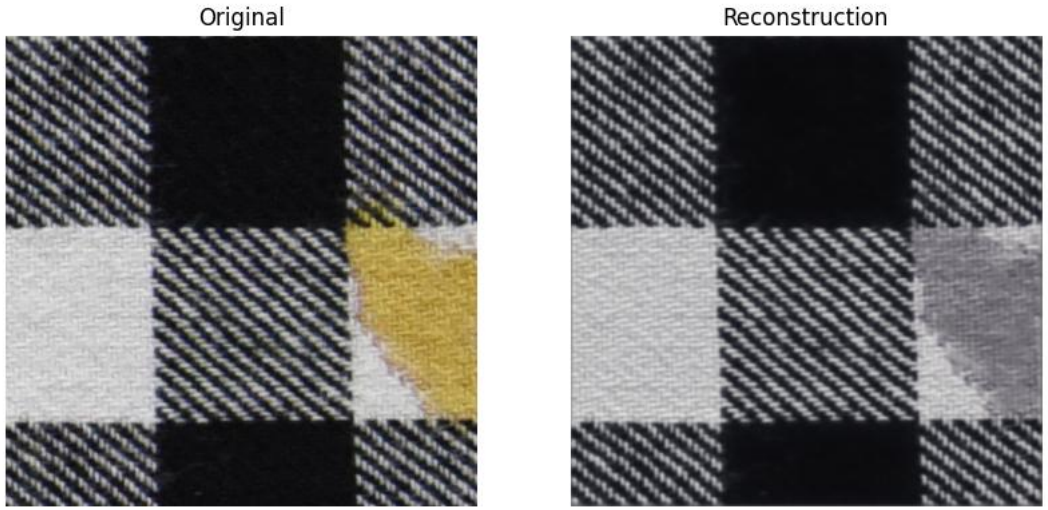
- Threshold (70th percentile of train): 0.003481
- ✅ Accuracy : 0.5985
- ✅ Precision: 0.2494
- ✅ Recall   : 0.2983
- ✅ F1-Score : 0.2717
- ✅ ROC AUC : 0.5097

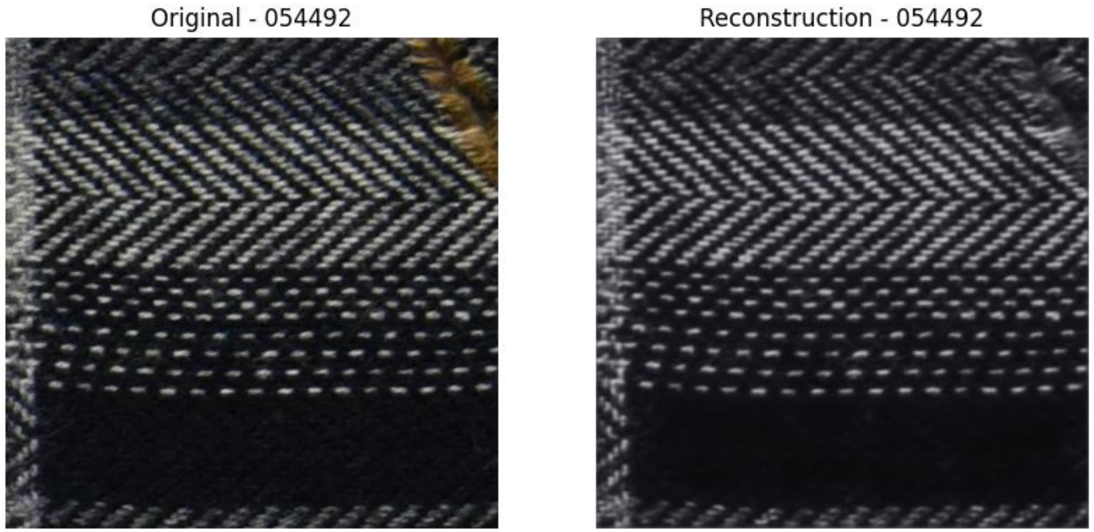


CAE-Results

- Reconstruction MSE: 0.004261
- ❌ Classified as DEFECTIVE

- Reconstruction MSE: 0.001280
- ✅ Classified as NORMAL





UNet-Based Segmentation

- **Goal:** Precise pixel-level defect masks
- **Architecture:** 4-stage encoder–decoder with skip connections

Training:

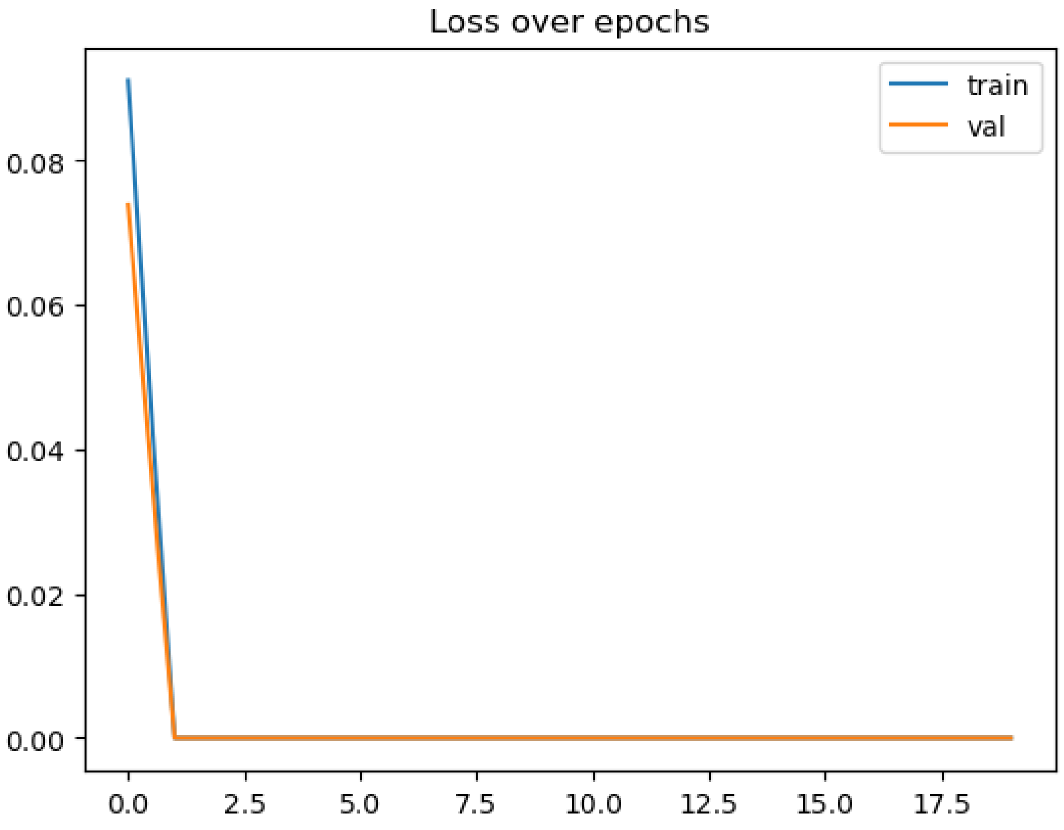
  - Input: 512×512
  - Loss: Binary Cross-Entropy on mask annotations
  - Optimizer: Adam, 20 epochs on TPU

- **Output:** High-resolution mask highlighting defect shape

UNet-Results

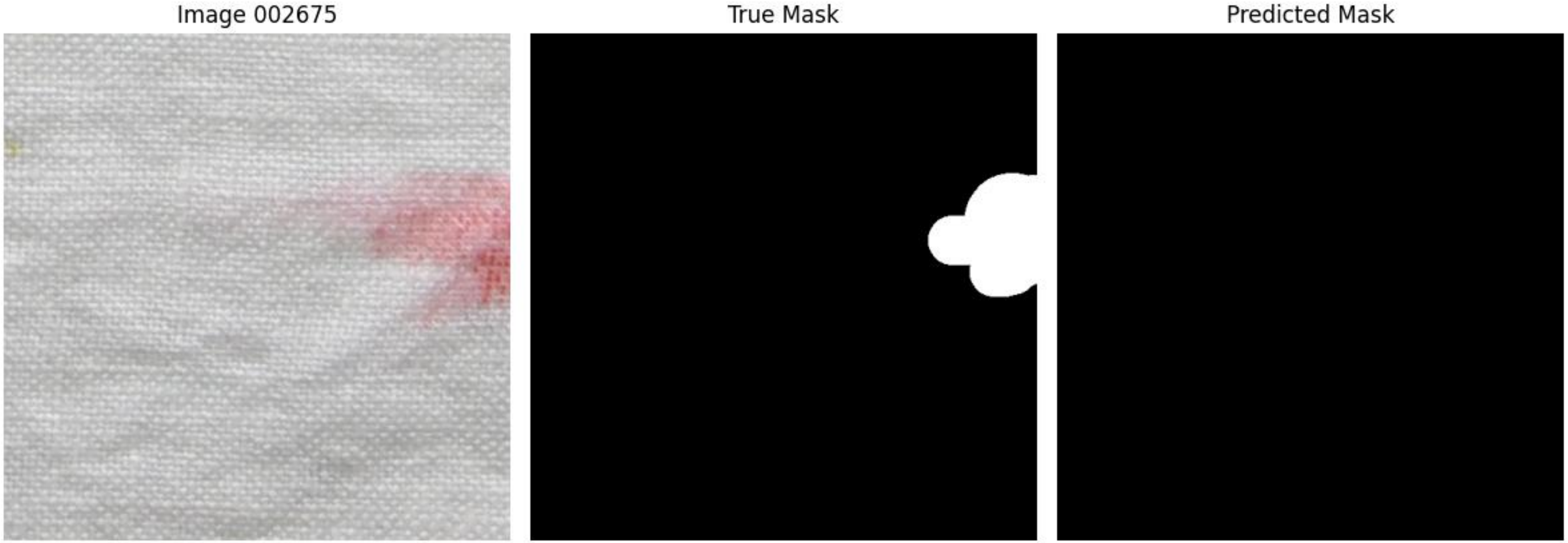
- Architecture

```
UNet(
  (d1): Sequential(
    (0): Conv2d(3, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU(inplace=True)
  )
  (p1): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (d2): Sequential(
    (0): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU(inplace=True)
  )
  (p2): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  (b): Sequential(
    (0): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU(inplace=True)
  )
  (u1): Upsample(scale_factor=2.0, mode='nearest')
  (c3): Sequential(
    (0): Conv2d(192, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU(inplace=True)
  )
  (u2): Upsample(scale_factor=2.0, mode='nearest')
  (c4): Sequential(
    (0): Conv2d(96, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU(inplace=True)
  )
  (out): Conv2d(32, 1, kernel_size=(1, 1), stride=(1, 1))
)
Params: 231617
```



UNet-Results

- Validation IoU: 0.1793
- Validation Dice: 0.3040



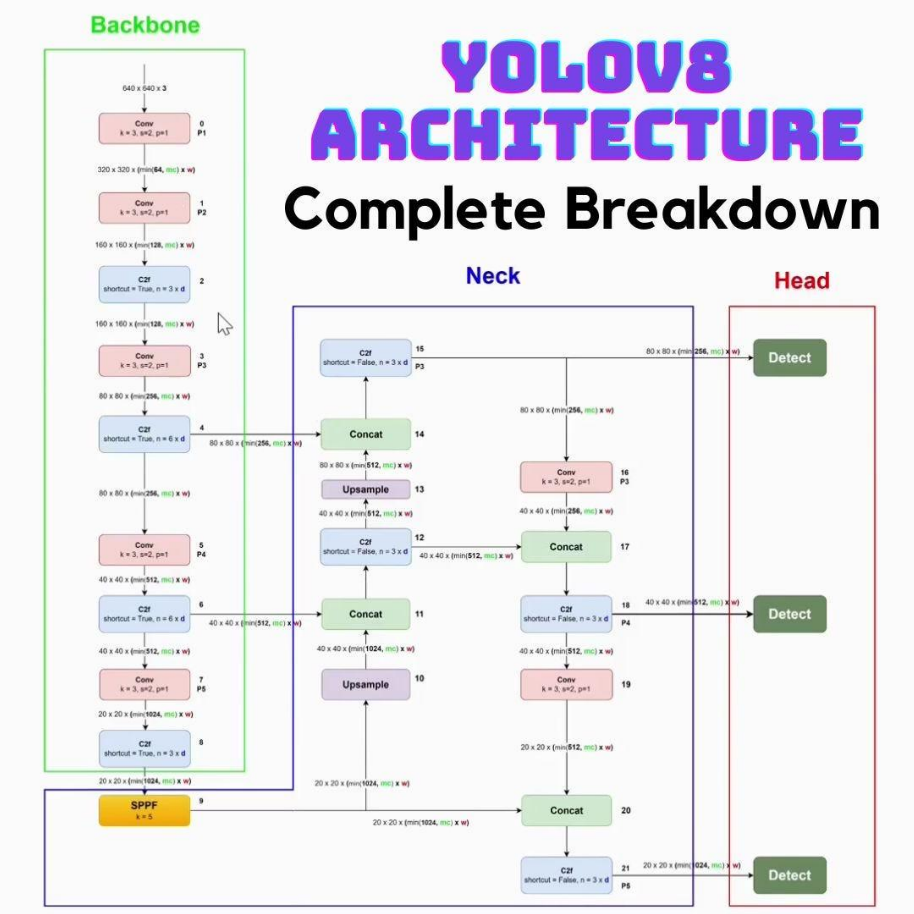
YOLO

- **Goal:** Fast, coarse localization via bounding boxes
- **Model:** yolov8s backbone, single-stage detector

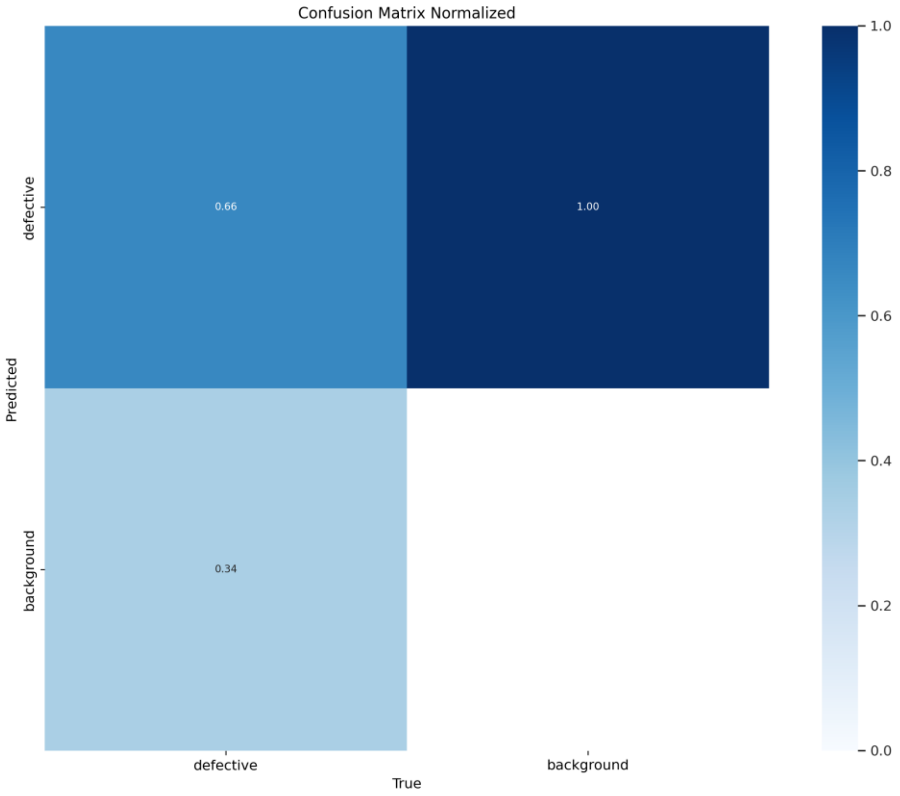
Training:

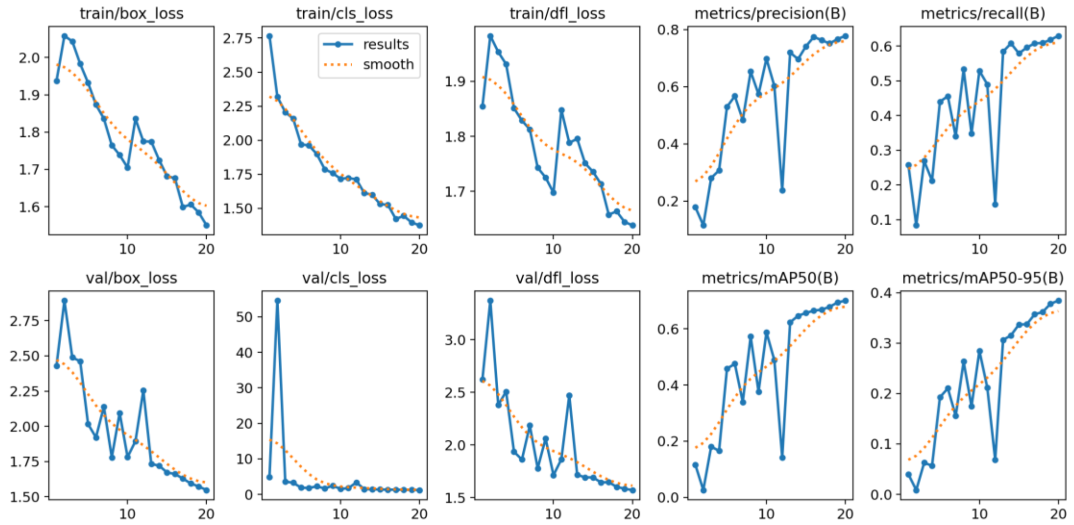
- Data: BBox annotations only
- Epochs: 20, batch=16, img size=640×640
- Metrics: Precision, Recall, mAP@50, mAP@50–95

- **Output:** Defect BBoxes + confidence scores

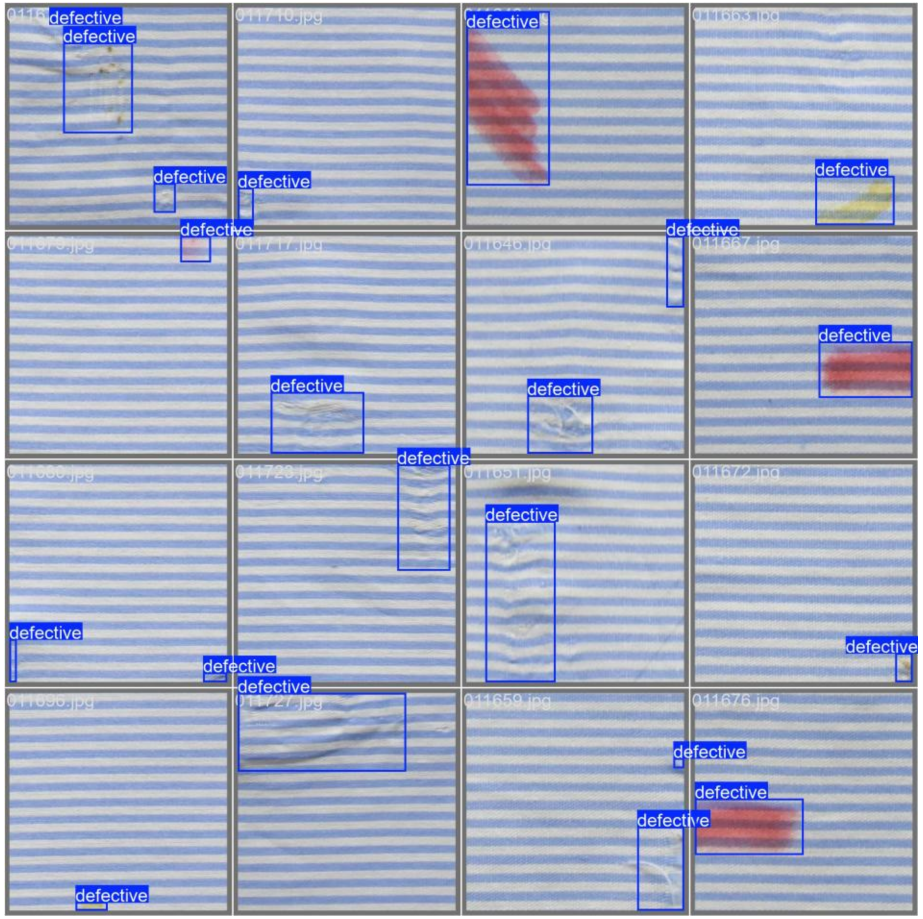


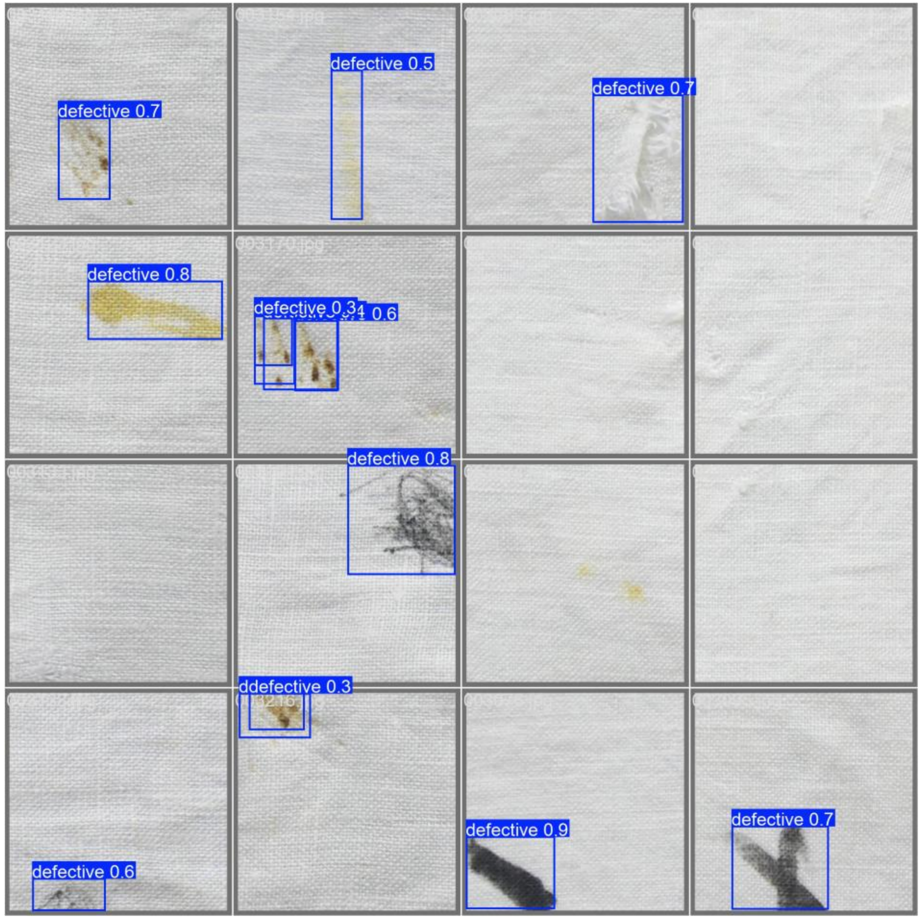
YOLO-Results





YOLO-Results





Conclusion & Future Directions

Method	Data Used	Key Strength	Main Weakness
UNet	Mask annotations	Precise pixel masks	Label-heavy, slow
YOLO	BBox annotations	Fast detection	Coarse localization
CAE	Normal-only	No defect labels	False positives on novel textures

- **Multi-Model Cascade:** Combine CAE → YOLO → UNet for hierarchical screening and mask refinement.
- **Real-Time Deployment:** Optimize and benchmark on live production lines with edge-device acceleration.

References

- C. Zhang, S. Feng, X. Wang and Y. Wang, "ZJU-Leaper: A Benchmark Dataset for Fabric Defect Detection and a Comparative Study," in IEEE Transactions on Artificial Intelligence, vol. 1, no. 3, pp. 219-232, Dec. 2020.
- D. Chetverikov and A. Hanbury, “Finding defects in texture using regularity and local Orientation,”Pattern Recognit., vol. 35, no. 10, pp. 2165–2180, 2002.
- F. Timm and E. Barth, “Non-parametric texture defect detection using weibull features,” in Proc. Int. Soc. Opt. Eng., vol. 7877, 2011, pp. 150–161.
- H. Y. Ngan and G. K. Pang, “Regularity analysis for patterned texture inspection,” IEEE Trans. Autom. Sci. Eng., vol. 6, no. 1, pp. 131–144, Jan. 2009.
- K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. Int. Conf. Learn. Representations (ICLR), 2015.
- R. Varghese and M. Sambath, “YOLOv8: A Novel Object Detection Algorithm with Enhanced Performance and Robustness,” in Proc. 2024 Int. Conf. Advances in Data Engineering and Intelligent Computing Systems (ADICS), 2024, pp. 1–6.