

# | Autonomous Damage Recognition in Ultrasonic Non-Destructive Evaluation Using a Semi-Supervised GAN |

| **JonPaul Hooks** | Graduate Student

| **CSCE 768 – Pattern Recognition and Classification** | Final Project

Advisors - **Dr. Jianjun Hu** | **Dr. Sourav Banerjee**



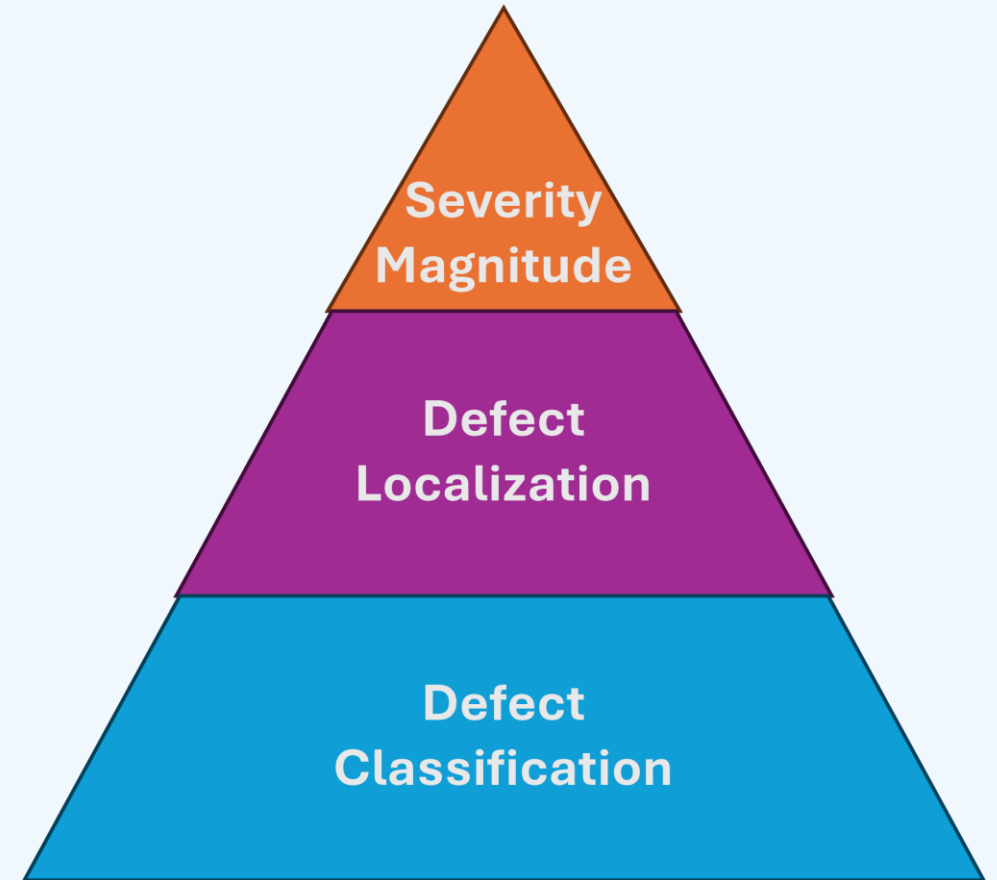
**Molinaroli College of  
Engineering and Computing**  
UNIVERSITY OF SOUTH CAROLINA

**Integrated Material  
Assessment & Predictive  
Simulation Laboratory**

**$i\sqrt{M\alpha P_s}$**

# Problem Statement

- Ultrasonic C-scan inspection is widely used to detect subsurface defects in composite materials
  - Manual inspection inconsistent, time-consuming, and difficult for subtle or noisy defects
- Supervised ML methods require large labeled datasets—scarce in NDE
- **Goal:** Develop a *semi-supervised* model that:
  - Detects presence of defects
  - Localizes the spatially
  - Quantifies severity
- Approach based on SGAN reconstruction + classification



# Technical Challenges

- Highly variable scan textures and noise patterns
- Very small labeled dataset; labels only indicate defect/no-defect (no segmentation)
- Need for interpretable defect localization
- GAN training instability (balancing generator & discriminator)
- Ultrasonic data requires preservation of fine-scale texture

filename	label
100MHz_2.JPG	1
24ply16plyscan10Mhz-001.JPG	1
24ply16plyscan10Mhz-002.JPG	0
24ply16plyscan10Mhz-012.JPG	1
500x_Z48323_60x60_Inside.JPG	0

Binary defect/no-defect label csv for training



# Related Work

## 1. Prajapati et al. (2025) - *Semi-supervised GAN for ultrasonic defect classification*

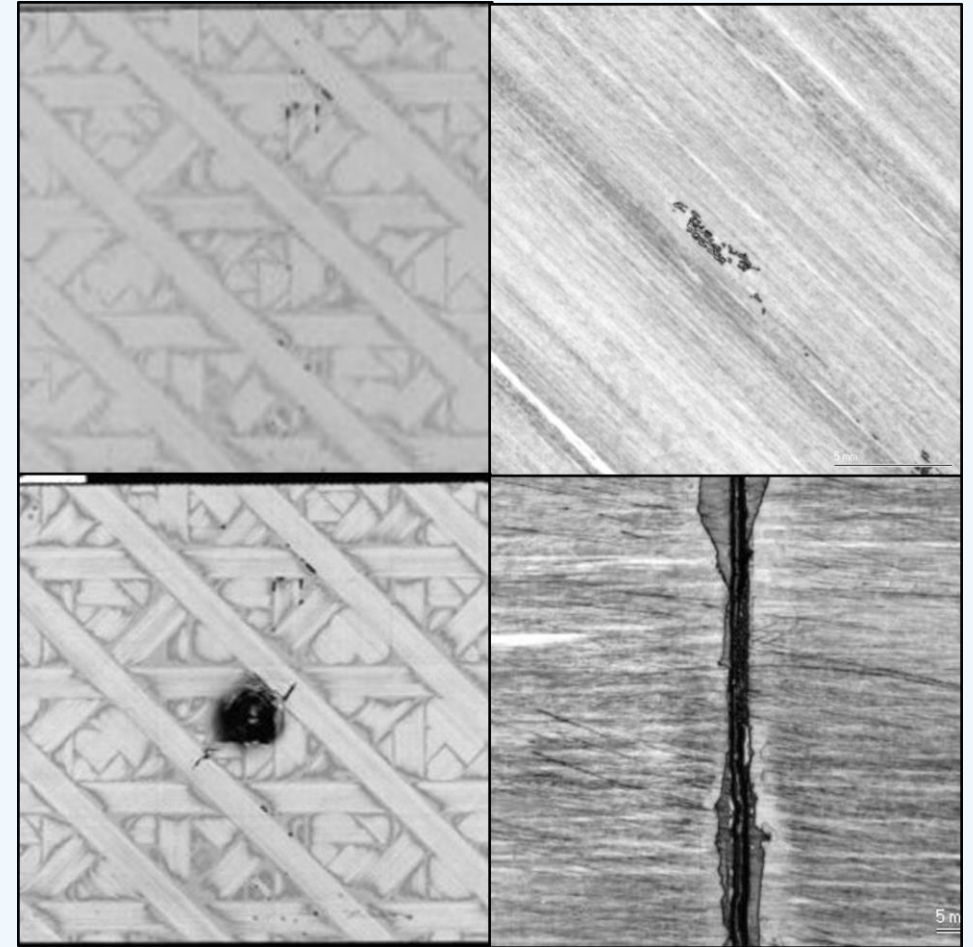
- Related: Shows SGANs work well for low-labeled NDE data
- Different: Their generator doesn't reconstruct inputs; my SGAN produces **heatmaps + severity maps** for interpretable defect localization

## 2. Schlegl et al. (2017) - *GAN reconstruction for anomaly detection*

- Related: Introduces the idea of using **reconstruction error** as a localization tool
- Different: Their method is unsupervised and non-NDE; mine incorporates **supervision**, ultrasonic-specific preprocessing, and **quantitative severity scoring**

# Dataset

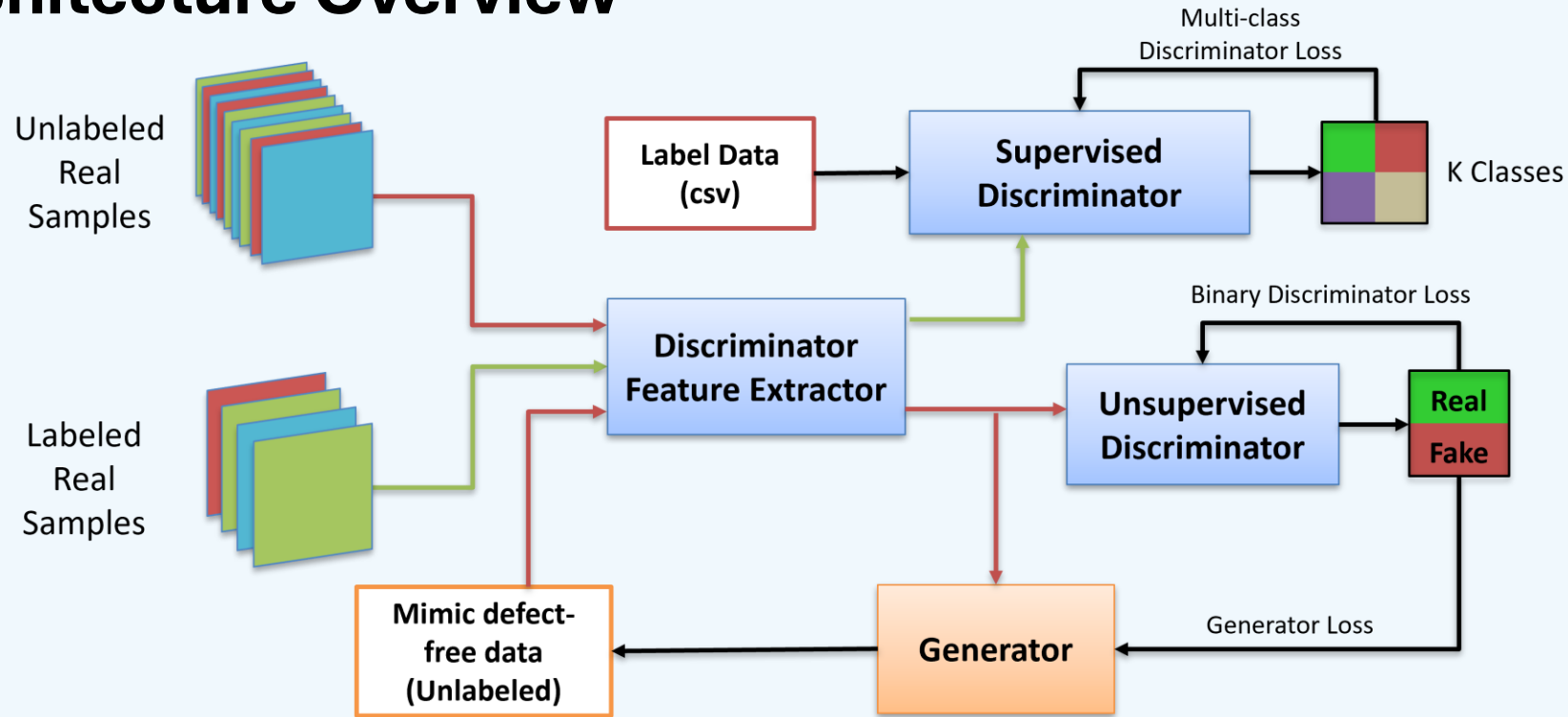
- Ultrasonic C-scan images of CFRP composite laminates
- Each file labeled as 0 = healthy, 1 = defective
- Preprocessing (from *data.py*):
  - Grayscale  $\rightarrow$  256 $\times$ 256
  - Random rotation ( $\pm 5^\circ$ ), horizontal flip
  - Normalize to [-1, 1]
  - Tensor conversion



C-scan examples illustrating structural variations and typical defect modes present in the dataset



# SGAN Architecture Overview



## Generator (UNet-based)

- Learns to reconstruct defect-free version of input
- Skip connections preserve spatial detail
- Produces output used to compute spatial residual heatmaps

## Discriminator

- Dual-headed:
  - Real/Fake discrimination
  - Defect/No-defect classification
- Enables semi-supervised learning

# Training Loss Functions

## Discriminator Loss

- Real/Fake BCE
- Supervised CE for defect classification
- Fake recognition for reconstructions

$$\mathcal{L}_{D,\text{real}} = \text{BCE}(D_{\text{adv}}(x), 0.92)$$

$$\mathcal{L}_{D,\text{cls}} = \text{CE}(D_{\text{cls}}(x), y)$$

$$\mathcal{L}_{D,\text{fake}} = \text{BCE}(D_{\text{adv}}(G(x)), 0)$$

$$\mathcal{L}_D = \mathcal{L}_{D,\text{real}} + \mathcal{L}_{D,\text{cls}} + \mathcal{L}_{D,\text{fake}}$$

## Generator Loss

- Adversarial loss (fool D)
- L1 reconstruction loss ( $\lambda = 20$ )
- Classification-forced healthy label

$$\mathcal{L}_{G,\text{adv}} = \text{BCE}(D_{\text{adv}}(G(x)), 0.92)$$

$$\mathcal{L}_{G,\text{rec}} = \|G(x) - x\|_1$$

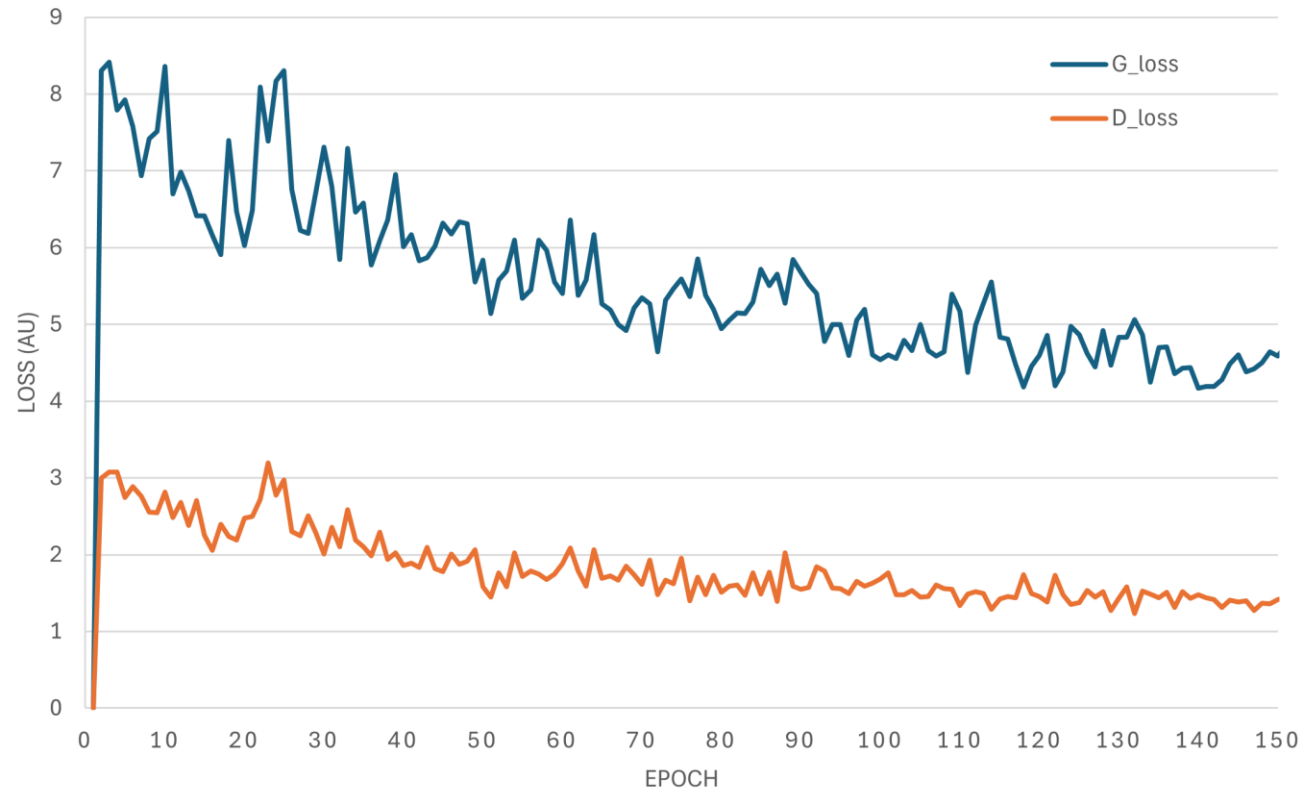
$$\mathcal{L}_{G,\text{sup}} = \text{CE}(D_{\text{cls}}(G(x)), 0)$$

$$\mathcal{L}_G = \mathcal{L}_{G,\text{adv}} + \lambda \mathcal{L}_{G,\text{rec}} + \gamma \mathcal{L}_{G,\text{sup}}$$



# Training Pipeline

GENERATOR VS DISCRIMINATOR LOSS



- Implemented in *trainer.py*
  - Alternating G/D updates
  - Extra G steps when D becomes too strong
  - Severity metric from top-k reconstruction error
- Optimization
  - Adam optimizers ( $\beta_1=0.5$ ,  $\beta_2=0.999$ )
  - Adaptive update frequency
  - Label smoothing (0.92 real)

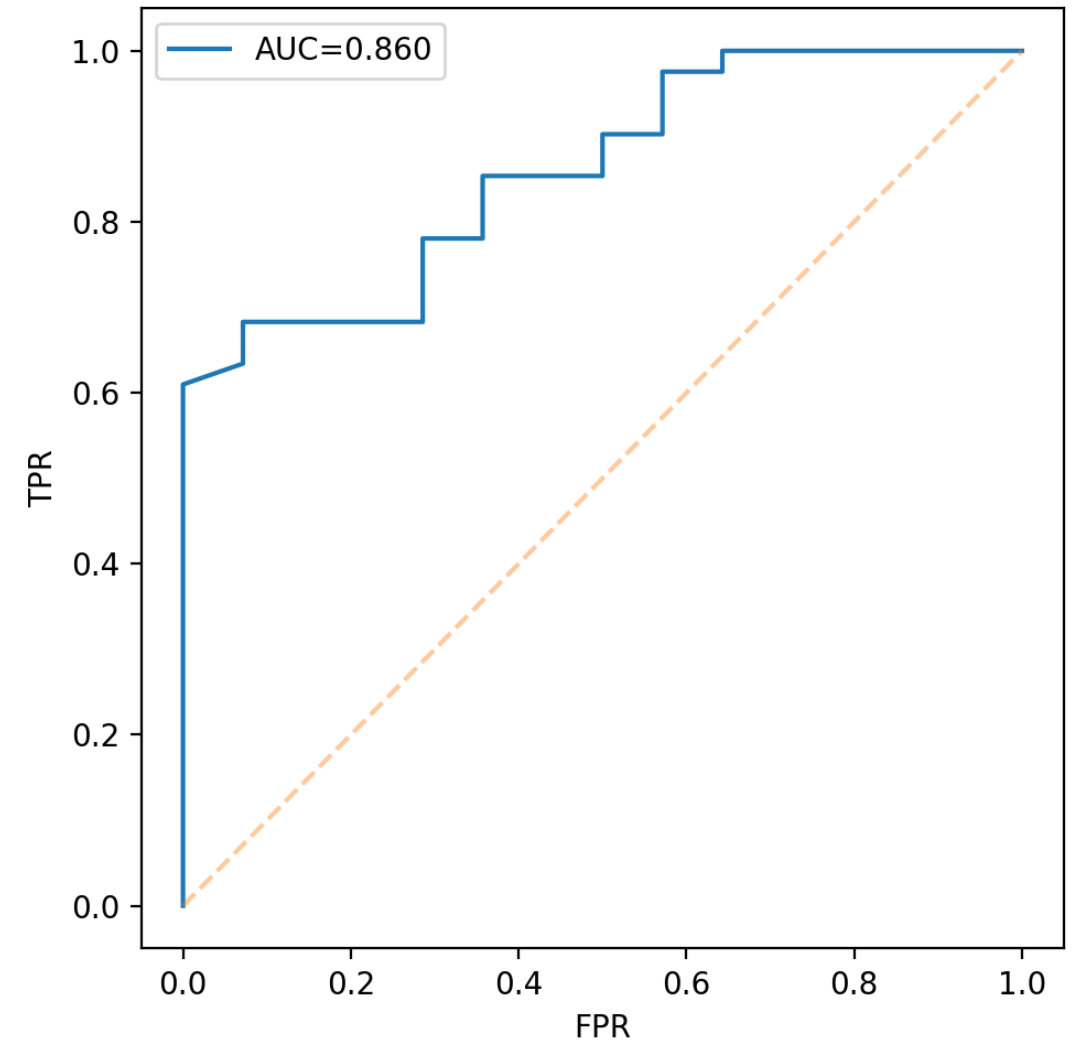


# K-Fold Cross Validation

epoch	G_loss	D_loss	val_auc
1	8.305547202	2.994782854	0.698606272
2	8.417826264	3.072160465	0.757839721
3	7.789292936	3.077182584	0.677700348
4	7.922417923	2.741973568	0.527874564
5	7.576630327	2.885773747	0.766550523
6	6.93380234	2.758417169	0.724738676

Output per-epoch metric csv example

- Implemented in *train\_cv.py*
  - Stratified splits
  - Per-epoch metrics: G loss, D loss, AUC
  - Saves checkpoints each epoch
  - Combined summary CSV
- Useful for small NDE datasets with label imbalance

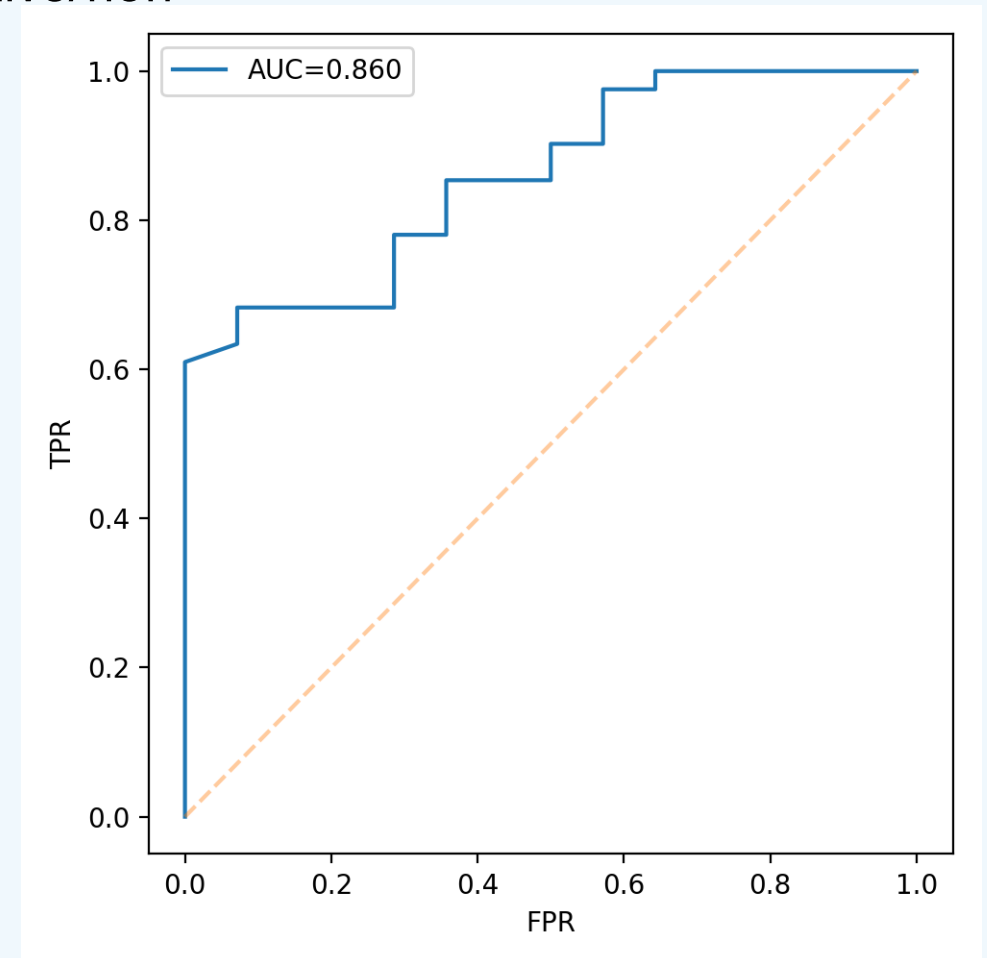
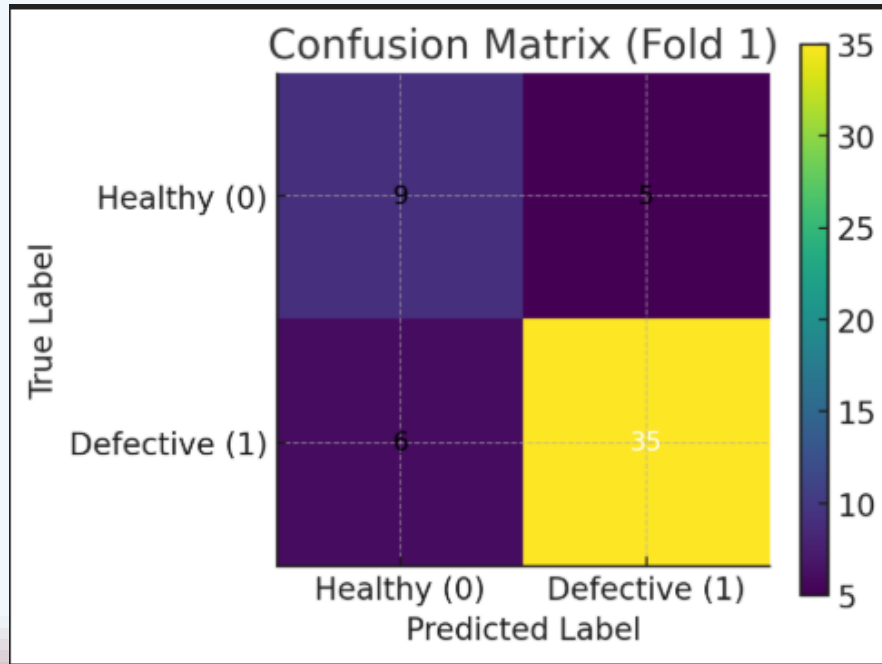


ROC curve for Fold 1 with corresponding AUC



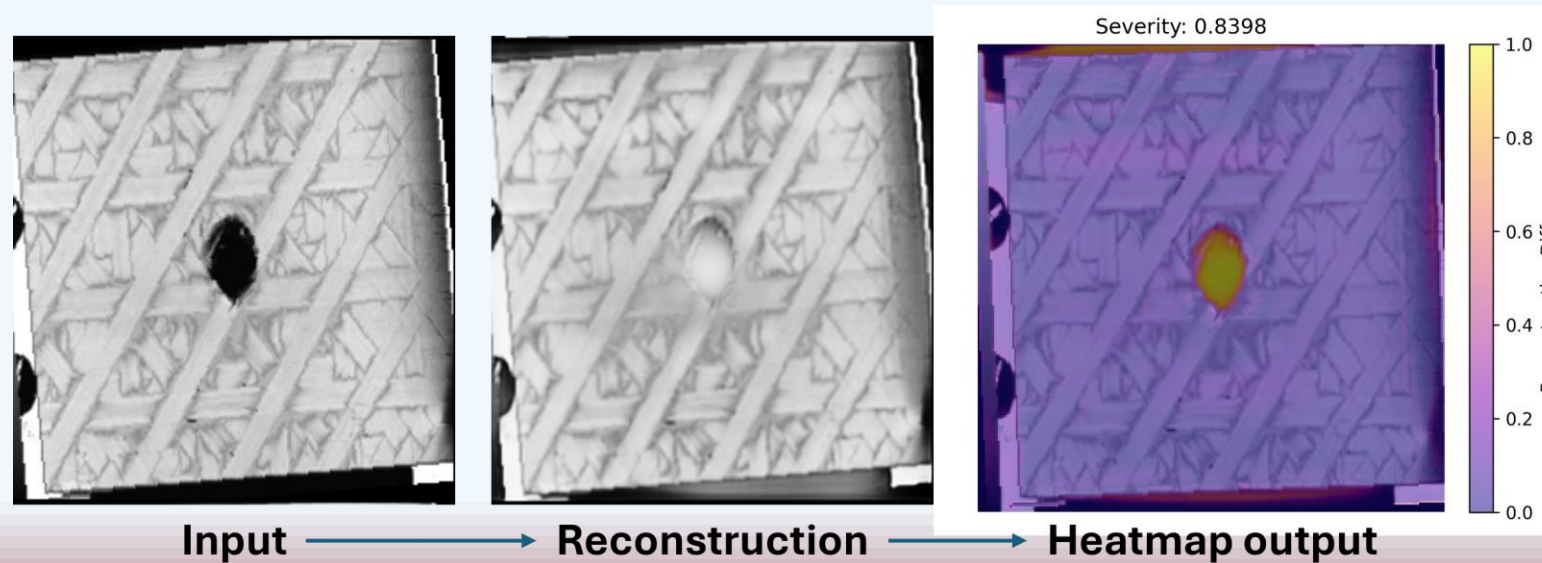
# Results: Classification

- Discriminator achieves strong separation between defective/non-defective samples
- Consistent ROC-AUC across folds
- Low cross-fold variance indicates stable generalization



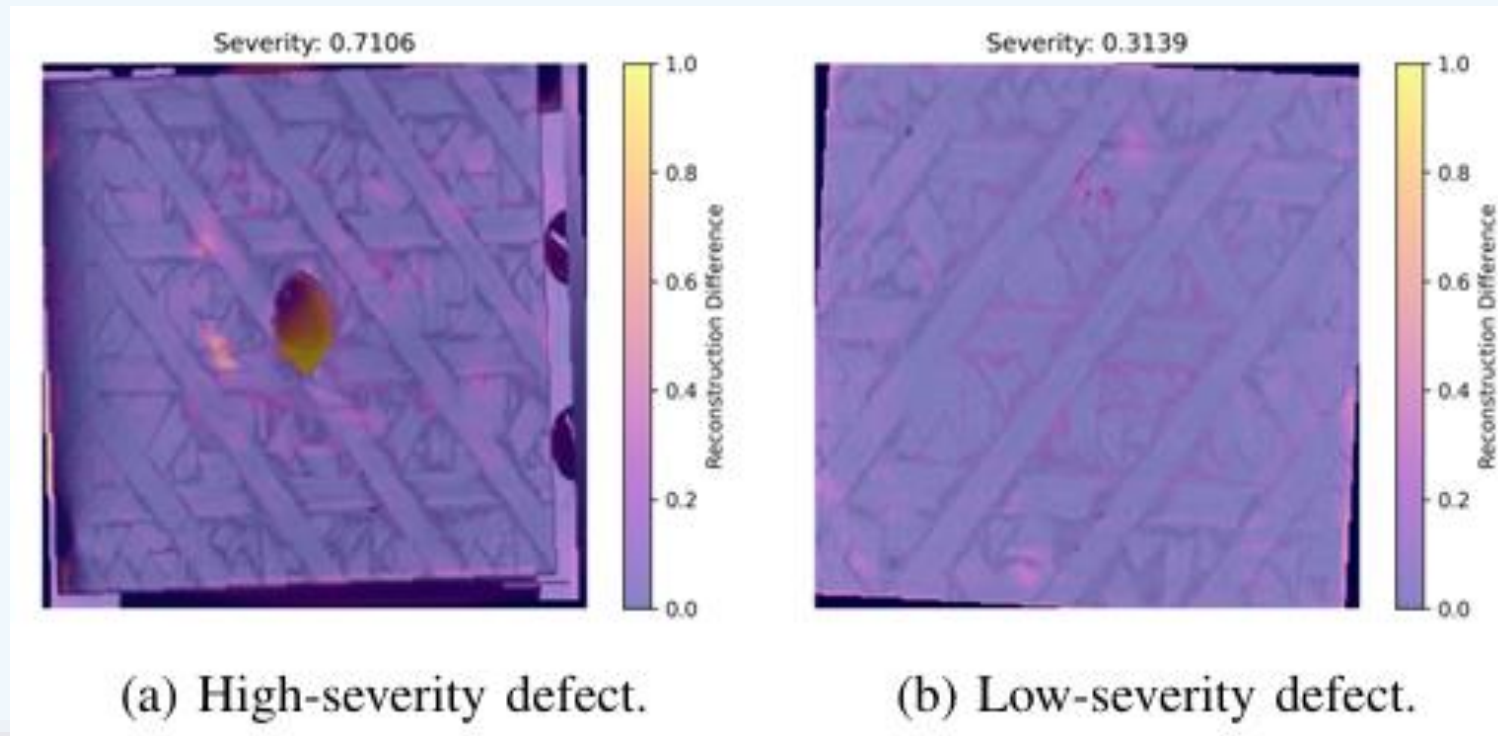
# Results: Reconstruction & Localization

- Generator reconstructs smooth “healthy” structure
- Residual =  $|\text{input} - \text{reconstruction}|$  highlights local defects
- Produces intuitive heatmaps
- Severity score computed using top-k residual intensity



# Results: Severity Estimation

- Top-k residual error correlates with physical defect magnitude
- High-severity vs low-severity defects clearly separated



# Broader Impact

## Benefits:

- Automated, interpretable NDE
- Less reliance on large labeled datasets
- Provides classification + localization + severity estimation
- Extensible to other modalities (B-scan, A-scan)

## Limitations:

- GAN training instability
- Severity metric empirical (not physically constrained)
- Reconstruction may blur subtle fine-scale defects



# Future Work

- Add **physics-informed loss terms** incorporating ultrasonic wave propagation
  - PDE-constrained generator penalty
- Expand to multi-modal ultrasonic data
  - A-scans, B-scans, multi-frequency inputs
- Experiment with newer architectures
  - **Diffusion-based** reconstruction models
  - Vision **transformers** for discriminator
- Improve severity metric calibration using physical simulation



# References

- [1] K. K. Prajapati, A. Ghosh, and M. Mitra, “Semi-supervised generative Adversarial Network (SGAN) for damage detection in a composite plate using guided wave responses,” *Mechanical Systems and Signal Processing*, vol. 232, p. 112686, Jun. 2025. doi:10.1016/j.ymssp.2025.112686
- [2] K. K. Prajapati, “Lamb Wave-Based Structural Health Monitoring using Deep Learning Techniques,” thesis, 2025
- [3] J. Hooks, “CSCE768 Final-Project,” GitHub, Dec. 2025. [Online]. Available: [https://github.com/jhooks5313/CSCE768 Final-Project.git](https://github.com/jhooks5313/CSCE768-Final-Project.git)
- [4] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” in *Proc. NeurIPS*, 2016.
- [5] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, “Unsupervised anomaly detection with generative adversarial networks to guide marker discovery,” in *Proc. IPMI*, 2017.
- [6] M. Raissi, P. Perdikaris, and G. E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, 2019.
- [7] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, 2021.
- [8] J. Hu, “Lecture 9: Generative Models,” CSCE 768 Pattern Classification, Dept. of Computer Science and Engineering, Univ. of South Carolina, 2025, [PowerPoint slides].

