**About Project**

**🧠 About the Project**

**🔍 Purpose**

This project focuses on **automated flaw detection in Non-Destructive Testing (NDT)** using deep learning. It helps identify **flaw (defect)** and **no-flaw (normal)** regions in industrial inspection images (ultrasonic, X-ray, etc.) to improve quality, safety, and throughput.

**🧩 Dataset**

We use the **Koomas NDT\_ML\_Flaw** dataset. Data is extracted into .npy shards (images.f16.npy and labels.u1.npy) with a manifest.json that indexes parts and counts for efficient IO.

**🧠 Model Overview**

A CNN is trained (via train\_model.py) to classify each image strip as:

* 1 → Flaw
* 0 → No Flaw

Preprocessing includes normalization and cropping a specific flaw band (≈ 1100–3100 px from a 7168-px width), mirroring training/inference for consistency. Models and artifacts are saved in models/ (best.keras, optional threshold.txt, training\_log.csv, metrics.txt).

**⚡ Prediction Pipeline**

1. Load the trained model (best.keras / final.keras).
2. Preprocess (crop, resize, normalize) each strip.
3. Output a **probability** for *flaw*.
4. Apply a configurable **decision threshold** to classify **Flaw** vs **No Flaw**.

**📊 Streamlit Features**

* **Predict:** Upload PNG/JPG or pick a row from dataset shards for inference.
* **Threshold Tuning:** Confusion matrix, Precision, Recall, F1, ROC-AUC vs threshold.
* **Validate (Stats):** Class balance and shard size distribution.
* **Explore Dataset:** Browse sample rows and labels.
* **Training Logs:** View training CSV logs and metrics.

**🏭 Industrial Application**

Integrates into QC pipelines to automate detection, reduce inspection time, and provide auditable statistics.