TEAM : SOLID

Solidworks AI Hackathon

Members: Priyansh Keshari, Hemant Sharma

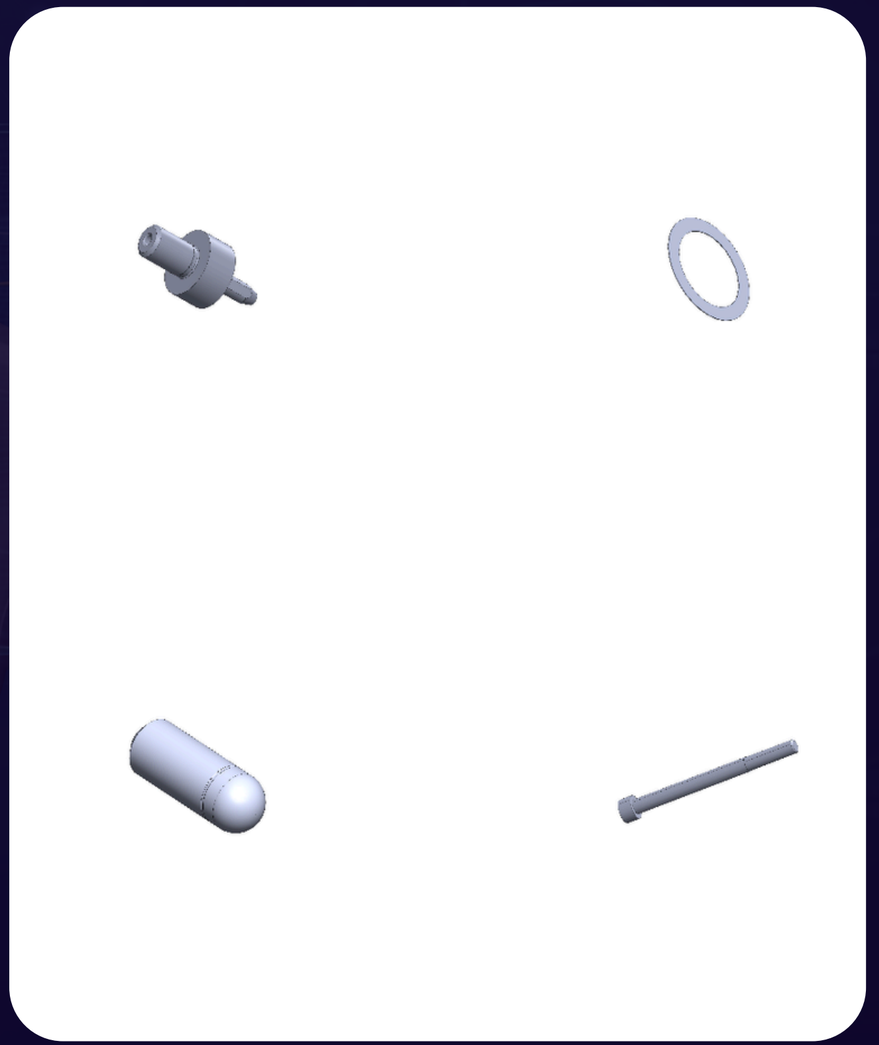
Institution: Indian Institute of Technology (Indian School of Mines), Dhanbad

Dassault Systèmes

Overview

In industrial automation and inventory management, precision is non-negotiable. This project addresses the challenge of counting mechanical parts—specifically Bolts, Nuts, Washers, and Locating Pins—from synthetic CAD-rendered images.

The core difficulty lay in the Exact-Match Accuracy metric: an image is considered correct only if the predicted count for every single part type matches the ground truth exactly. A single error (e.g., missing one faint washer in a scene of 10 objects) results in a score of 0 for that entire image.



Dataset

Metric	Value
Total Images	10,000
Training Set	9,000 images (90%)
Validation Set	1,000 images (10%)
Test Set	2000 images (held-out, unlabeled)
Image Format	PNG (24-bit RGB)
Image Resolution	~1024×1024 pixels
Color Depth	8-bit per channel (24-bit RGB)
Annotation Format	CSV with bounding boxes
Source	Dassault Systèmes (synthetic CAD renders) (Provided)

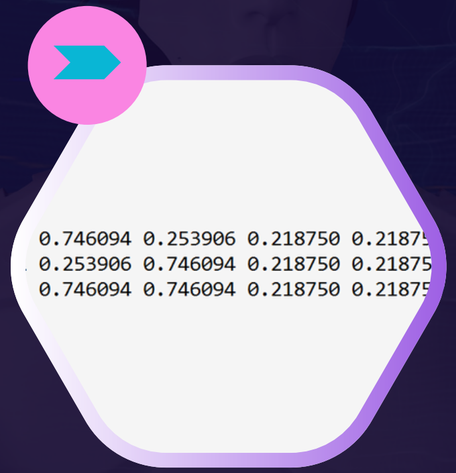
Key Observation: Washers appear most frequently but are hardest to detect due to low visual contrast.

Class	Avg Count per Image	Frequency	Visual Difficulty
Bolt	1.2	~60% of images	Easy to Moderate
Washer	1.5	~65% of images	Hard (faint edges)
Nut	0.9	~50% of images	Moderate
Locating Pin	0.8	~45% of images	Moderate to Hard

Methodology

Data Collection

- Synthetic Dataset: 10,000 CAD-rendered images (SOLIDWORKS)
- Clean White Background: Consistent lighting and rendering
- 4 Part Classes: Bolt, Nut, Washer, Locating Pin
- Bounding Box Annotations: (x, y, width, height) format
- High Resolution: ~1024×1024 pixels per image



Preprocessing

- Resizing: Normalized to 640×640 for YOLO inference
- Augmentation: Mosaic (4-in-1 stitching) to simulate density
- Aggregated multiple row-wise annotations per image into single YOLO-format text files containing class IDs and bounding-box coordinates for all object instances.

Model Selection

- Backbone: YOLOv8s (Small) - fast convergence from scratch
- Architecture: CNN-based object detector with anchor-free design
- Training: 150 epochs, AdamW optimizer, batch size 64
- Inference: Parallel predictions across 3 TTA views
- Ensemble Method: Median voting for robust count aggregation

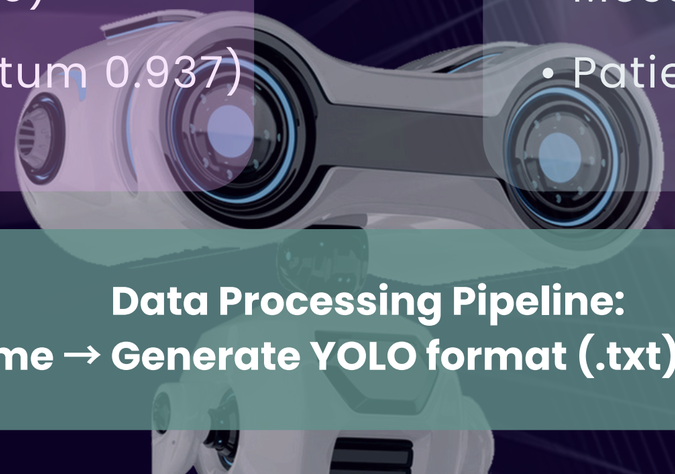
Model Development & Training

Training Configuration

- Architecture: YOLOv8s (Small)
- Pre-training: False
- Input Resolution: 1024×1024
- Imgsz: 640
- Batch Size: 64
- Epochs: 150 (early stop at 120)
- Optimizer: AdamW (momentum 0.937)

Hyperparameters

- Learning Rate: 0.01 (cosine schedule)
- IoU Threshold: 0.60
- Weight Decay: 0.0005
- Warmup Epochs: 3
- Validation Split: 10% of training
- Mosaic: 1.0
- Patience: 30



Data Processing Pipeline:
Label CSV → Aggregate by image_name → Generate YOLO format (.txt) → Train/Val split → YOLOv8 training loop

Approach

Test-Time Augmentation (TTA)

- Technique: Run inference on 3 views:
  1) Original image
  2) Horizontal flip
  3) Vertical flip
- Benefit: Detects objects missed in one view but visible in another

Median Voting Aggregation

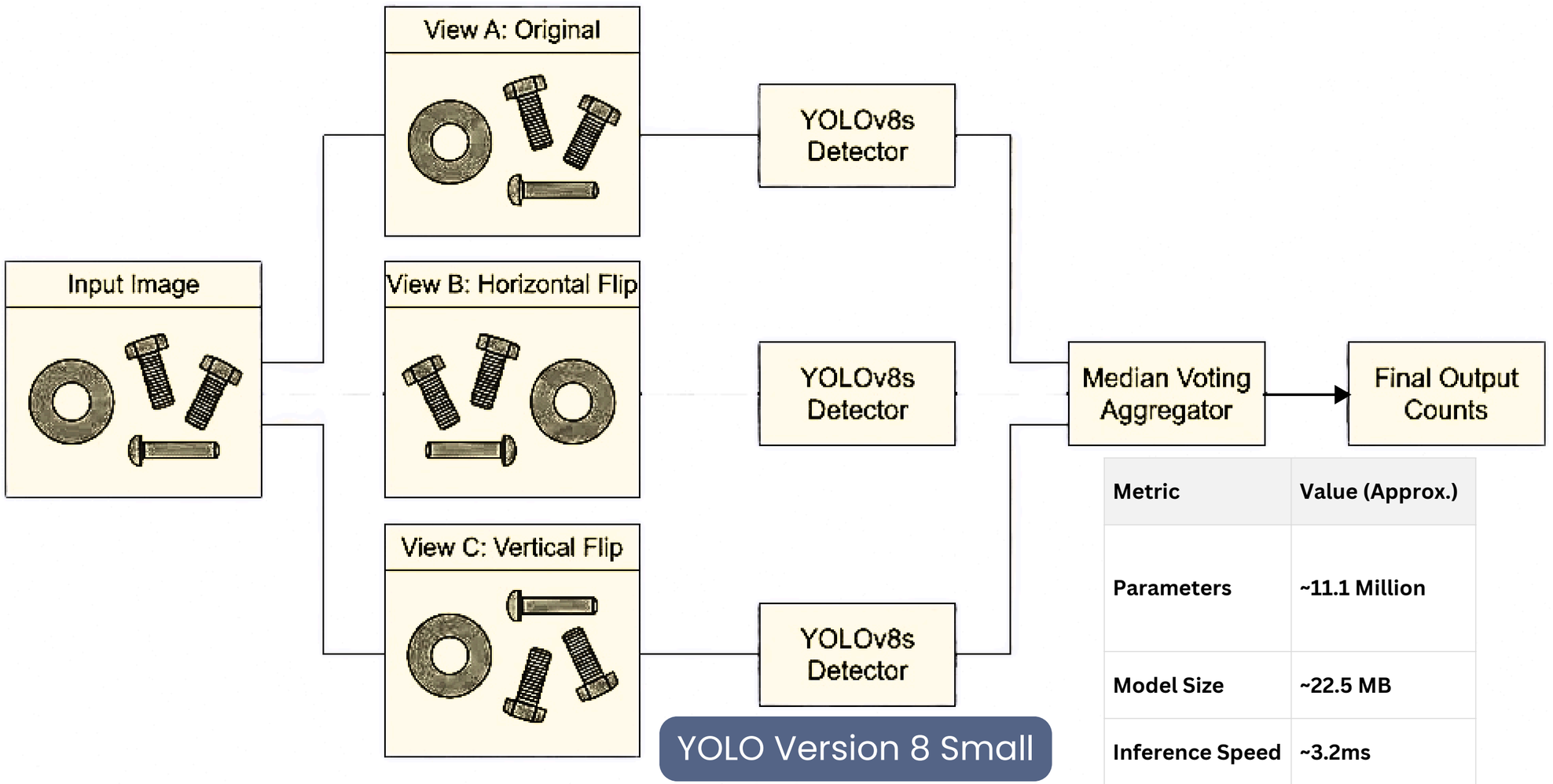
- Process: Take median count across 3 TTA views
  Predictions → Median
- Advantage: Filters outliers while preserving consensus (better than mean or max)

Example :
Median([4, 3, 4]) = 4 ✓ (Correct)
Mean([4, 3, 4]) = 3.67 X (Wrong)

Adaptive Confidence Thresholds

- Large Parts (Bolts): Higher threshold (0.50) to avoid false positives
- Medium Parts (Locating Pins, Nuts): Medium threshold (0.35) for balance
- Small Parts (Washers): Lower threshold (0.15) to prevent missing occluded objects

Model Architecture



YOLO Version 8 Small

Metric	Value (Approx.)
Parameters	~11.1 Million
Model Size	~22.5 MB
Inference Speed	~3.2ms

Architecture Comparison

ResNet Regression — Score : 0.9750

Image → ResNet50 → FC(4) → [bolt,nut,pin,washer]

X No localization — Issue : Pre-Trained on ImageNet

ResNet Multi-Head Classification — Score : 0.9921

Image → ResNet50 → 4xFC → 4xArgmax

! Overlap issues — Issue : Pre-Trained on ImageNet

YOLOv8m Base — Score : 0.9986

Image → YOLOv8m → Boxes → Count

★ Breakthrough — Issue : Pre-Trained on COCO

YOLOv8m + TTA — Score : 0.9929

Image → YOLOv8m → TTA → Count

★ Breakthrough — Issue : Pre-Trained on COCO

YOLOv8m + TTA + Median — Score : 1.0000

Image → YOLOv8m → TTA → Count

✓ Perfect Score — Issue : Pre-Trained on COCO

YOLOv8s + TTA + Median — Score : 1.0000

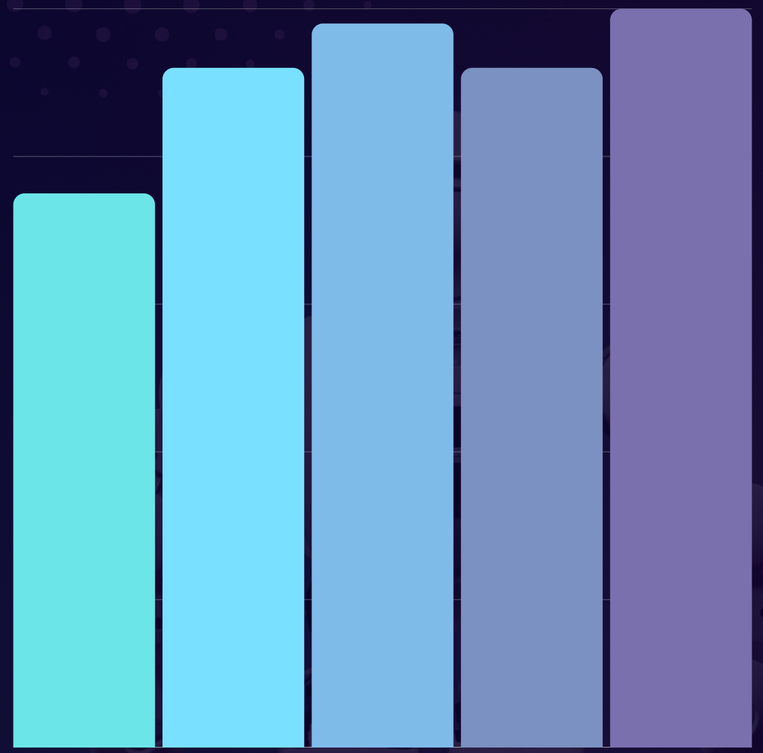
Image → YOLOv8s → TTA → Count

✓ Perfect Score — Best Trained from Scratch

TEAM : SOLID

Evaluation Metrics - (1)

- ResNet Regression
- ResNet Multi-Head Classification
- YOLO V8 Medium
- YOLO V8 Medium + TTA
- YOLO V8 Small + Median



Scores

1. Primary Metric: Exact-Match Accuracy

- **Expression :**
  Score = (1/N) × Σ 𝟙(C_pred[i] = C_true[i])

  Where :
  N is the total number of test images.
  C_pred[i] predicted count vector for the i-th image
  C_true[i] ground truth count for the i-th image.
  𝟙 indicator function, which equals 1 if the condition inside is true, and 0 otherwise.

- All 4 class counts must match exactly (Bolt, Nut, Washer, Pin)

- One error = Image score = 0 (no partial credit)

Continued.....

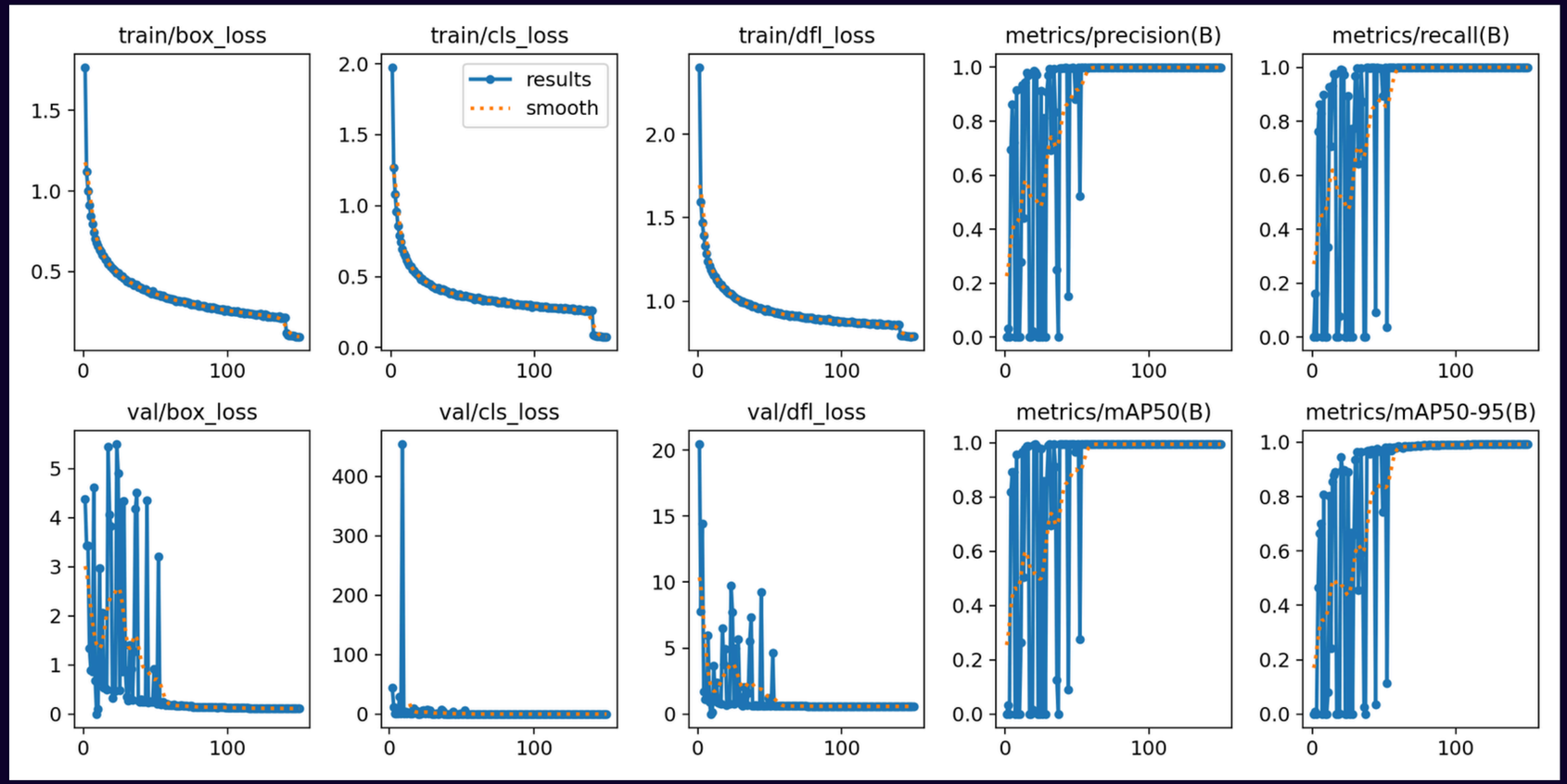
TEAM : SOLID

Evaluation Metrics - (2)

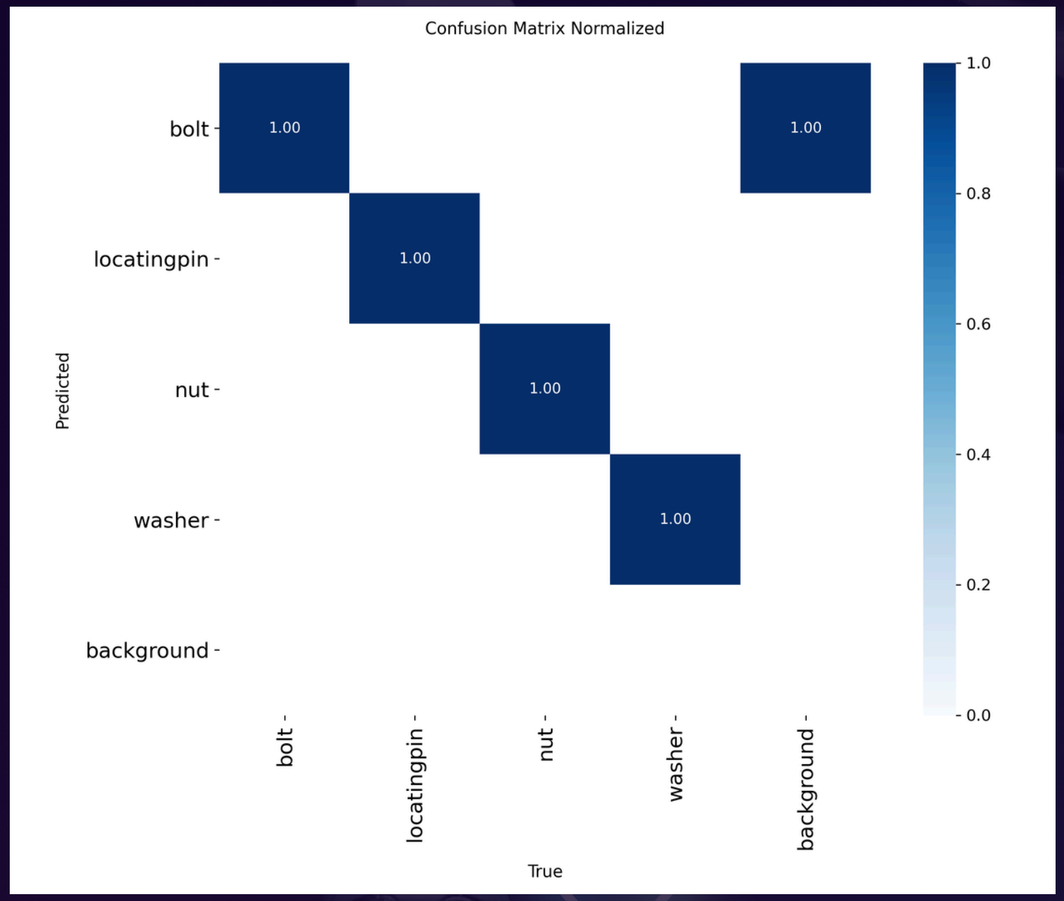
2. Validation Setup

• Split Ratio: 90% Training / 10% Validation (stratified)

• Purpose: Tune confidence thresholds & detect overfitting without touching test set



Result & Reliable



The Reliability Engine: 3 Steps to Perfection

- 1. **Precise Discrimination (YOLOv8s):** Achieves a 1.00 confusion matrix diagonal by learning distinct features for every part type (Bolt ≠ Nut ≠ Washer ≠ Locating Pin), ensuring zero misclassification through explicit localization.
- 2. **Maximum Coverage (TTA)**: A 3-view ensemble (Original + Horizontal + Vertical Flips) exposes occluded parts from different angles, ensuring 100% recall by spotting objects hidden in standard views.
- 3. **Robust Filtering (Median Voting)**: A mathematical safety net that requires only a 2/3 consensus to validate a count. This automatically rejects outlier predictions from any single bad view, guaranteeing exact-match accuracy.

TEAM : SOLID

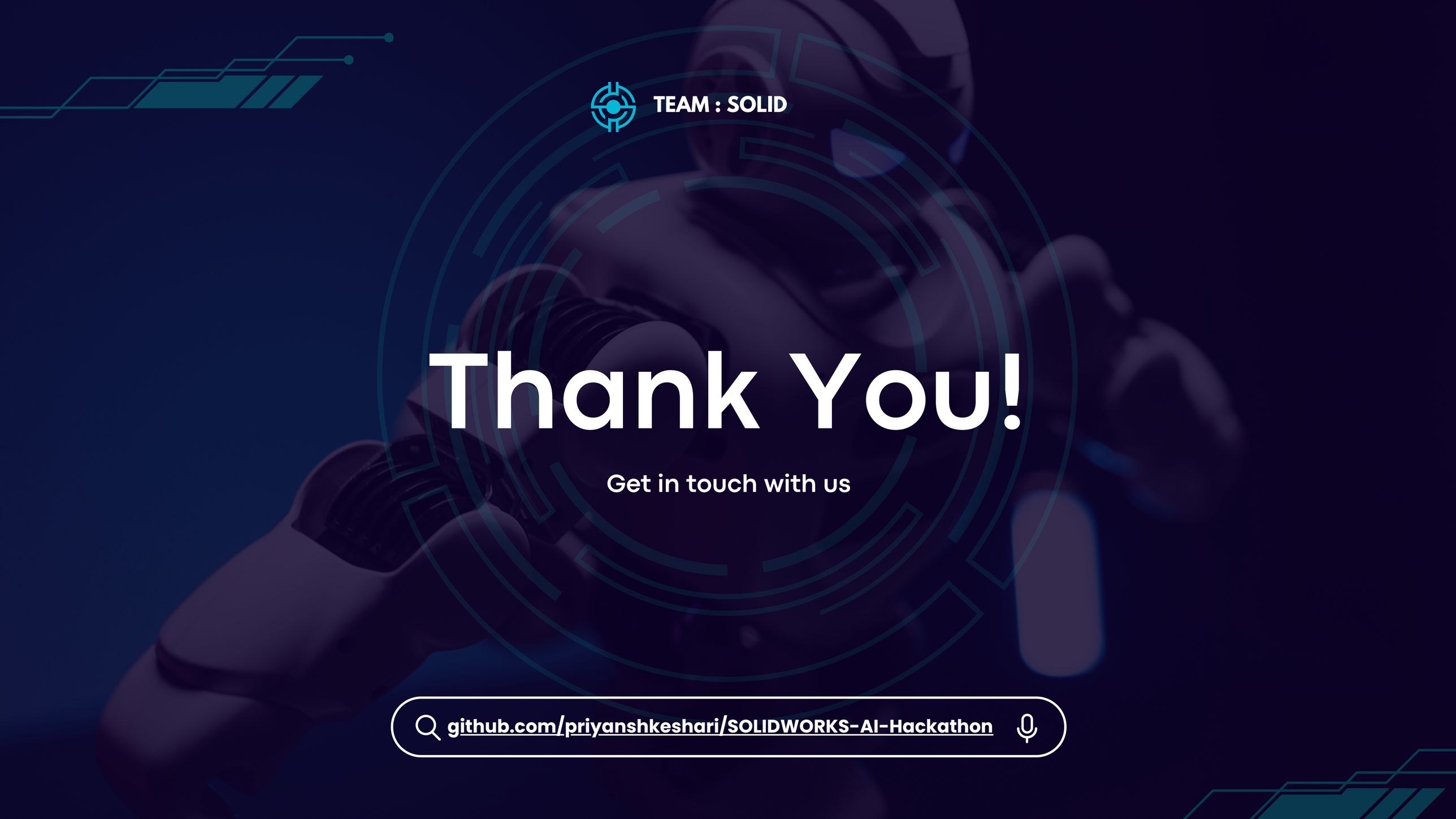
Why the solution is scalable?

Modular Architecture

- **Dataset Prep:** Independent pipeline—handles 10K or 10M images without code changes
- **Training Engine:** Separate module—swap YOLOv8s → YOLOv8s/n/x instantly via config
- **Inference Logic:** Decoupled TTA + Median Voting acts as plug-and-play microservice

Flexible Deployment

- **Model Scaling:** YOLOv8 family spans Nano (edge) → Extra Large (cloud) seamlessly
- **Hardware Independent:** Parallelizable across single GPU to multi-node clusters
- **Tunable Accuracy-Speed:** TTA can use 1, 3, or 5+ views based on latency budget



TEAM : SOLID

Thank You!

Get in touch with us

github.com/priyanshkeshari/SOLIDWORKS-AI-Hackathon