PROBLEM STATEMENT AND TEAM DETAILS

Problem Statement: Space Station Object Detection Using YOLOv8

Team Name: Git Push Mafia

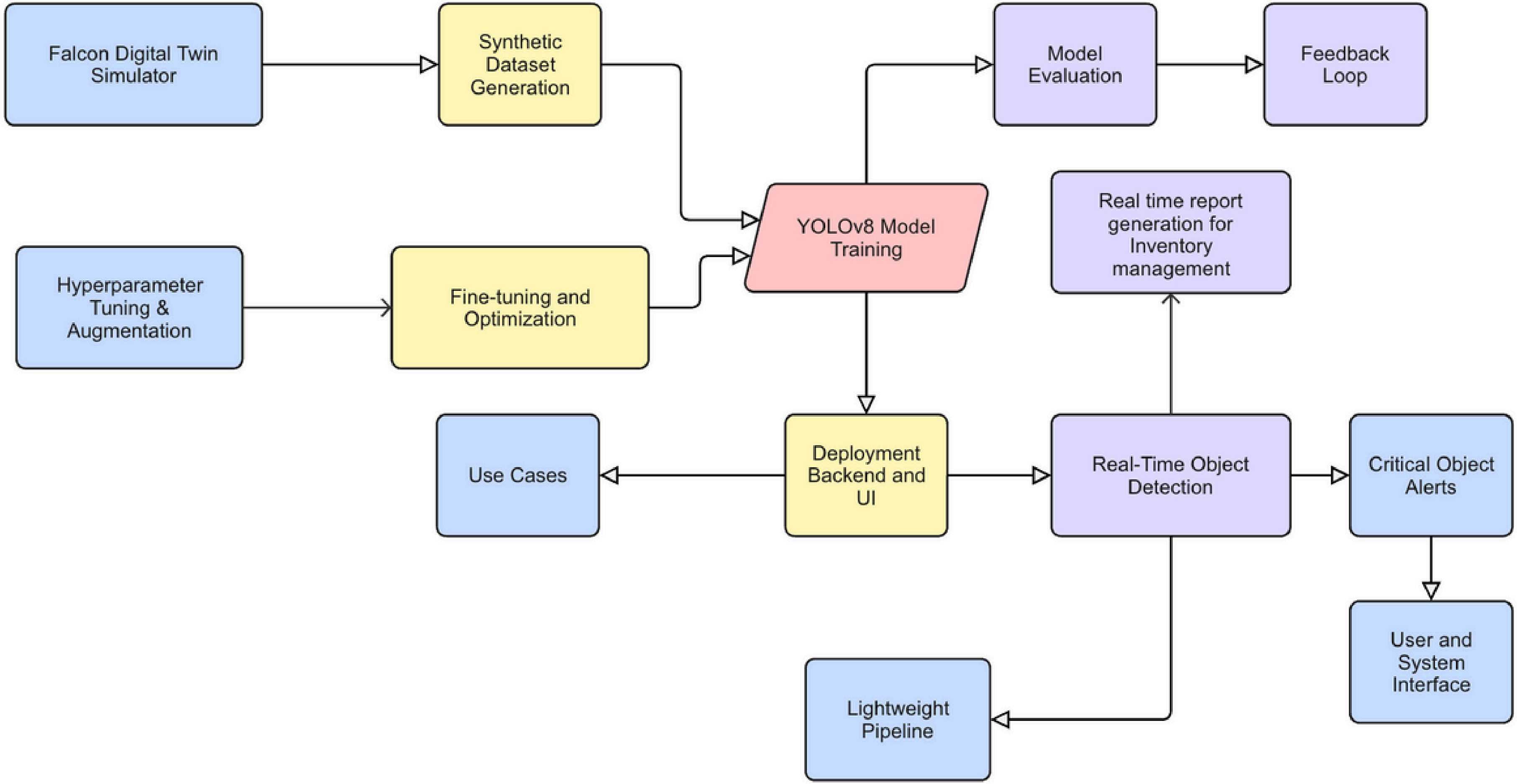
Team Leader Name: Sarthak Pandey

Institute Name: DTU , NSUT

Theme Name: Safe Operations with Synthetic AI

Team Leader Email ID: pandeysarthak1510@gmail.com

PROCESS FLOW DIAGRAM

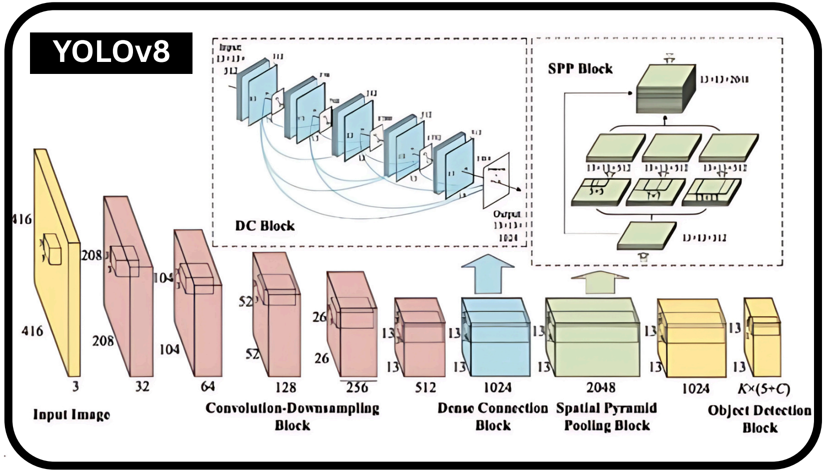


BRIEF ABOUT THE IDEA

Identifying critical safety gear like fire extinguishers and oxygen tanks is vital onboard space stations, but manual checks are risky and inefficient. With real-world data hard to access, we built an **object detection model trained entirely on synthetic data using Duality AI's Falcon digital twin.**

- Fully synthetic training using **Falcon simulation**
- Detects **essential safety equipment** with high accuracy
- Works under **occlusion, low light, and clutter**
- **Automates monitoring**, reducing crew workload and risk
- Supports **autonomous, self-monitoring** space environments

A compact, scalable solution advancing safety and autonomy in space.



Robust Object Detection
The model accurately identifies critical safety equipment in diverse space station conditions, including occlusion and poor lighting.

Advanced Fine-Tuning
Leveraged synthetic data variations and domain-specific adjustments to optimize performance for real-world-like scenarios.

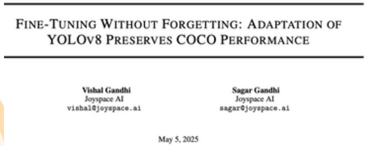
Good Evaluation Metrics
Achieved high precision, recall, and mAP-scores, ensuring reliability and minimal false detections in critical environments.

Use Case & Continuous Improvement
Designed for real-time space station monitoring, with scope for iterative training as more synthetic or real data becomes available.

COMPARITIVE STUDY OF MODELS

YOLOv8s MODEL

The training strategy was to fine-tune a pre-trained YOLOv8s model on the custom dataset.
This was done by loading the base yolov8s.pt model, configuring it with the specific dataset paths, and then re-training it for 100 epochs. The model's performance was then measured on the unseen test set to get a final accuracy score.



FINE-TUNING WITHOUT FORGETTING: ADAPTATION OF YOLOv8 PRESERVES COCO PERFORMANCE

Vishal Gandhi
Joyspace AI
vishal@joyspace.ai

Sagar Gandhi
Joyspace AI
sagar@joyspace.ai

May 5, 2025

SOTA FINE TUNING

The "Freeze 10" strategy is a fine-tuning method that locks the first 10 layers of a pre-trained model,
preserving their expert knowledge of basic features like edges and colors.
This focuses all training efforts on the deeper layers, resulting in faster, more efficient learning for your specific task with less risk of overfitting.
YOLOv8m model is fine tuned on the given dataset using this SOTA technique.

YOLOv8m STRONG AUGMENTATION

The strategy combines the powerful YOLOv8m model with strong augmentation to achieve maximum accuracy and robustness.

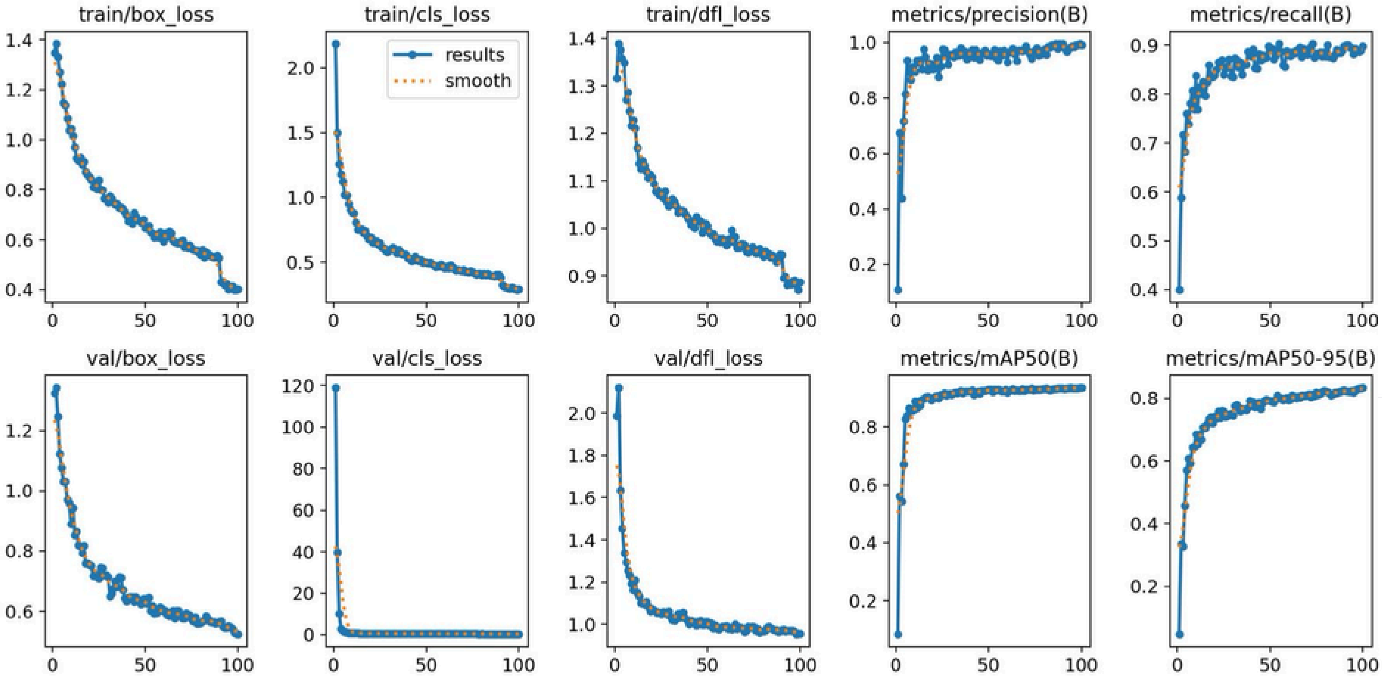
The larger YOLOv8m provides a high potential for accuracy, while aggressive augmentations—like adding fog, rotating images, or applying random dropout—create challenging training scenarios.

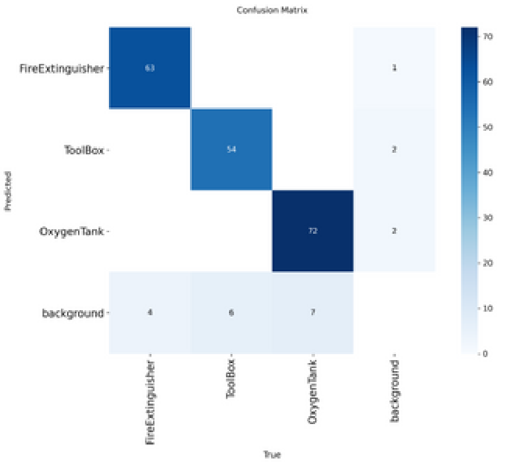
YOLOv8n MODEL

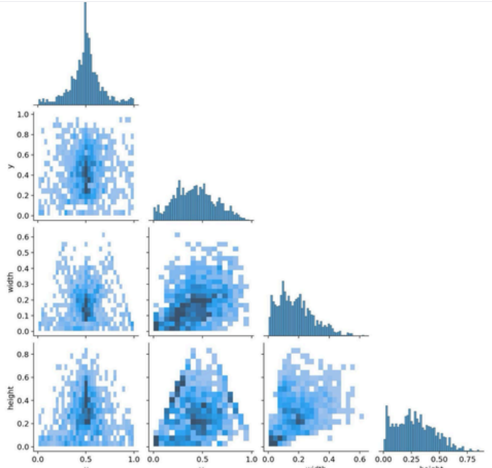
YOLOv8n ('Nano') is the smallest, fastest, and most lightweight model in the YOLOv8 series. It's specifically designed for high-speed, real-time object detection on devices with limited computational power, such as mobile phones and drones.

This focus on speed and efficiency comes with a trade-off of slightly lower accuracy compared to larger YOLOv8 models.

METRICS


train/box_loss
train/cls_loss
results
smooth
train/dfl_loss
metrics/precision(B)
metrics/recall(B)
val/box_loss
val/cls_loss
val/dfl_loss
metrics/mAP50(B)
metrics/mAP50-95(B)


Confusion Matrix
FireExtinguisher
ToolBox
OxygenTank
background
Predicted
True
63
1
54
2
72
2
4
6
7


y
width
height

FEASIBILITY

No real-world data needed

Hardware-friendly pipeline

Real-time performance

MARKET USE

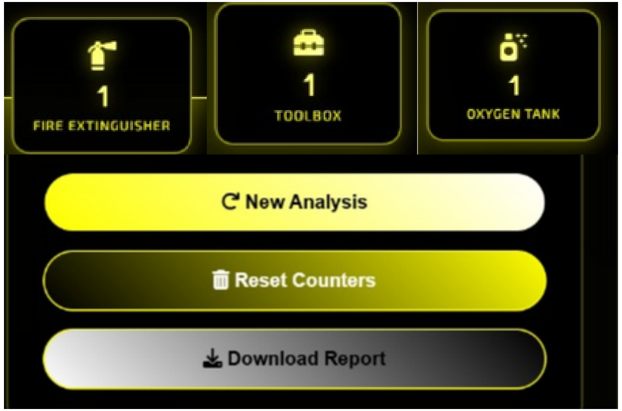
Space Agencies.

Autonomous Robotics

Beyond Space

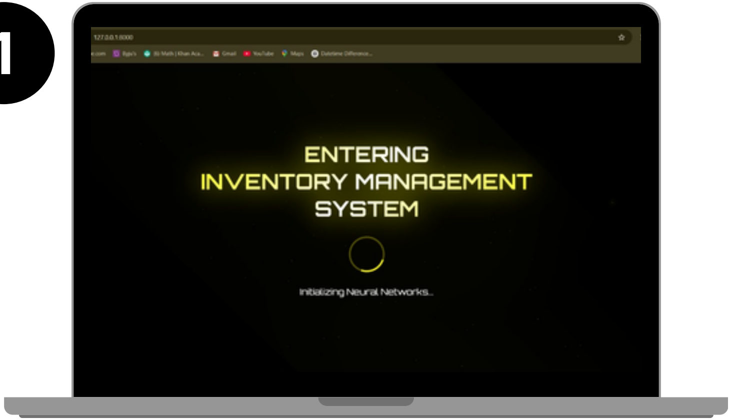
INVENTORY MANAGEMENT

Real-Time Gear Sorting Dashboard:
Safety equipment such as Fire Extinguishers, Toolboxes, and Oxygen Tanks are automatically detected and visually sorted with count indicators and intuitive icons.
CSV Export:
The complete inventory status, including detected item types, counts, and timestamps can be downloaded in .csv format for logging, audit, or further analysis.

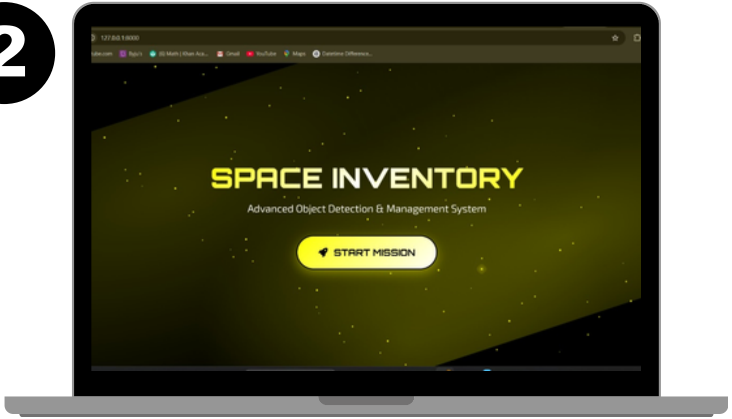


SNAPSHOTS OF PROTOTYPE

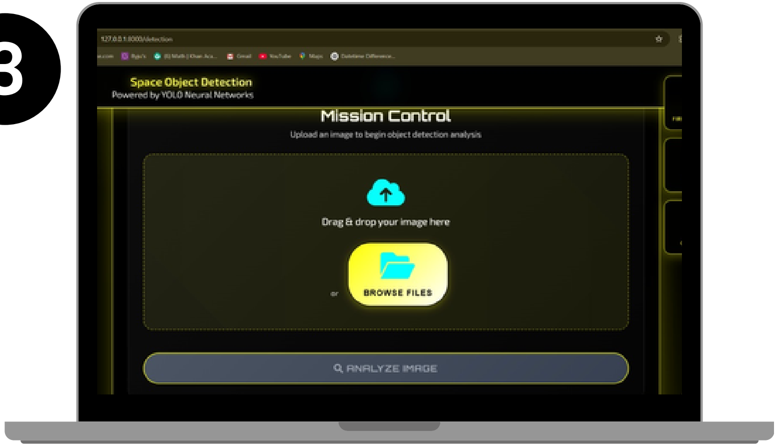
1



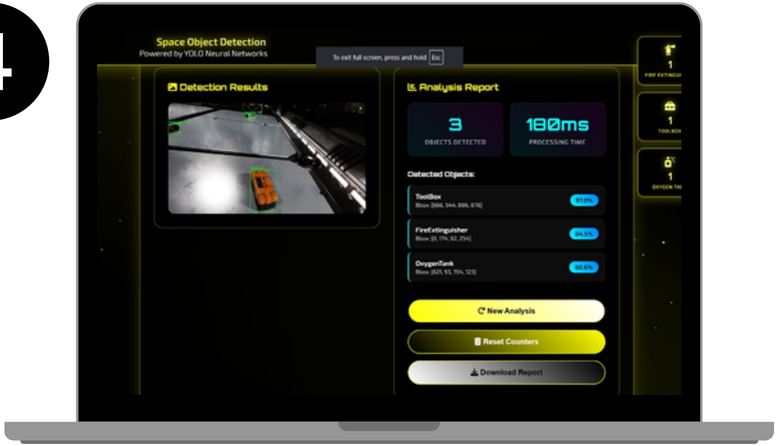
2



3



4



REAL WORLD USE CASES

Space Missions (NASA, ISRO, ESA)

- **Detect and track objects like toolboxes, oxygen tanks, and fire extinguishers inside spacecraft or space stations.**
- **Enables crew safety, inventory automation, and autonomous system support where manual monitoring is risky or impossible.**

Smart Manufacturing & Industry 4.0

- **Real-time detection of tools or machinery in hazardous environments.**
- **Prevents operational errors due to misplaced or missing equipment.**
- **Ideal for robotic arms, automated assembly lines, and remote inspections.**

Education & Simulation Training

- **Falcon-based simulation helps train students, astronauts, or engineers in identifying and responding to objects in realistic digital environments.**
- **Supports VR/AR-based safety drills and skill-building modules.**

Disaster Response & Defense

- **Deployed in drones or robots to detect essential tools or safety equipment in fire zones, earthquake rubble, or war zones.**
- **Useful when sending humans in is unsafe or time-sensitive.**

TECHNOLOGY USED

FastAPI is used to build the backend API.
Jupyter is used for scripting and experimentation.
FALCON generates the synthetic dataset.
Python is the primary programming language.
YOLOv8 is implemented for object detection.
Seaborn and **Matplotlib** are used for plotting.
NumPy and **Pandas** handle data processing.
HTML, CSS, and **JavaScript** build the web interface.