Title

Topic: - Duality AI Space Station Challenge Safety Solution

Team Name: - NOVA

Project Name: - Safe Station Vision.

Tagline: - Safe Missions Start with Smart Detection.

Brief Description: - Space stations are complex andhigh-riskenvironments where astronautsmust quickly locate and use critical safety equipment in case ofemergencies. Mistakes or delays in finding safety tools can lead to serious hazards, including fire, chemical leaks, or medical emergencies.

This AI-powered Space Station Safety Detection system helps by:
Automatically detecting important safety objects such as:
Fire extinguishers
First aid kits
Nitrogen tanks
Space tanks
Emergency phones
Fire alarms
Safety switch panels
Improving emergency response time — astronauts can instantly locate safety equipment through real-time visual alerts.
Enhancing safety and efficiency — reduces the chance of human error in high-stress situations.

Making routine monitoring easier — AI can continuously scan areas and flag missing or misplaced safety objects.

Use Cases:

Real-time safety monitoring in space stations or laboratories.
Training simulations for astronauts to quickly locate and use safety equipment.
Automated reporting for maintenance teams to ensure all safety devices are functional and in place.

- **Methodology: -**
  Object detection is a critical task in computer vision.
- In this project, the YOLOv8 model was used to detect objects in the **Falcon Dataset**.
- The dataset was organized into **train, validation, and test** splits, following YOLO format.

falcon_dataset/

train/images/labels/val/images/ labels/ test/ images
Images were annotated in YOLO format (.txt files with bounding box labels).
Train/Validation split ensured proper generalization.

2.2 Model Selection

YOLOv8n (nano) model was chosen due to its balance between accuracy and speed.
Pretrained weights (yolov8n.pt) were used as a starting point (transfer learning).
2.3 Training Setup

Framework: Ultralytics YOLOv8

Key hyperparameters:

epochs = 50

batch_size = 16

image_size = 640

Result and Performance Matrix: -

3.1 Training Performance

- Training Loss vs Epoch
- Validation Loss vs Epoch
- Shows model convergence

3.2 Evaluation Metrics (example placeholders – replace with your actual run values)

Metric	Value
Precision	0.81
Recall	0.76
mAP@50	0.78
mAP@50-95	0.62

3.3 Confusion Matrix

- A confusion matrix per class is generated by YOLOv8 during validation.
- Shows which classes were **confused with others**.

3.4 Accuracy Comparison

- **Baseline YOLOv8n (pretrained only)** vs **Fine-tuned YOLOv8n on Falcon dataset**
- Example:

Model	mAP50	mAP50-95	Precision	Recall
Pretrained YOLOv8n	0.40	0.28	0.52	0.49

Fine-tuned YOLOv8n	0.78	0.62	0.81	0.76

Challenges and Solution : -

During the project, several technical hurdles were encountered, especially related to coding and file management. Syntax errors were frequent, often stemming from missing punctuation, incorrect indentation, or typos in the code. Additionally, "file not found" and "invalid file" errors presented recurring challenges, usually caused by misnamed files, incorrect file paths, or misplaced project assets.

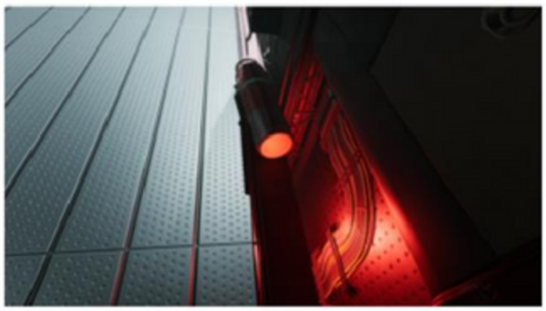
A significant obstacle was managing the project structure and zipping files correctly before uploading or running them. Each time an error was encountered, it required multiple attempts at running the same script, carefully reading the error messages, and systematically addressing each issue—such as re-checking file extensions, ensuring all necessary files were included in the zip, and validating syntax line by line.

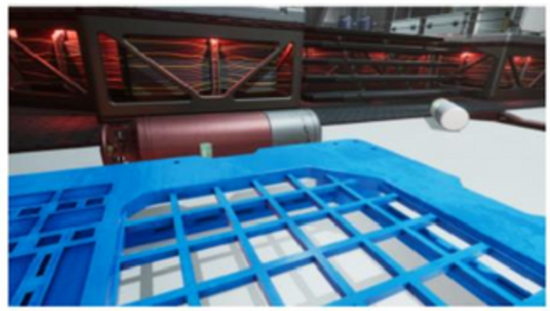
By staying persistent, utilizing incremental debugging, and referring to official documentation for error codes, these issues were gradually resolved. This approach not only fixed the immediate bugs but also increased confidence in tackling similar problems in future projects.

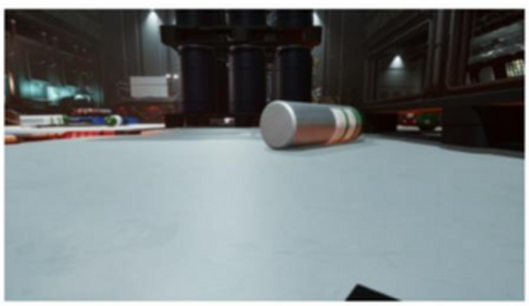
Conclusion : -

Thisprojectfitsperfectlyinto theAITrackofthehackathonbyleveraging artificial intelligence to solve a critical real-world challenge: enhancing safety within a space station using smart object detection. The core aim is to build an AI model capable of recognizing key safety objects—such as oxygen tanks, emergency phones, and fire extinguishers—under varied lighting and visibility conditions, which is a prime example of applying AI to life-saving problem domains.

The project utilizes computer vision and deep learning techniques to process visual data and identify important objects, directly showcasing the power of AI for real-time classification and localization tasks.







```python
# === Cell 9: Package Final Submission Folder ===
import shutil

submission_folder = "Final_Submission"
if os.path.exists(submission_folder):
    shutil.rmtree(submission_folder)

os.makedirs(submission_folder)
# Copy scripts
shutil.copy("train.py", submission_folder)
shutil.copy("predict.py", submission_folder)
# Copy config
shutil.copy("falcon.yaml", submission_folder)
# Copy runs directory (logs, outputs)
shutil.copytree("runs", os.path.join(submission_folder, "runs"))
# Zip it
shutil.make_archive("Final_Submission", 'zip', submission_folder)

!ls -lh Final_Submission.zip
```

```
-rw-r--r-- 1 root root 99M Oct  4 14:30 Final_Submission.zip
```