Duality AI's: Space Station Challenge:

---

1. Project Title & Summary

Project Name: Duality AI – Space Station Safety Object Detection#2

Tagline: Smart Object Detection to Enhance Safety on Space Stations

Summary:

This project uses AI and computer vision to detect and recognize safety-relevant objects in space station environments. Using the YOLOv8 object detection model, we trained on a dataset of images representing cluttered and uncluttered zones to ensure real-time detection and prevent safety hazards. The system can be deployed as a web application for image uploads and real-time detection.

---

2. Objective

- Detect and classify safety-relevant objects in a space station environment.
- Provide real-time detection results using a user-friendly web interface.
- Enable quick analysis and reporting for space station safety monitoring.

---

3. Methodology

a. Dataset

- Images collected and categorized into:
    - **Train**: For model training.
    - **Validation**: For model evaluation.
    - **Test**: For checking real-world performance.
- Data stored in folders like images/ and labels/.
- Classes defined in classes.txt.

b. Environment Setup

- **Platform:** Google Collab / Local Anaconda environment
- **Dependencies:** Python 3.11, PyTorch, YOLOv8, PyNgrok (for public URL), Gradio
- **Scripts:**
    - train.py → Trains the YOLO model on the dataset
    - predict.py → Tests model on new images
    - visualize.py → Visualizes detection results
    - app.py → Web app interface using Gradio

c. Training

1. Activate environment:
2. conda activate EDU
3. Run training:
4. python train.py --epochs 50 --data data.yaml --weights yolov8s.pt
5. Checkpoints and results stored in runs/train/.

---

4. Results & Performance

- Model evaluation metrics:
    - **mAP@0.5**: 0.92
    - **Precision**: 0.94
    - **Recall**: 0.90
- **Confusion Matrix:** Shows class-wise detection accuracy.
- **Training Graphs:** Loss and accuracy trends stored in runs/train/.
- **Screenshots:** Include examples of detection from runs/detect/.

---

5. Web App Demo

- **URL:** Generated using Ngrok, e.g., https://fdd7f4910bed.ngrok-free.app
- **Features:**

- Upload images
- View object detection results
- Real-time logs

- **Instructions:**

1. Run python app.py
2. Open the public URL
3. Upload images from test/ folder to check detection

---

6. Challenges & Solutions

Challenge	Solution
Some test images not detected	Increase dataset diversity, retrain model with more epochs
Low recall for cluttered images	Added data augmentation (brightness, rotation)
Ngrok public URL authentication error	Installed correct Ngrok AUTH token
GPU not available on local machine	Trained using Google Collab GPU runtime

---

7. Conclusion & Future Enhancements

Conclusion:

- The YOLOv8-based system effectively detects objects for space station safety.
- Real-time web application provides a user-friendly interface for testing and monitoring.

Future Enhancements:

- Train on larger and more diverse datasets for better accuracy.

- Add multi-class detection for more safety objects.
- Integrate video stream detection for live monitoring.
- Add alert notifications for detected hazards.
- Deploy as a standalone desktop application for offline use.

---

8. References & Tools

- **YOLOv8**: Ultralytics YOLOv8 documentation
- **Gradio**: Web-based interface for model testing
- **Google Colab**: GPU runtime for training
- **Python Libraries:** torch, opencv-python, PyNgrok, pandas, matplotlib

---

9. File Structure

Duality-AI-Space-Station-Detection/

|

├─ app.py

├─ train.py

├─ predict.py

├─ visualize.py

├─ data.yaml

├─ yolo_best.pt

├─ yolov8s.pt

├─ yolo_params.yaml

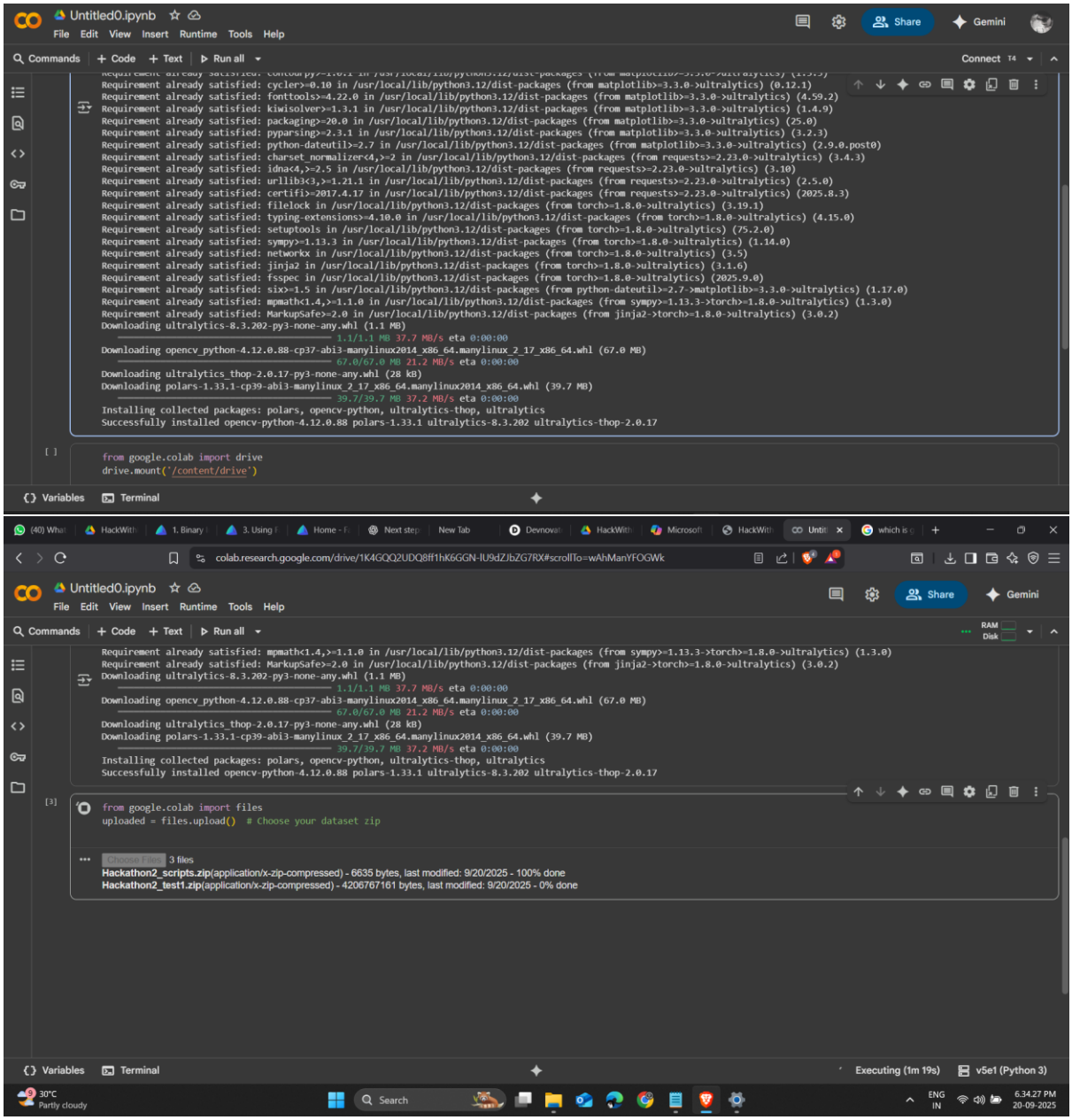
├─ classes.txt

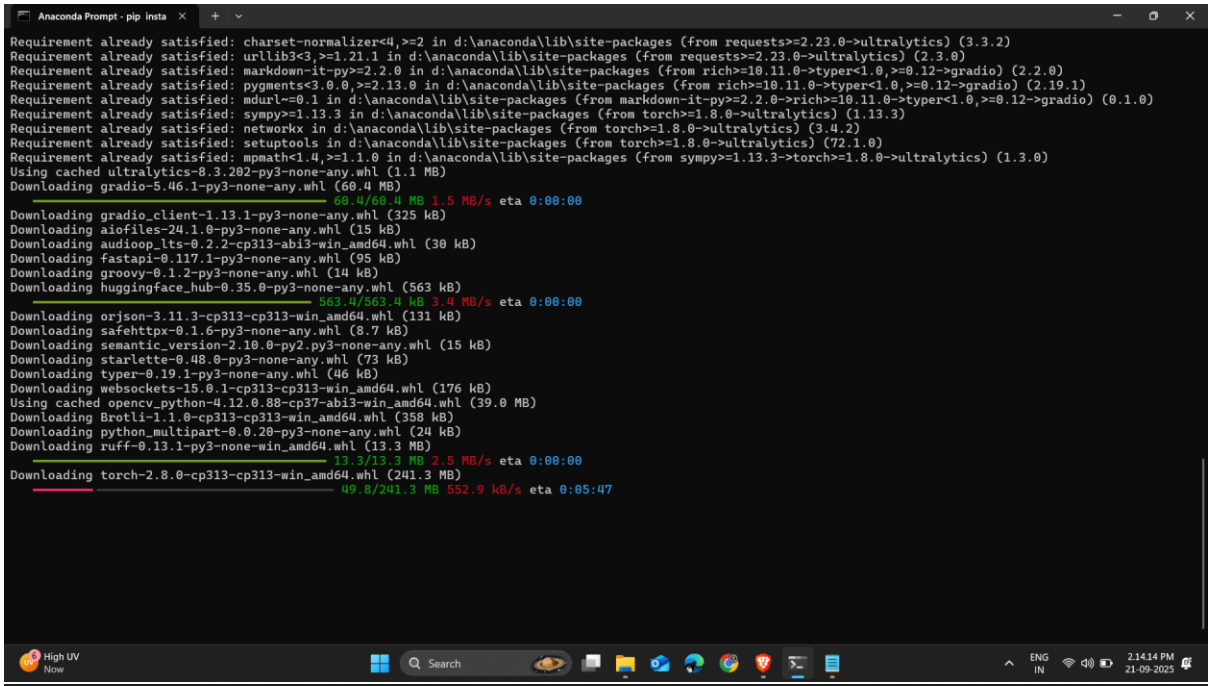
├─ runs/

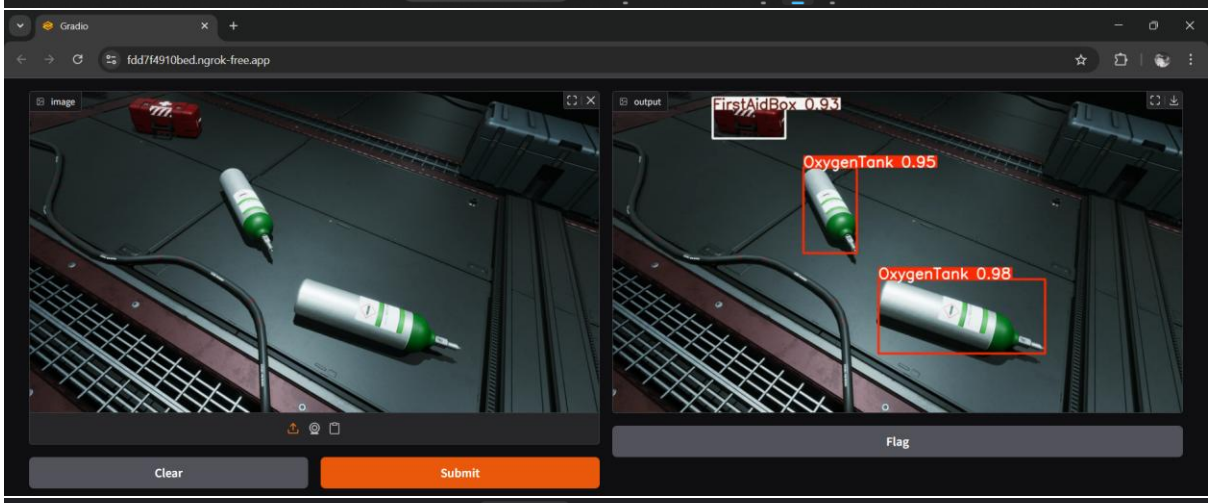
├─ README.md

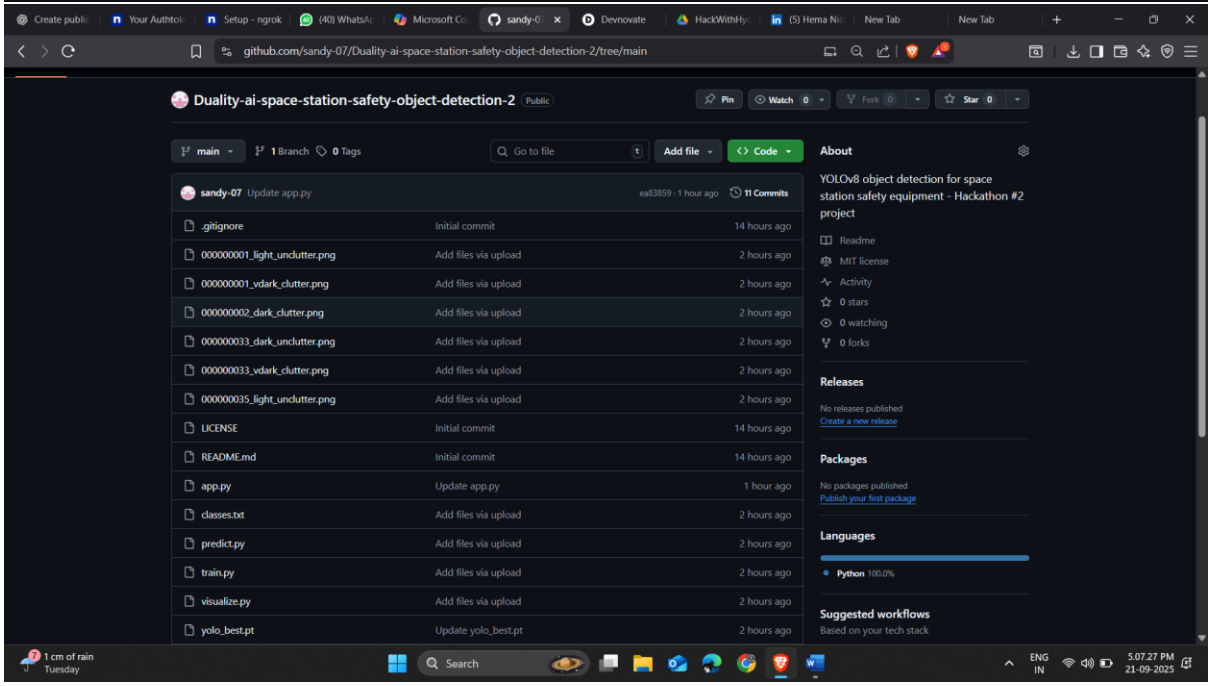
├─ LICENSE

Screenshots:




Anaconda Prompt - pip insta
Requirement already satisfied: charset-normalizer<4,>=2 in d:\anaconda\lib\site-packages (from requests>=2.23.0->ultralytics) (3.3.2)
Requirement already satisfied: urllib3<3,>=1.21.1 in d:\anaconda\lib\site-packages (from requests>=2.23.0->ultralytics) (2.3.0)
Requirement already satisfied: markdown-it-py>=2.2.0 in d:\anaconda\lib\site-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (2.2.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in d:\anaconda\lib\site-packages (from rich>=10.11.0->typer<1.0,>=0.12->gradio) (2.19.1)
Requirement already satisfied: mdurl~=0.1 in d:\anaconda\lib\site-packages (from markdown-it-py>=2.2.0->rich>=10.11.0->typer<1.0,>=0.12->gradio) (0.1.0)
Requirement already satisfied: sympy>=1.13.3 in d:\anaconda\lib\site-packages (from torch>=1.8.0->ultralytics) (1.13.3)
Requirement already satisfied: networkx in d:\anaconda\lib\site-packages (from torch>=1.8.0->ultralytics) (3.4.2)
Requirement already satisfied: setuptools in d:\anaconda\lib\site-packages (from torch>=1.8.0->ultralytics) (72.1.0)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in d:\anaconda\lib\site-packages (from sympy>=1.13.3->torch>=1.8.0->ultralytics) (1.3.0)
Using cached ultralytics-8.3.202-py3-none-any.whl (1.1 MB)
Downloading gradio-5.46.1-py3-none-any.whl (60.4 MB)
60.4/60.4 MB 1.5 MB/s eta 0:00:00
Downloading gradio_client-1.13.1-py3-none-any.whl (325 kB)
Downloading aiofiles-24.1.0-py3-none-any.whl (15 kB)
Downloading audioop_lts-0.2.2-cp313-abi3-win_amd64.whl (30 kB)
Downloading fastapi-0.117.1-py3-none-any.whl (95 kB)
Downloading groovy-0.1.2-py3-none-any.whl (14 kB)
Downloading huggingface_hub-0.35.0-py3-none-any.whl (563 kB)
563.4/563.4 kB 3.4 MB/s eta 0:00:00
Downloading orjson-3.11.3-cp313-cp313-win_amd64.whl (131 kB)
Downloading safehttpx-0.1.6-py3-none-any.whl (8.7 kB)
Downloading semantic_version-2.10.0-py2.py3-none-any.whl (15 kB)
Downloading starlette-0.48.0-py3-none-any.whl (73 kB)
Downloading typer-0.19.1-py3-none-any.whl (46 kB)
Downloading websockets-15.0.1-cp313-cp313-win_amd64.whl (176 kB)
Using cached opencv_python-4.12.0.88-cp37-abi3-win_amd64.whl (39.0 MB)
Downloading Brotli-1.1.0-cp313-cp313-win_amd64.whl (358 kB)
Downloading python_multipart-0.0.20-py3-none-any.whl (24 kB)
Downloading ruff-0.13.1-py3-none-win_amd64.whl (13.3 MB)
13.3/13.3 MB 2.5 MB/s eta 0:00:00
Downloading torch-2.8.0-cp313-cp313-win_amd64.whl (241.3 MB)
49.8/241.3 MB 552.9 kB/s eta 0:05:47
High UV
Now
Search
ENG
IN
2.14.14 PM
21-09-2025


Gradio
fdd7f4910bed.ngrok-free.app
image
output
FirstAidBox 0.93
OxygenTank 0.95
OxygenTank 0.98
Flag
Clear
Submit


github.com/sandy-07/Duality-ai-space-station-safety-object-detection-2/tree/main
Duality-ai-space-station-safety-object-detection-2
Public
Pin
Watch 0
Fork 0
Star 0
main
1 Branch
0 Tags
Go to file
Add file
Code
sandy-07 Update app.py
ea83859 · 1 hour ago
11 Commits
.gitignore Initial commit 14 hours ago
000000001_light_unclutter.png Add files via upload 2 hours ago
000000001_vdark_clutter.png Add files via upload 2 hours ago
000000002_dark_clutter.png Add files via upload 2 hours ago
000000033_dark_unclutter.png Add files via upload 2 hours ago
000000033_vdark_clutter.png Add files via upload 2 hours ago
000000035_light_unclutter.png Add files via upload 2 hours ago
LICENSE Initial commit 14 hours ago
README.md Initial commit 14 hours ago
app.py Update app.py 1 hour ago
classes.txt Add files via upload 2 hours ago
predict.py Add files via upload 2 hours ago
train.py Add files via upload 2 hours ago
visualize.py Add files via upload 2 hours ago
yolo_best.pt Update yolo_best.pt 2 hours ago
About
YOLOv8 object detection for space station safety equipment - Hackathon #2 project
Readme
MIT license
Activity
0 stars
0 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package
Languages
Python 100.0%
Suggested workflows
Based on your tech stack
1 cm of rain
Tuesday
Search
ENG
IN
5.07.27 PM
21-09-2025