Object Correspondence in Digital Twin Terrain

Harshvardhan Sivasubramanian (12337780186)
Vijayramsriram Sathanandhan (1233713197)
Teja Vishnuvardhan Boddu (1234122970)

Introduction to Rock Correspondence Detection

Project Aim

Simulate and analyze object correspondence in a known environment using a UAV.

Scenario

The drone identifies and locates specific rocks in a custom simulated world, comparing positions across scenes to infer displacement from hypothetical events like natural calamities.

Relevance

Mimics challenges in planetary exploration such as site monitoring, change detection, and autonomous data logging.

Simulation Environment Creation

Data Collection

Captured high-resolution photos of a real arena from multiple angles to gather visual data for 3D reconstruction.

Photogrammetry with Meshroom

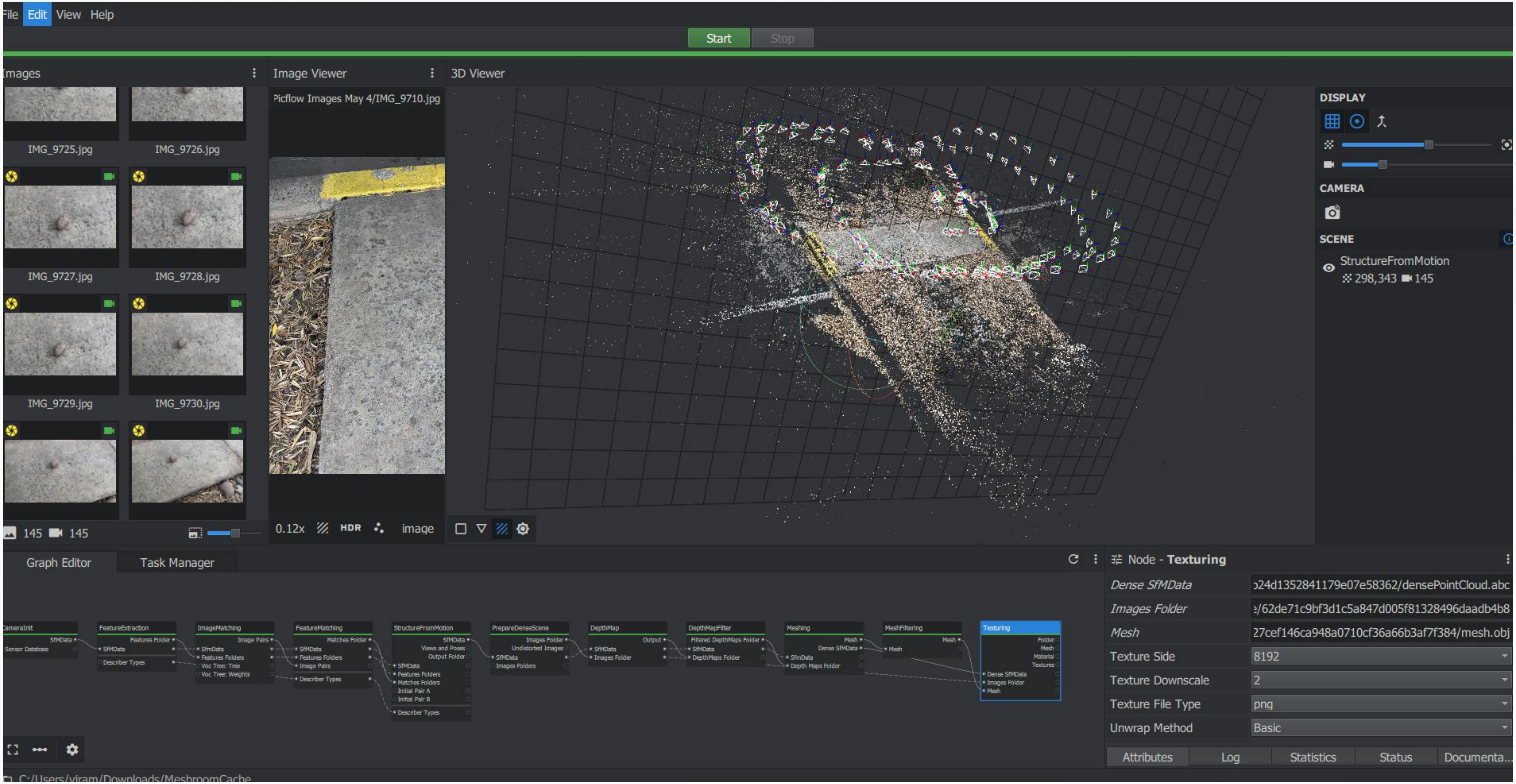
- Imported raw images into Meshroom.
- Generated full 3D textured model (OBJ + textures).

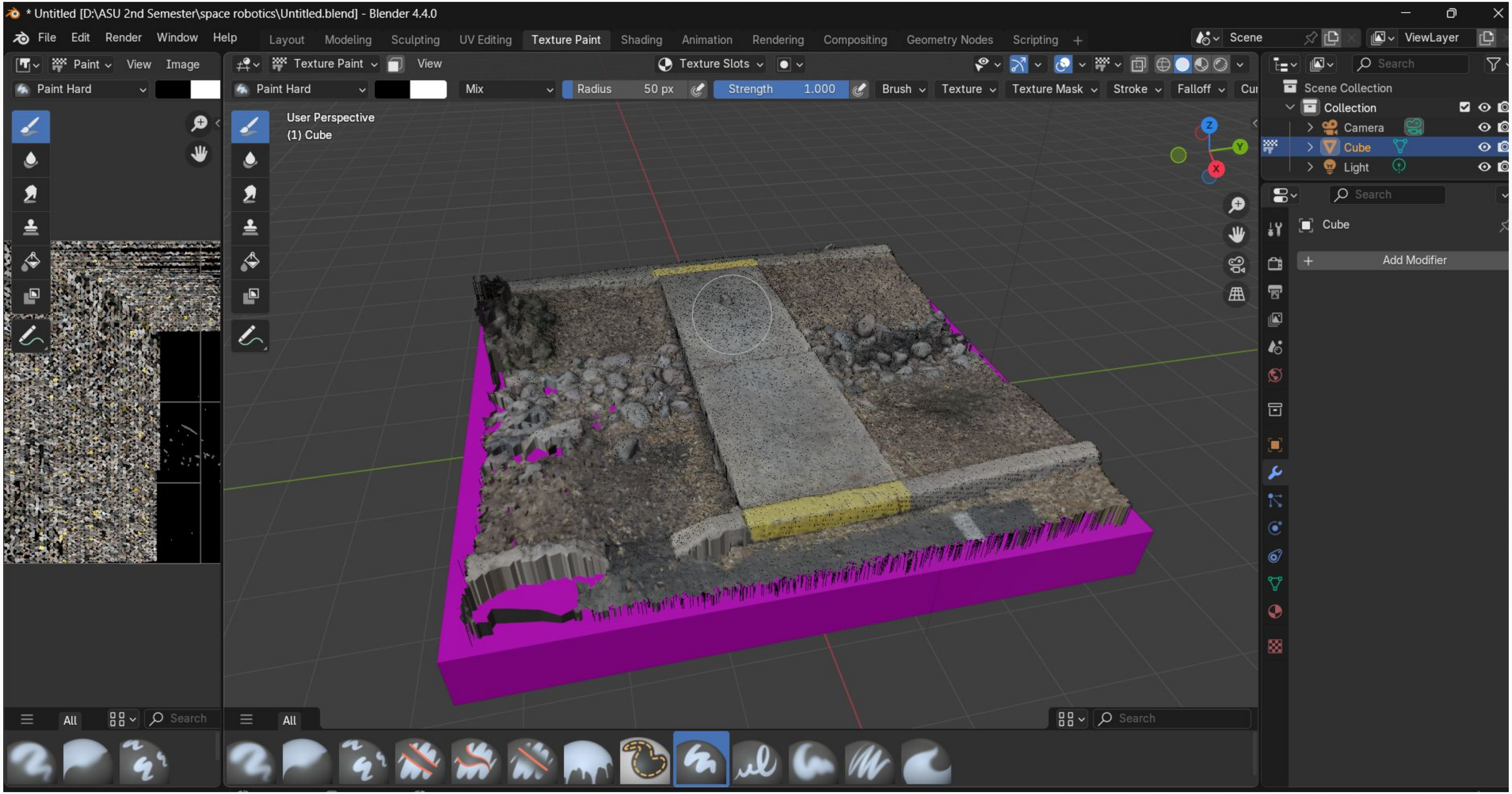
Model Cleanup in Blender

- Removed extraneous geometry.
- Corrected origin, scaling, and texture coordinates.
- Flattened ground surface for drone alignment.

Export to Gazebo

- Exported cleaned model as GLB or DAE.
- Added mesh as static model in SDF format.
- Adjusted pose, lighting, and materials manually.


File Edit View Help
Start Stop
Images
Image Viewer
3D Viewer
IMG_9725.jpg
IMG_9726.jpg
IMG_9727.jpg
IMG_9728.jpg
IMG_9729.jpg
IMG_9730.jpg
145 145
Picflow Images May 4/IMG_9710.jpg
0.12x HDR image
DISPLAY
CAMERA
SCENE
StructureFromMotion
298,343 145
Graph Editor
Task Manager
Node - Texturing
Dense SfMData
Images Folder
Mesh
Texture Side
8192
Texture Downscale
2
Texture File Type
png
Unwrap Method
Basic
Attributes
Log
Statistics
Status
Documenta...
C:/Users/viram/Downloads/MeshroomCache


* Untitled [D:\ASU 2nd Semester\space robotics\Untitled.blend] - Blender 4.4.0
File
Edit
Render
Window
Help
Layout
Modeling
Sculpting
UV Editing
Texture Paint
Shading
Animation
Rendering
Compositing
Geometry Nodes
Scripting
Scene
ViewLayer
Paint
View
Image
Paint Hard
Texture Paint
View
Texture Slots
Mix
Radius
50 px
Strength
1.000
Brush
Texture
Texture Mask
Stroke
Falloff
User Perspective
(1) Cube
Scene Collection
Collection
Camera
Cube
Light
Search
Cube
Add Modifier
All
Search

System Architecture Overview

1. ## Drone Launch Pipeline

   Launches PX4 x500_depth drone, starts Gazebo Sim with custom terrain, bridges odometry and camera topics, and initializes teleoperation and TF broadcaster.

2. ## YOLOv8 Detection Node

   Loads custom YOLOv8 model, subscribes to RGB camera feed, publishes bounding boxes and labels, and visualizes detections in real-time.

3. ## Coordinate Conversion Node

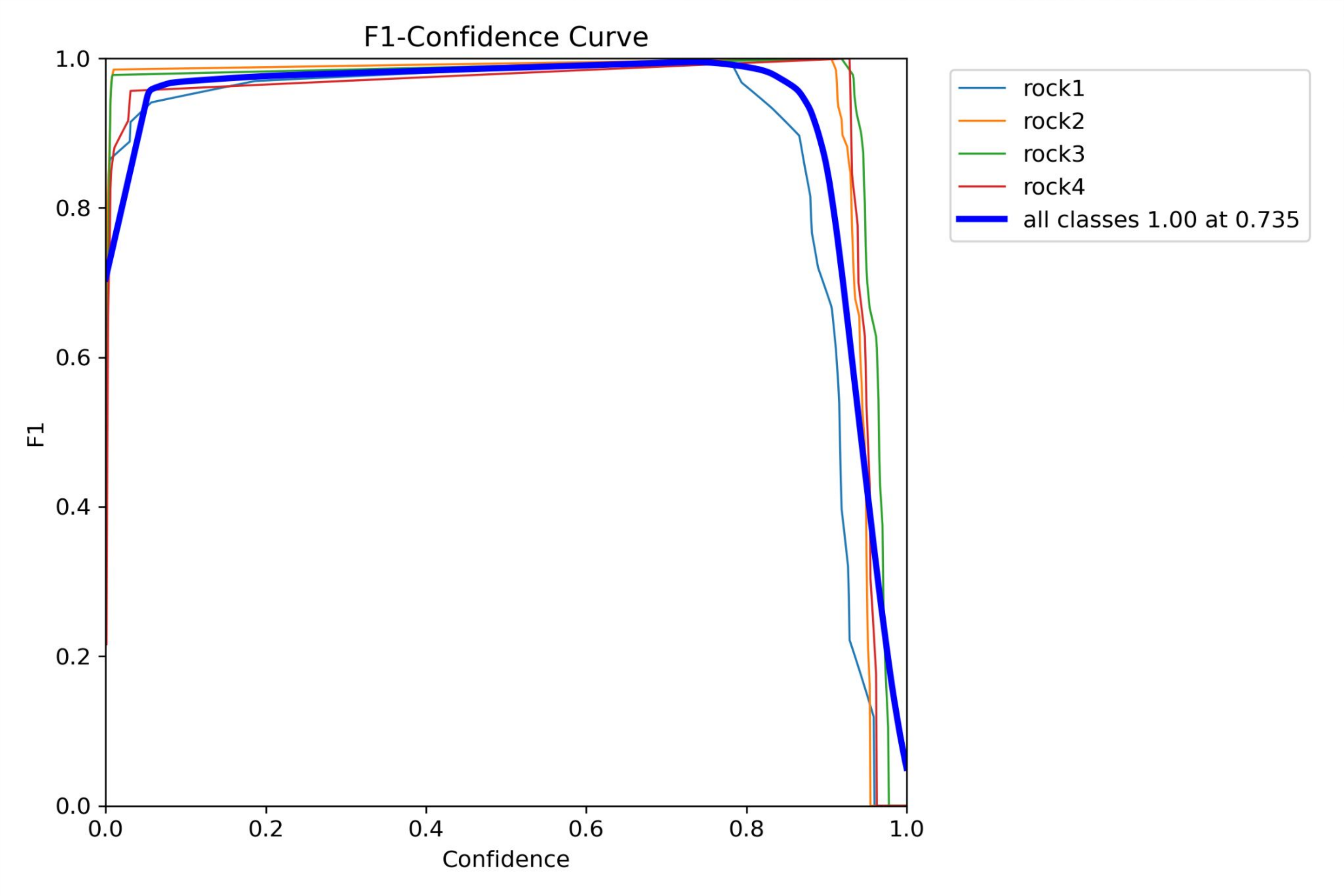
   Calculates 3D world coordinates from bounding boxes and depth data, transforms coordinates using TF2, and logs results in CSV.

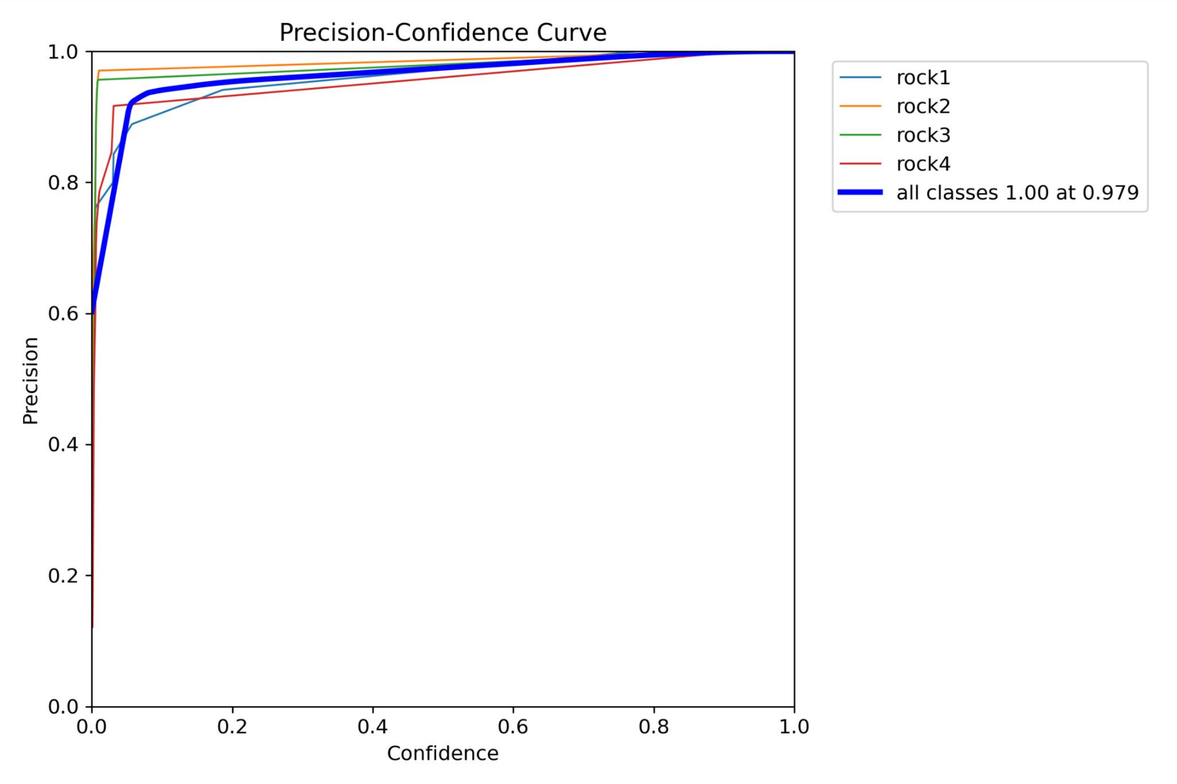
4. ## Odometry TF Broadcaster

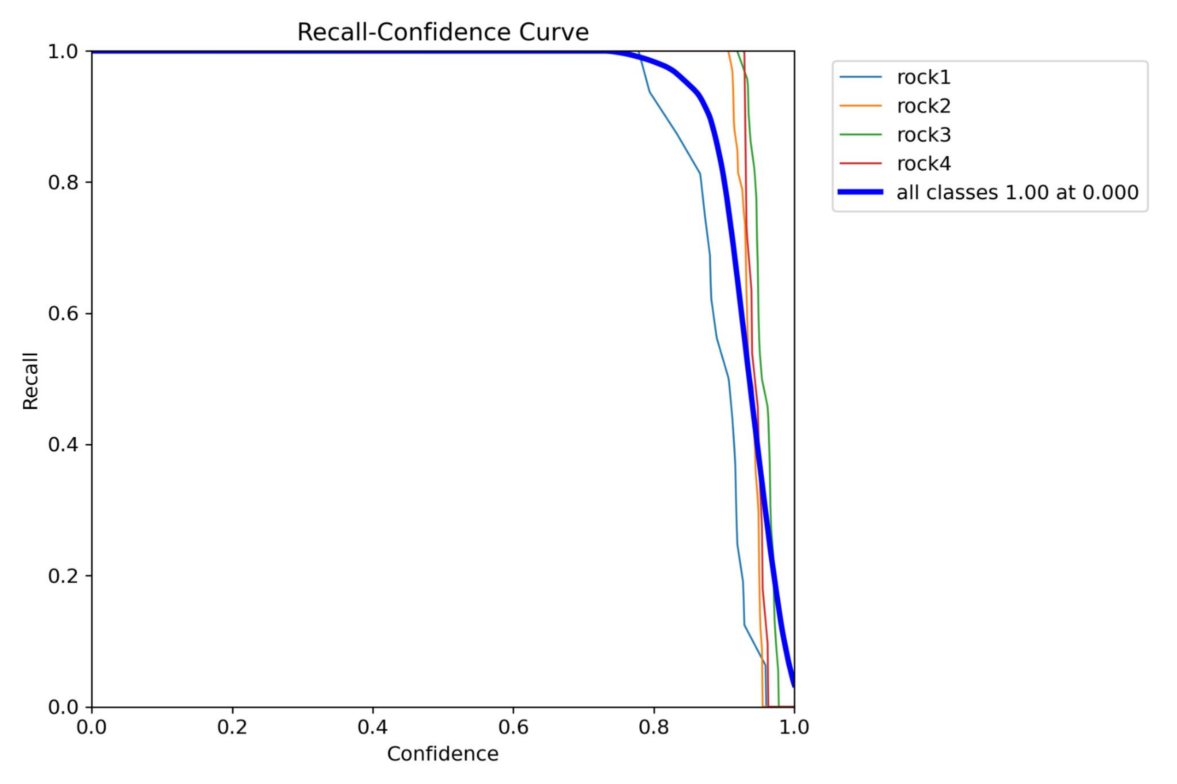
   Publishes dynamic transform from world to base_link using PX4 odometry, enabling global rock localization.

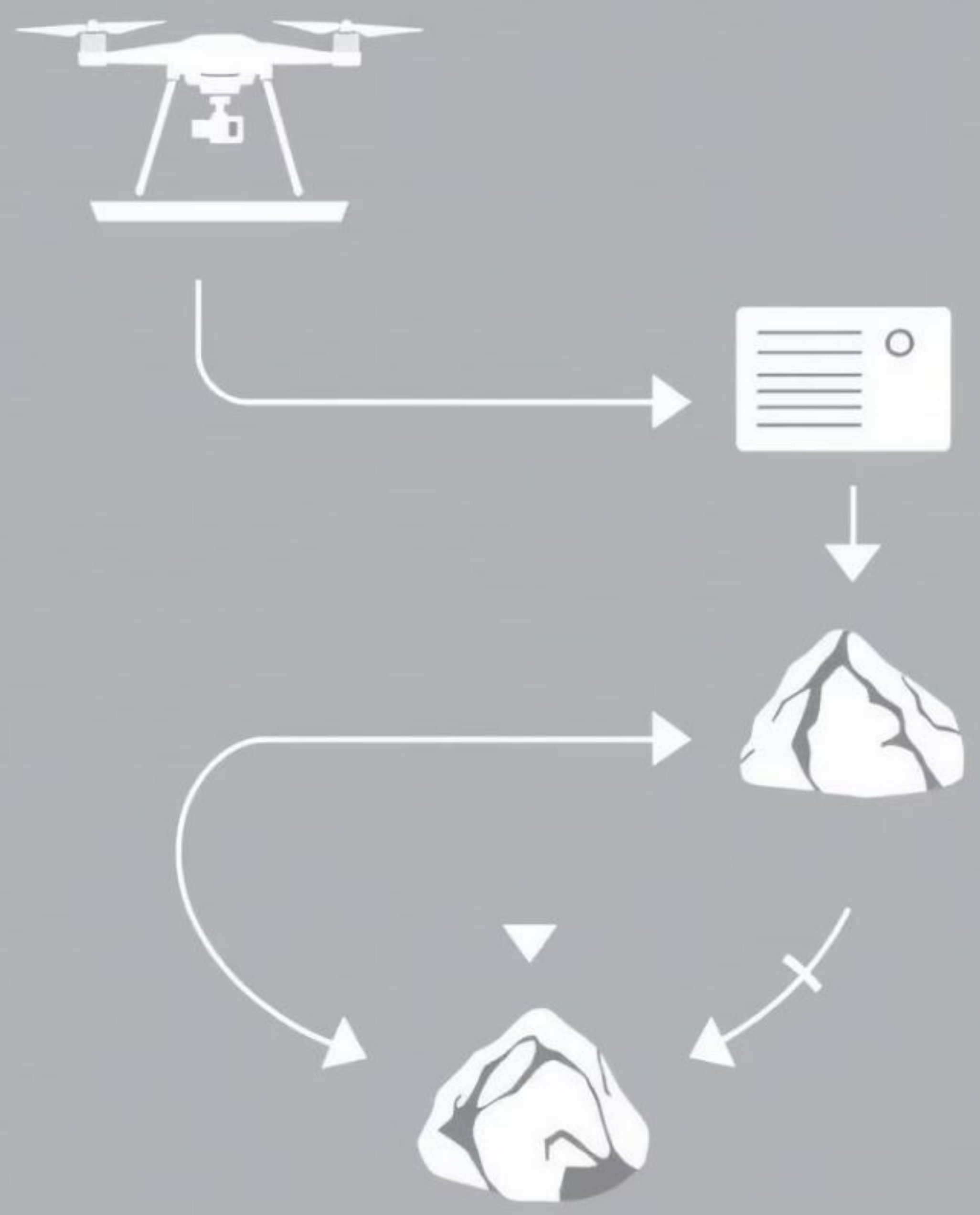
5. ## Keyboard Control

   Maps keyboard inputs to velocity commands and arming signals for drone motion control.


F1-Confidence Curve
F1
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
rock1
rock2
rock3
rock4
all classes 1.00 at 0.735


Precision-Confidence Curve
Precision
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
rock1
rock2
rock3
rock4
all classes 1.00 at 0.979


Recall-Confidence Curve
Recall
Confidence
0.0
0.2
0.4
0.6
0.8
1.0
rock1
rock2
rock3
rock4
all classes 1.00 at 0.000



Data Pipeline Workflow

1 **Launch Environment**

Spawns drone and bridges communication topics for sensors and control.

2 **Detect Rocks**

YOLO identifies rock positions in the RGB camera feed.

3 **Log Positions**

Coordinates are calculated from detection and depth data, then logged.

4 **Compare**

Repeat detection in a second terrain and compare CSV files to analyze displacement manually.

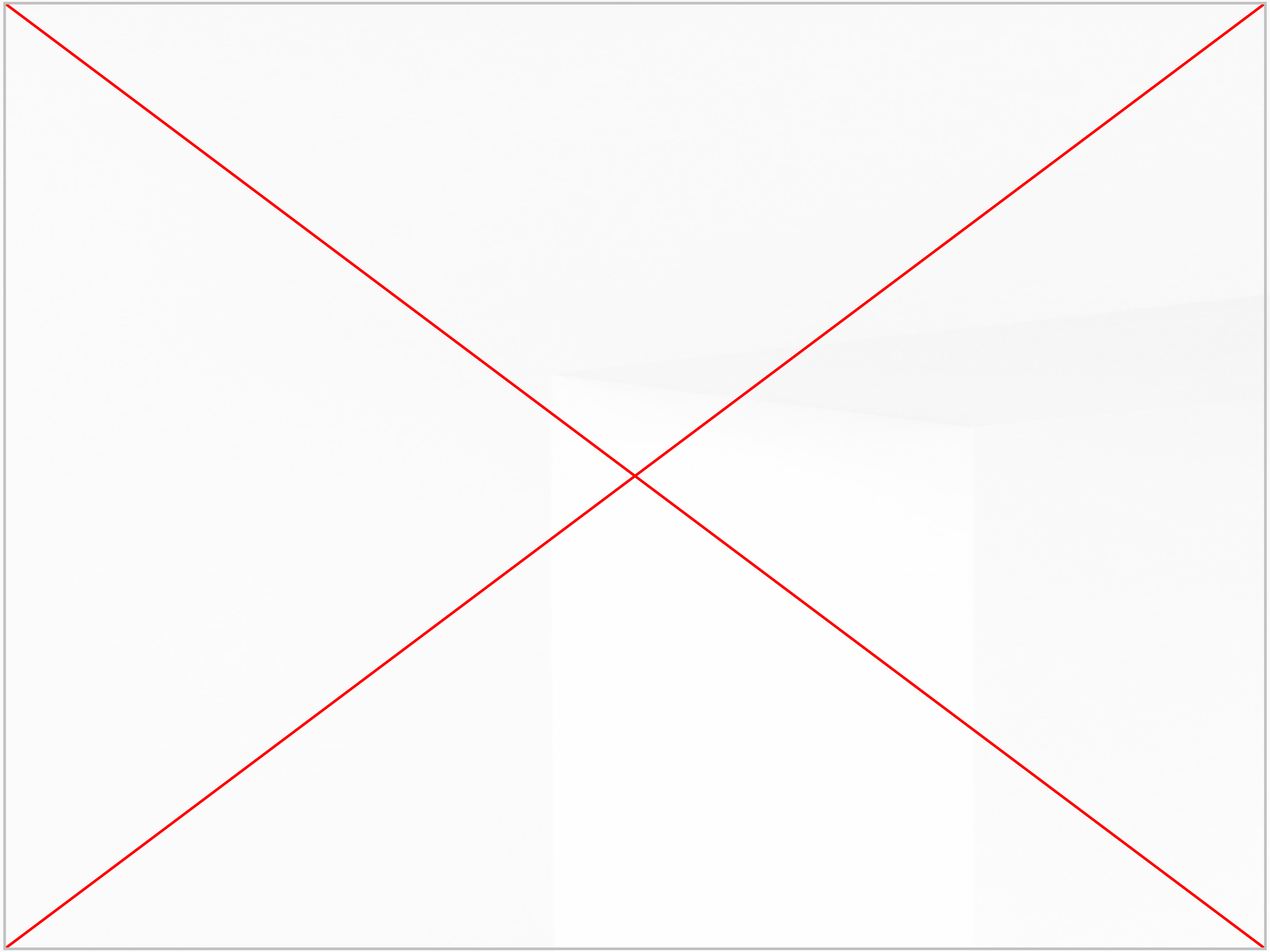

rock2_image_167.png
rock2 0.9
rock3_image_094.png
rock3 0.9
rock4_image_102.png
rock4 0.9
rock4_image_020.png
rock4 0.9
rock3_image_010.png
rock3 1.0
rock4_image_136.png
rock4 0.9
rock4_image_066.png
rock4 0.9
rock4_image_008.png
rock4 1.0
rock3_image_086.png
rock3 0.9
rock4_image_118.png
rock4 0.9
rock4_image_054.png
rock4 0.9
rock3_image_196.png
rock3 0.9
rock3_image_092.png
rock3 0.9
rock4_image_104.png
rock4 1.0
rock4_image_042.png
rock4 0.9
rock3_image_194.png
rock3 1.0

Environment 1

```
timestamp,label,confidence,x,y,z
1746726902.593524,rock4,0.36,-0.601,-1.207,-1.195
1746726902.943148,rock4,0.34,-0.611,-1.254,-1.256
1746726903.053563,rock4,0.30,-0.613,-1.272,-1.282
1746726903.179073,rock4,0.60,-0.621,-1.301,-1.293
1746726903.287671,rock4,0.29,-0.622,-1.327,-1.321
1746726903.411252,rock4,0.31,-0.635,-1.357,-1.348
1746726903.523124,rock4,0.25,-0.635,-1.360,-1.367
1746726903.637731,rock4,0.49,-0.648,-1.385,-1.401
1746726903.761122,rock4,0.28,-0.653,-1.406,-1.424
1746726903.874778,rock4,0.55,-0.664,-1.423,-1.449
1746726903.984471,rock4,0.47,-0.672,-1.432,-1.478
1746726904.223575,rock4,0.26,-0.685,-1.458,-1.526
```

Environment 2

```
1746726580.632126,rock4,0.67,-5.366,-4.415,-5.628
1746726580.730566,rock4,0.64,-5.391,-4.418,-5.620
1746726580.859473,rock4,0.44,-5.415,-4.419,-5.620
1746726580.953918,rock4,0.61,-5.410,-4.422,-5.621
1746726581.079890,rock4,0.62,-5.422,-4.422,-5.632
1746726581.221950,rock4,0.59,-5.422,-4.422,-5.632
1746726581.336422,rock4,0.66,-5.421,-4.422,-5.633
1746726581.451524,rock4,0.67,-5.420,-4.421,-5.635
1746726581.561032,rock4,0.69,-5.409,-4.421,-5.629
1746726581.693008,rock4,0.60,-5.417,-4.420,-5.639
1746726581.784531,rock4,0.69,-5.416,-4.420,-5.640
1746726581.935978,rock4,0.71,-5.396,-4.415,-5.633
1746726582.026807,rock4,0.62,-5.404,-4.413,-5.633
```



Highlights

Multi-Frame Localization

TF2 ensures accurate transformations between camera and world frames.

Depth and Pixel Fusion

Combines depth data with pixel coordinates for real-world scale extraction.

Automatic Camera Intrinsics

Camera parameters handled automatically via /camera_info topic.

YOLOv8 Fine-Tuning

Model trained on labeled multi-class rock data for robust detection.

Real-Time Performance

Operates with low-latency inference.

Results and Discussion

Detection Accuracy

Over 90% accuracy on test images, demonstrating reliable rock identification.

Localization Precision

Visual analysis estimates localization error between 0.1 and 0.2 meters.

Environment Loading

Gazebo loads optimized GLB terrain in under 15 seconds for efficient simulation.

Displacement Tracking

Successfully tracked rock displacement of approximately 0.4 meters across different terrains.

Project Limitations

Odometry Drift

Current system does not correct odometry drift due to lack of SLAM integration.

Static Transform Assumption

Assumes fixed camera-to-drone transform, which may limit accuracy in dynamic conditions.

Manual Rock Correspondence

No automated matching of rocks between scenes; requires manual comparison.

Mesh Simplification

Large terrains require mesh simplification to maintain performance.

Future Work and Enhancements

1. ## SLAM Integration

   Incorporate ORB-SLAM3 or Cartographer for precise odometry correction.

2. ## Automated Rock Matching

   Implement Nearest Neighbor algorithms to automate rock correspondence from CSV data.

3. ## Calibration Markers

   Use ArUco markers to improve calibration accuracy.

4. ## Real-World Testing

   Extend system to physical drone platforms with onboard computation.

5. ## 3D Point Cloud Analysis

   Apply CloudCompare or similar tools for full-scene displacement analysis.

Conclusion

This project demonstrates a full object correspondence pipeline using a simulated PX4 drone, YOLOv8 detection, and photogrammetry-based terrain. It successfully bridges vision-based AI models with robotics simulation, laying a foundation for real-world robotic exploration and monitoring tasks. The approach highlights the potential for autonomous site monitoring and change detection in planetary exploration and other applications.