PROJECT PANOPTES

COTS drone photogrammetry
+ real-time data fusion

USS HORNET DEFENSE HACKATHON

NOV 2024

COMPETING TRACK

Track 1: Autonomy & Counter-UAS
Determining a GPS position of object using only the camera from FPV drone

Sponsor tech used
CodeMetal, Codeium

Members
Yuxi Liu, Max Wilson, Yoyo Yuan, Atharva Patel, Suhas Palawala, Abhijith Varma Mudunuri

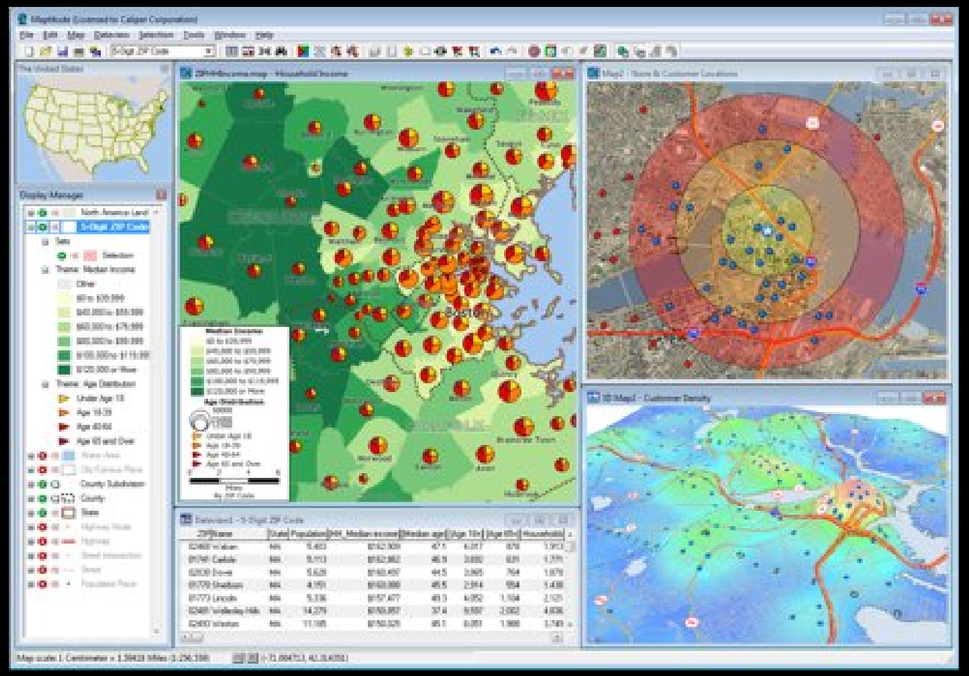
GitHub

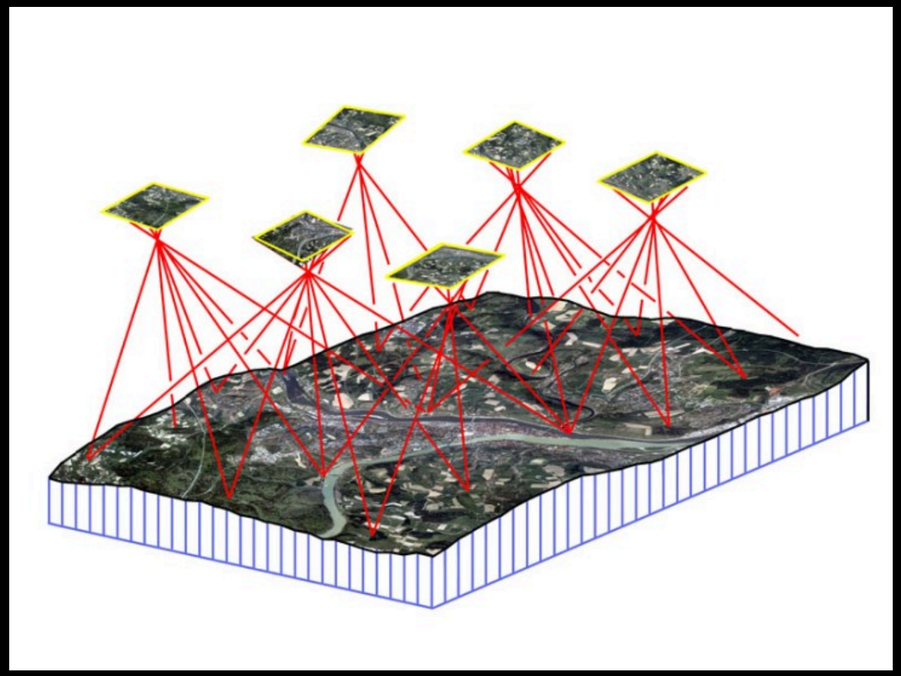
PROBLEM

On the modern, dynamic battlespace, **accurate, real-time information** is critical for effective operations. Drones have been deployed to good effect for surveillance, but they still rely mainly on human operator photo-interpretation, a critical slow-down in the observe-orient part of the OODA loop.

High-accuracy drone photogrammetry can accurately map objects on the ground. However, such systems are typically expensive and fragile, unsuited for the chaos of frontlines.

SOLUTION OVERVIEW

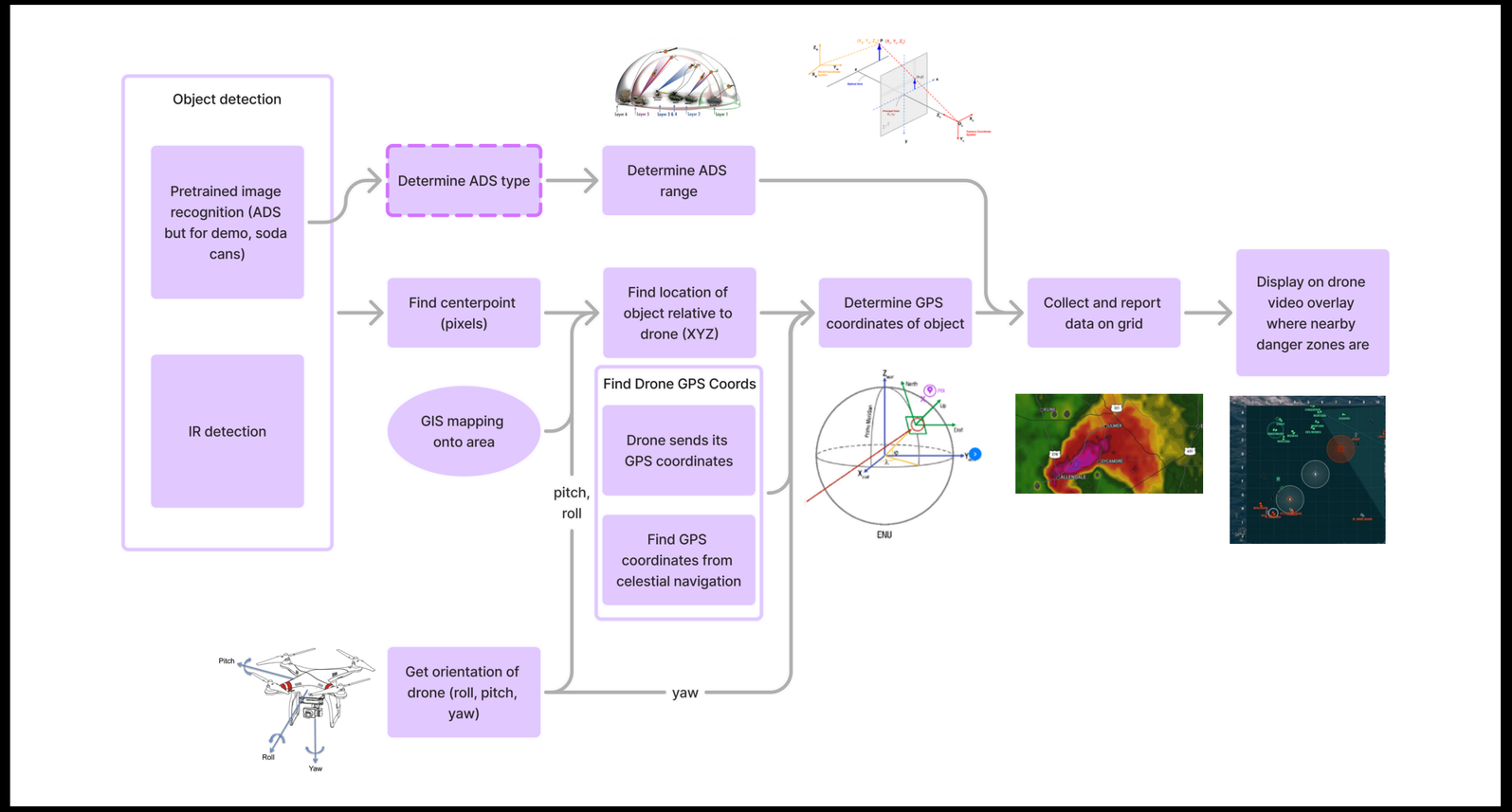


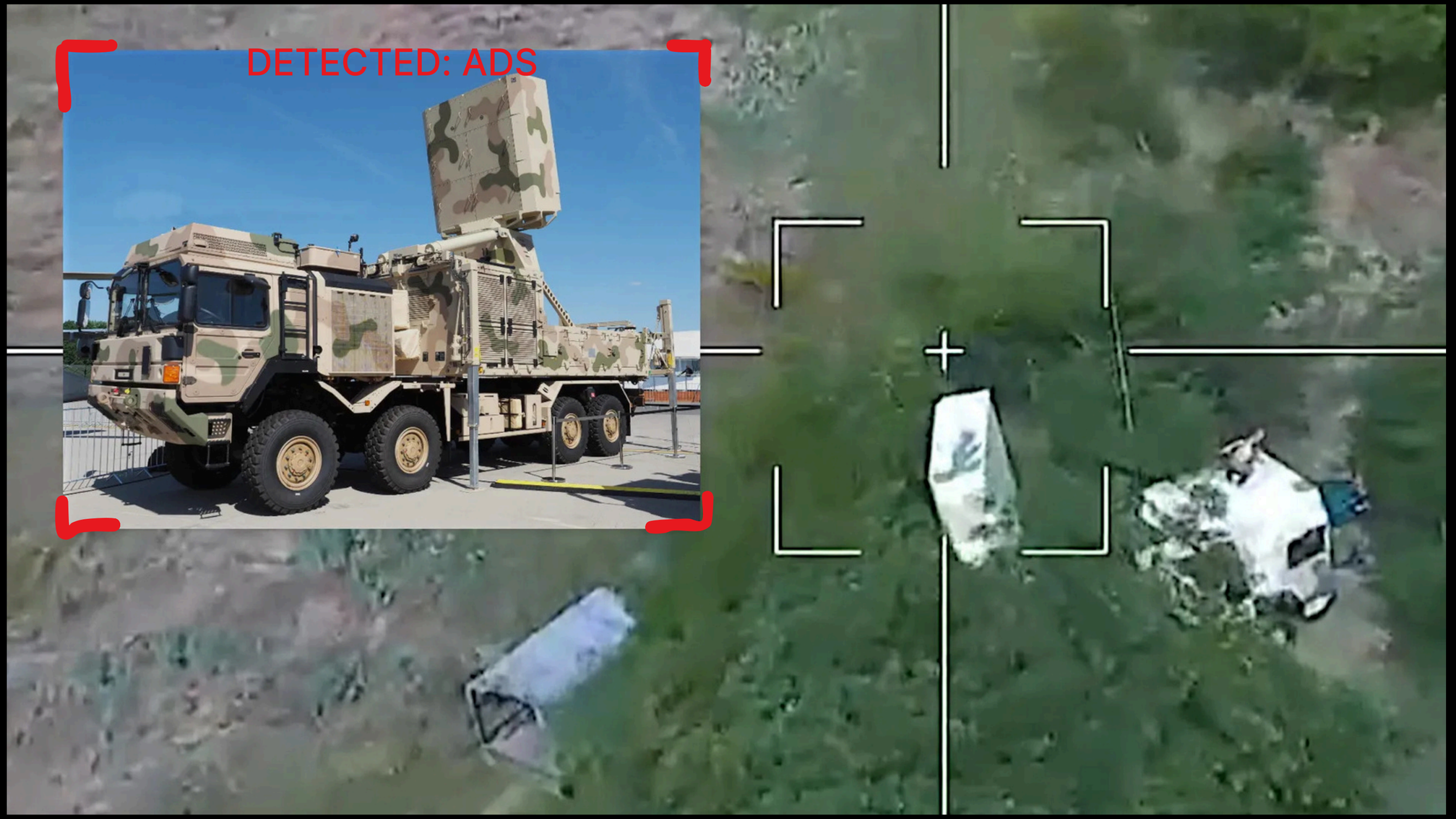


We implemented technologies to **automate** drone positioning with **low-costs.**

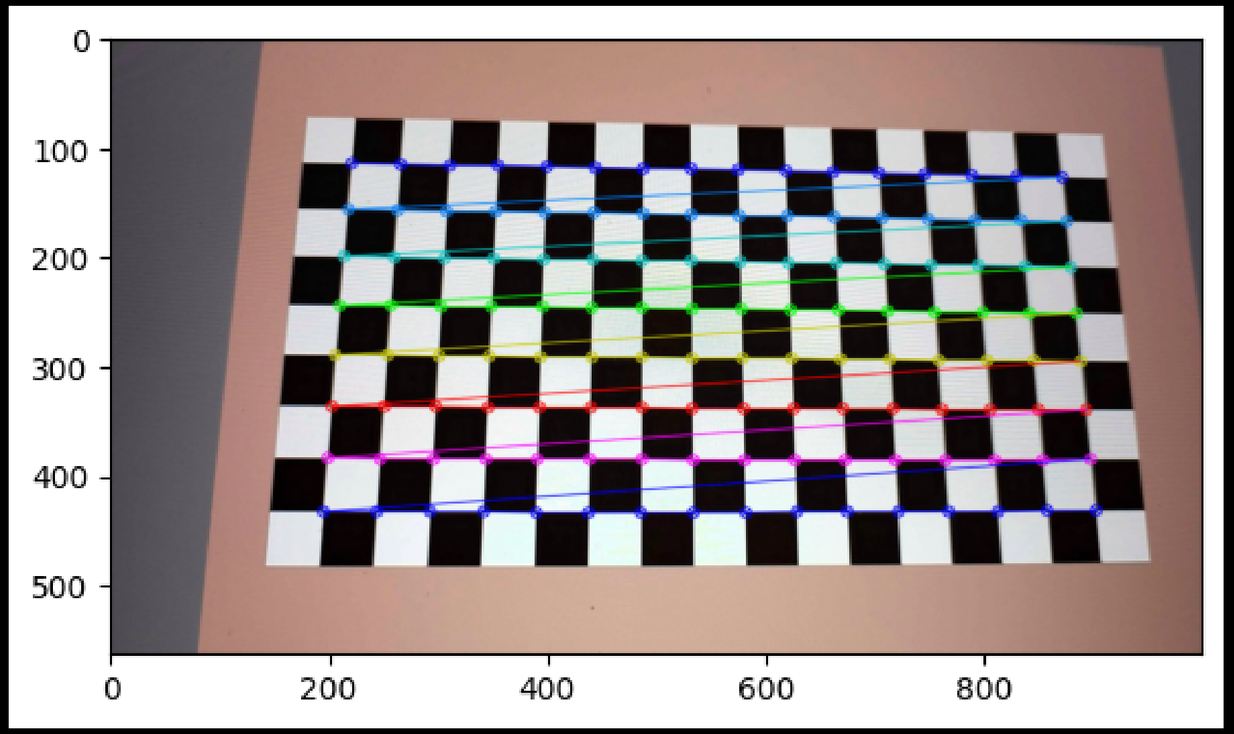
1. COTS drone photogrammetry
2. Object of Interest detection-localization with open-source computer vision
3. battlespace GIS display (“the Overseer”)
4. Also, celestial navigation as backup to GPS.

FLOWCHART


Object detection
Pretrained image recognition (ADS but for demo, soda cans)
IR detection
Determine ADS type
Find centerpoint (pixels)
GIS mapping onto area
Get orientation of drone (roll, pitch, yaw)
Pitch
Roll
Yaw
pitch, roll
yaw
Determine ADS range
Find location of object relative to drone (XYZ)
Find Drone GPS Coords
Drone sends its GPS coordinates
Find GPS coordinates from celestial navigation
Determine GPS coordinates of object
ENU
Collect and report data on grid
Display on drone video overlay where nearby danger zones are

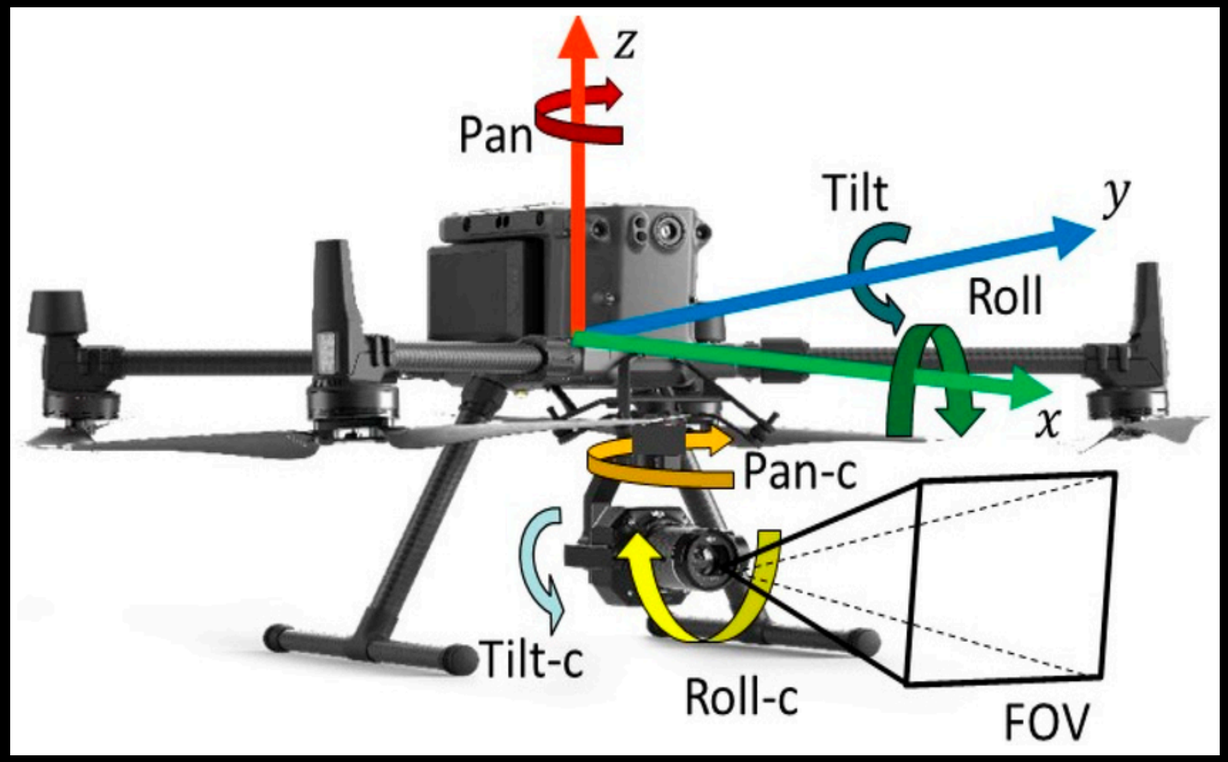

DETECTED: ADS

COTS PHOTOGRAMMETRY

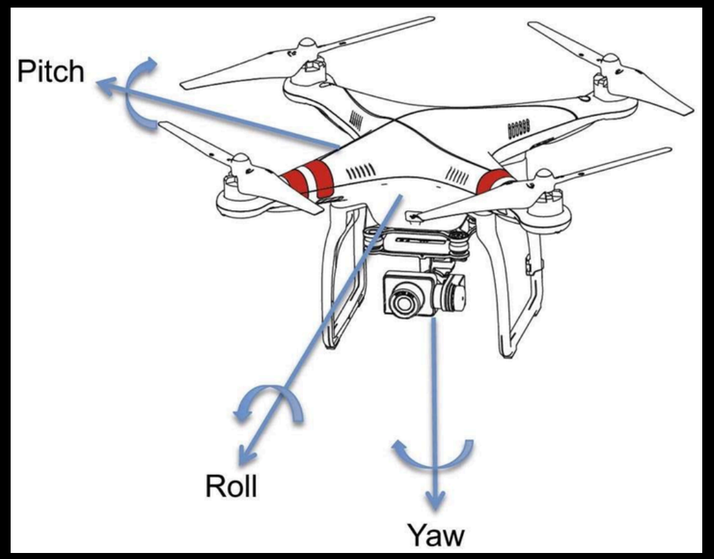


1. Pre-flight on-drone camera calibration with a single checkerboard pattern

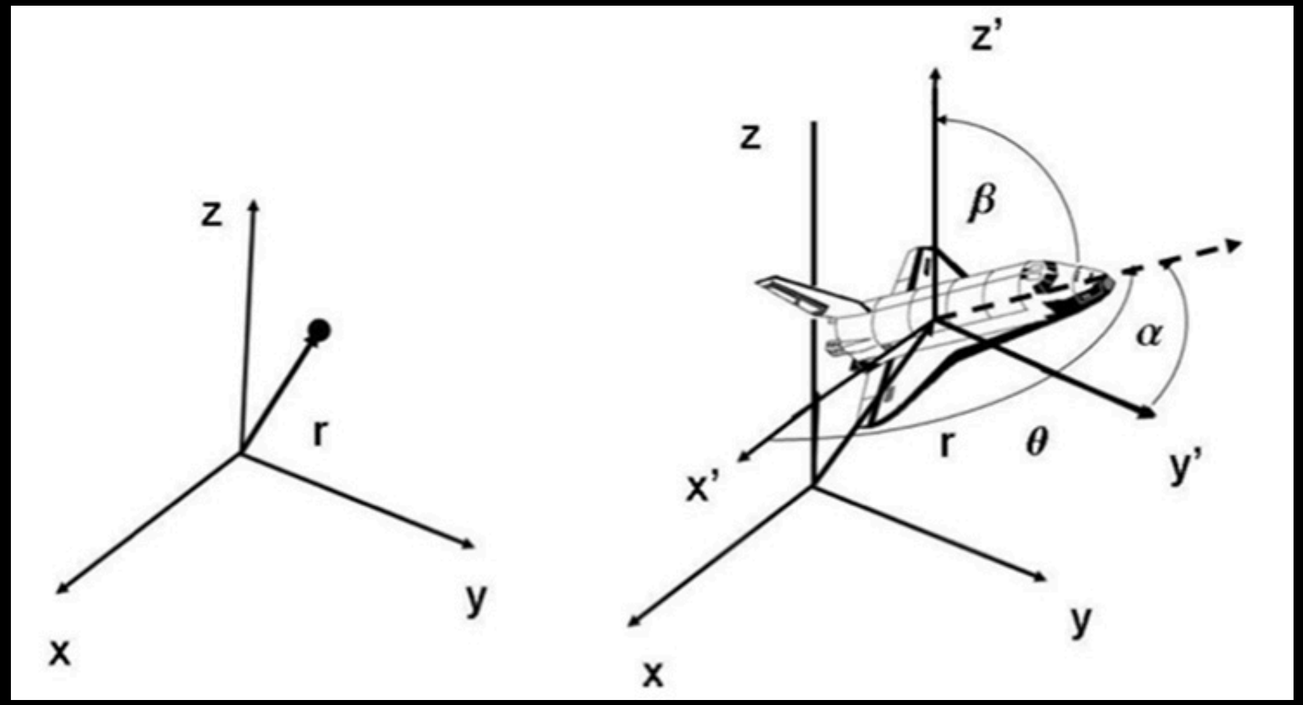
COTS PHOTOGRAMMETRY



Calibration corrects for misalignment between camera coordinate frame and drone frame.

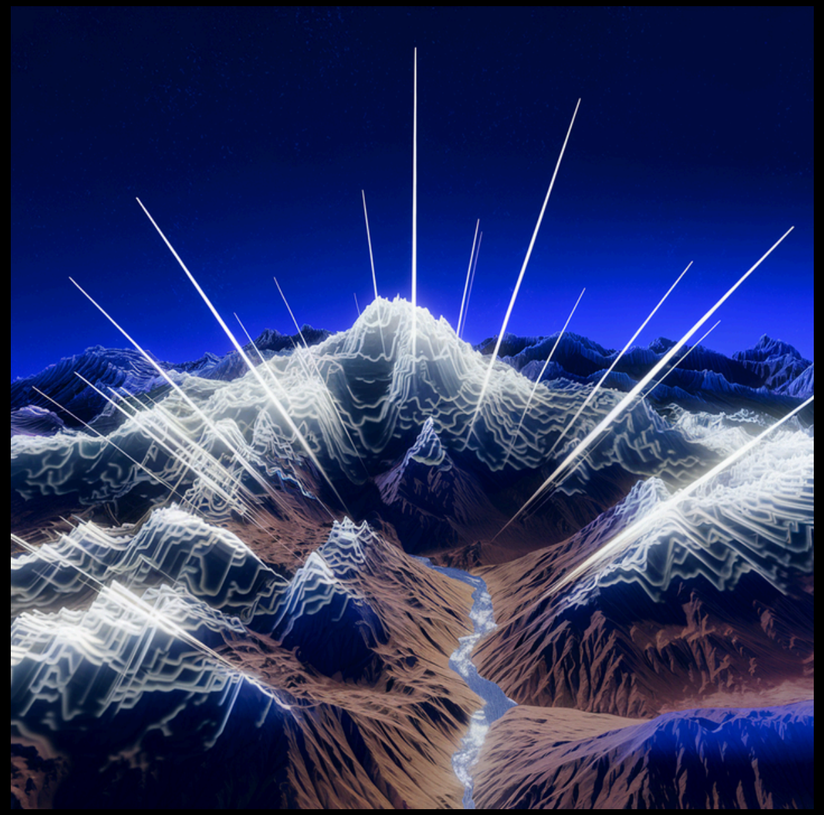


COTS PHOTOGRAMMETRY



2. After calibration, any photo taken by the drone is tagged with its position and orientation (6 Degrees of Freedom)

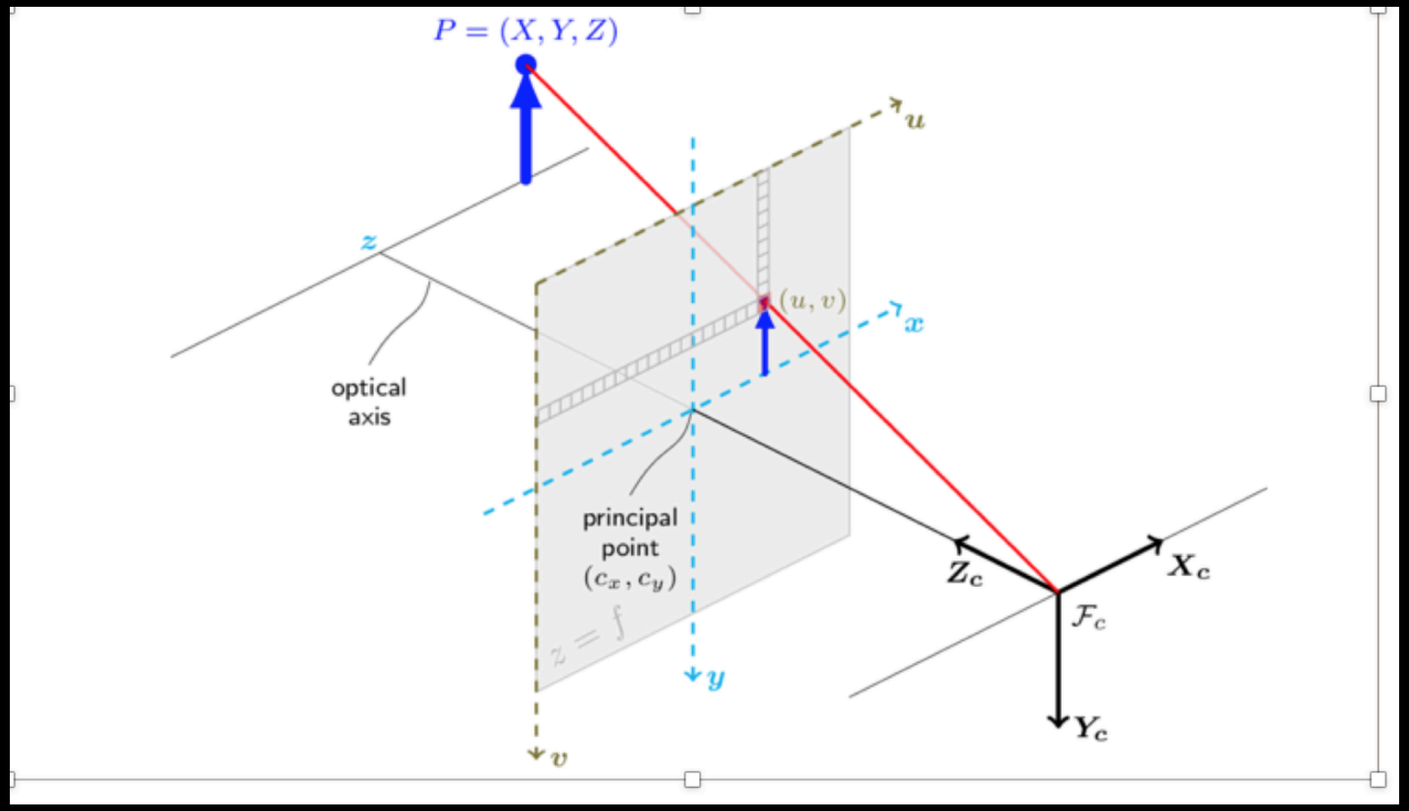
COTS PHOTOGRAMMETRY



3. This is integrated with a GIS, so that each photo is mapped to a region on the terrain

- back-projecting the pixels into rays,
- then intersecting the rays with the ground surface.

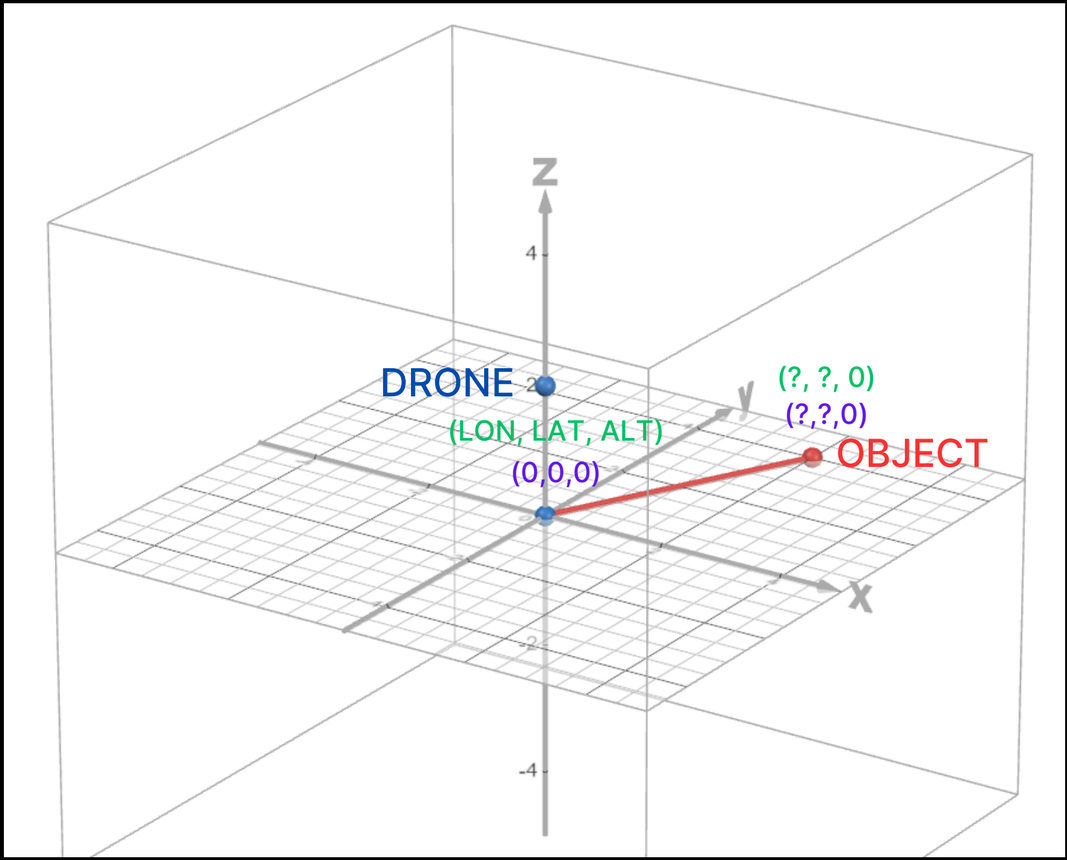
COTS PHOTOGRAMMETRY



Technical details:

1. Use photographic back-projection to convert pixel-coordinates on a photo to ray directions in the drone's coordinate system

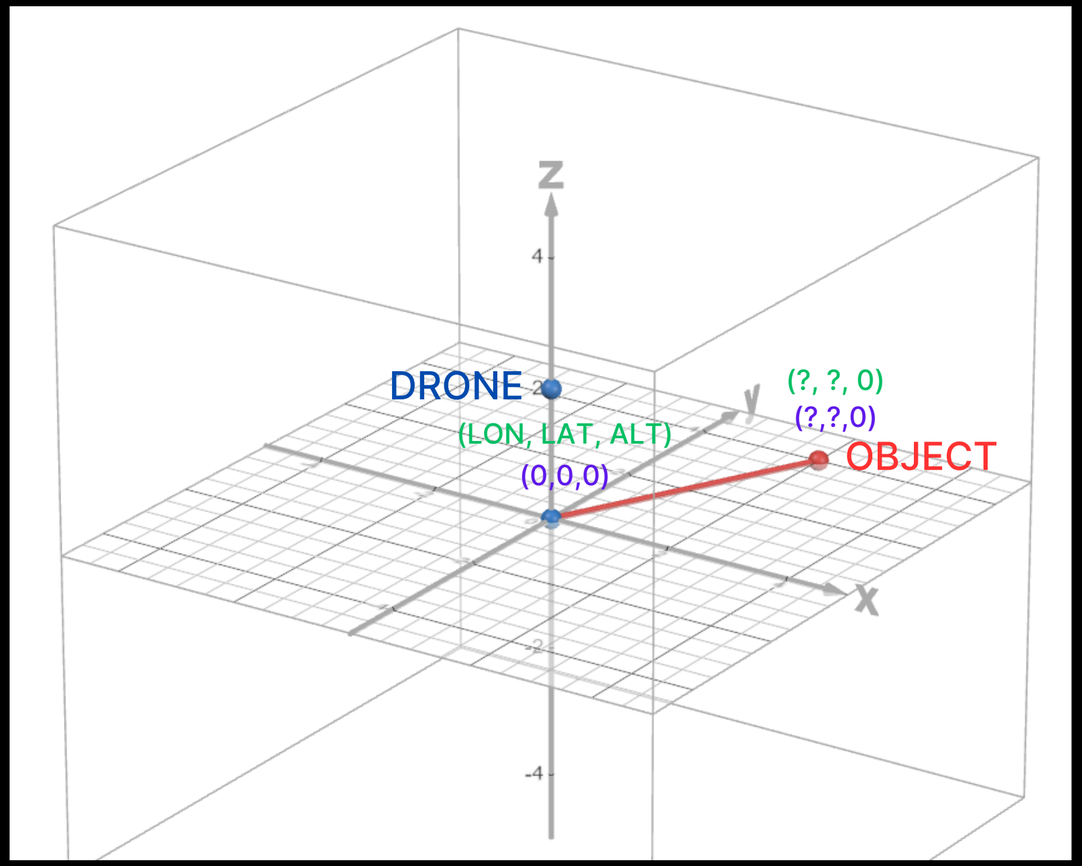
COTS PHOTOGRAMMETRY



Technical details:

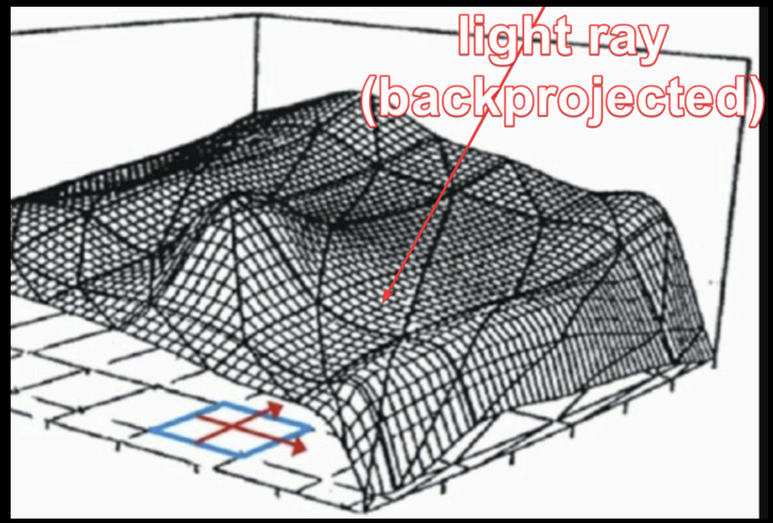
2. Use the drones 6DOF to convert the ray directions in the drone's frame to the local Cartesian frame (North, West, Up)

COTS PHOTOGRAMMETRY

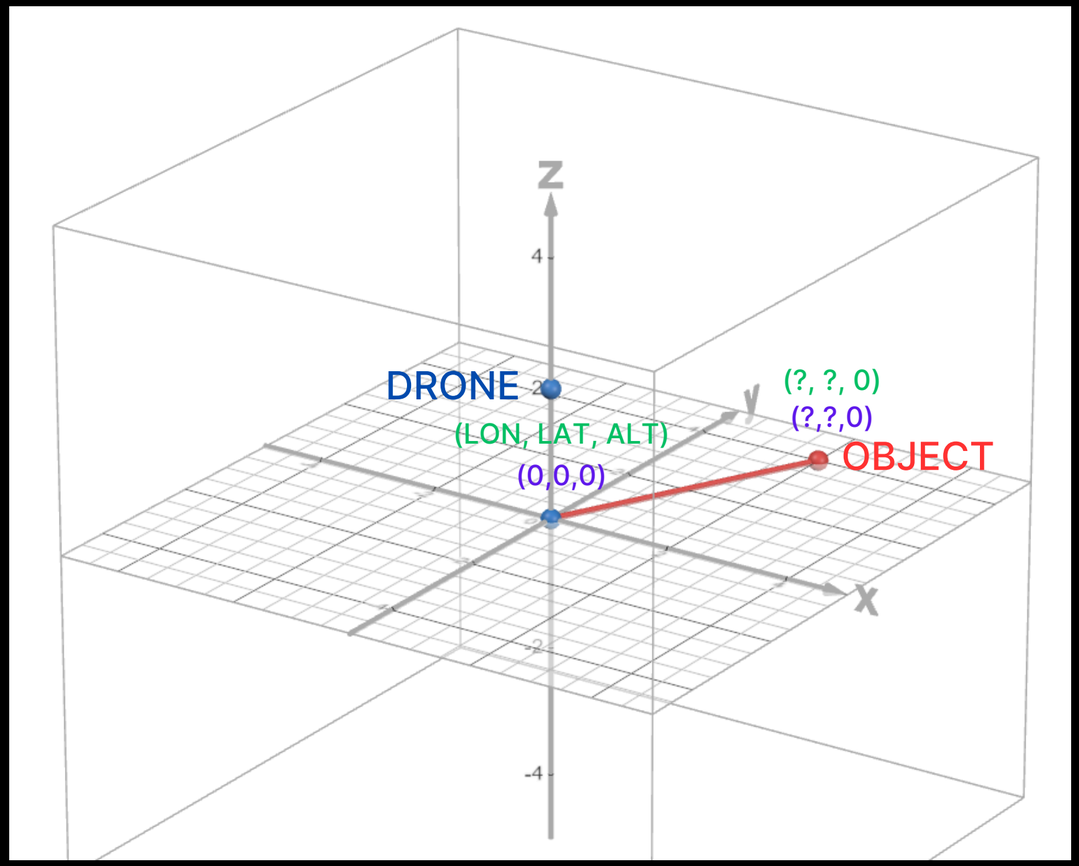


Technical details:

3. Use a GIS to solve for the intersection of the light ray with the polygonal model of the terrain.

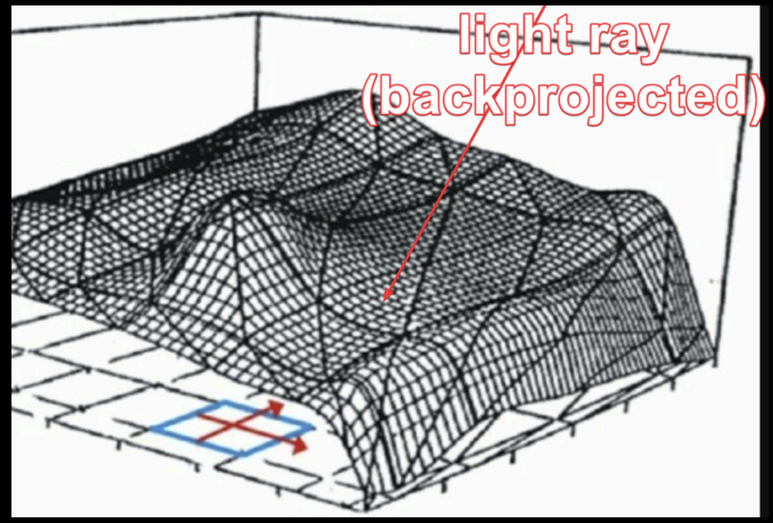


COTS PHOTOGRAMMETRY

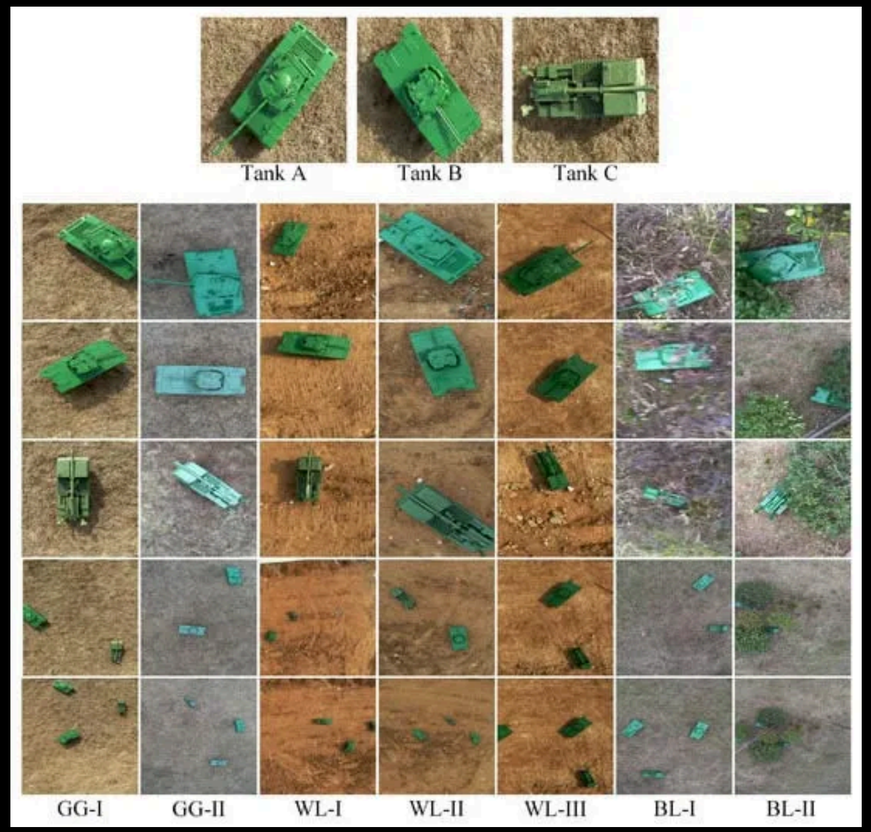


Technical details:

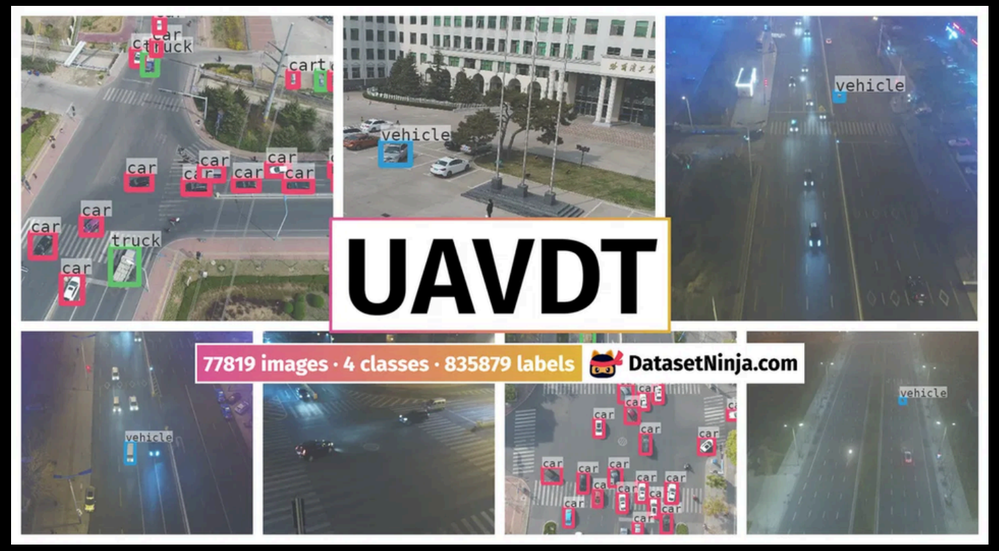
4. Theoretical precision ~4 meters (~1° orientation precision * ~100 m drone height + ~2 m positional precision)



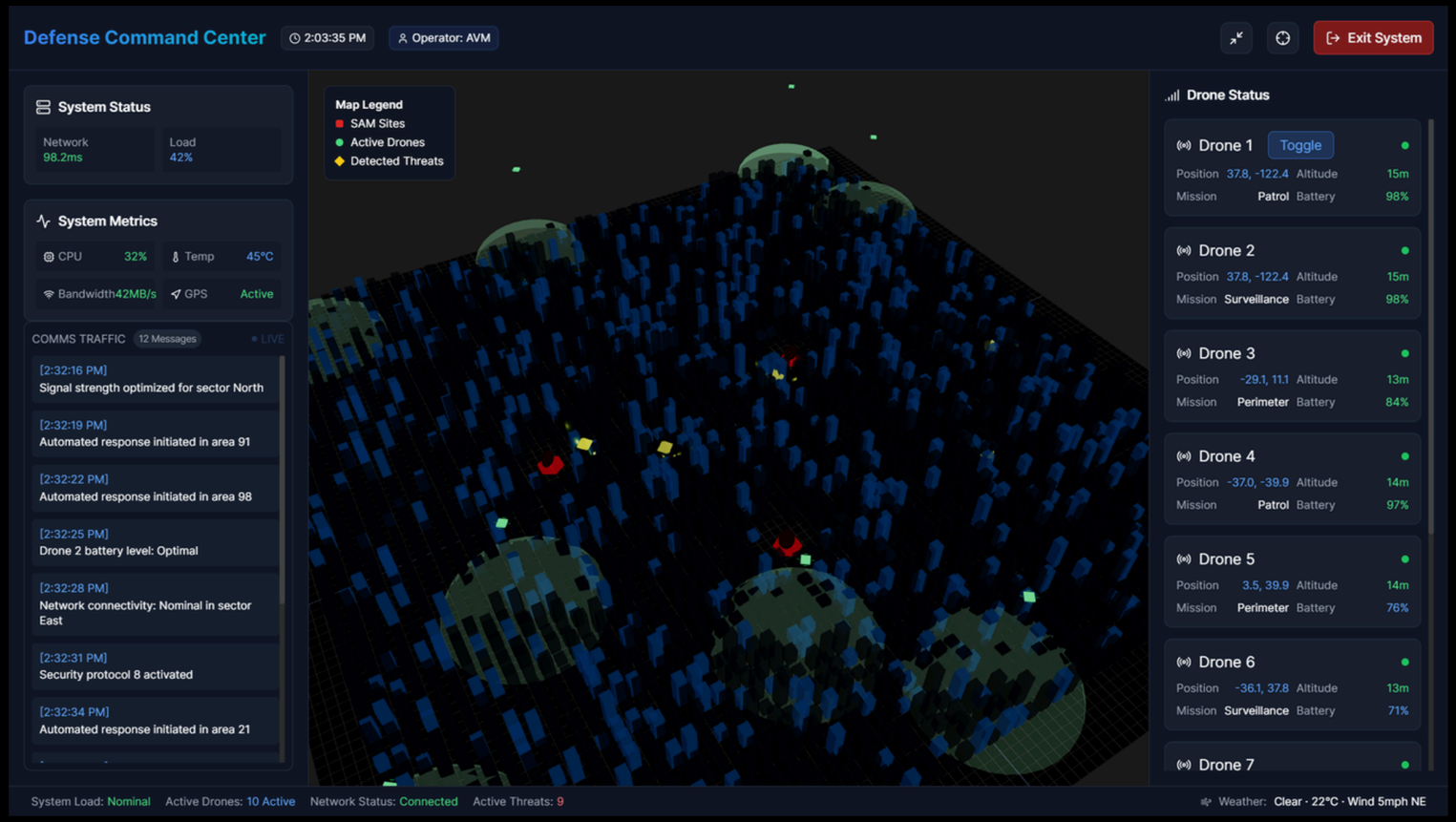
OBJECT DETECTION-LOCALIZATION



Standard open-source neural network (YOLOv5n) to detect and localize objects in images.
Picked since it’s near-SOTA on UAV benchmarks.
(My team member has this code locally)



THE OVERSEER



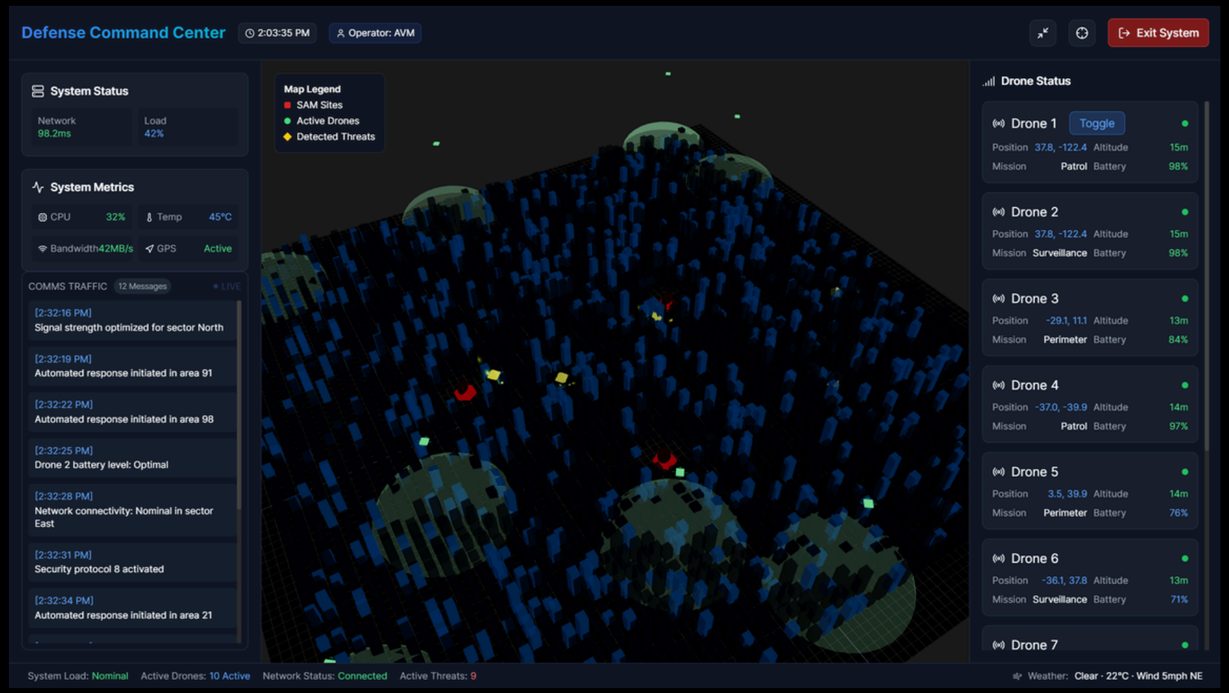
Real-time updated map fusing data from multiple sources.

- Known objects of interest (OoI)
- Unconfirmed OoI found by drone

Operator can click on OoI to display drone photos and footage for the OoI.

Operator can confirm or disconfirm the OoI.

THE OVERSEER



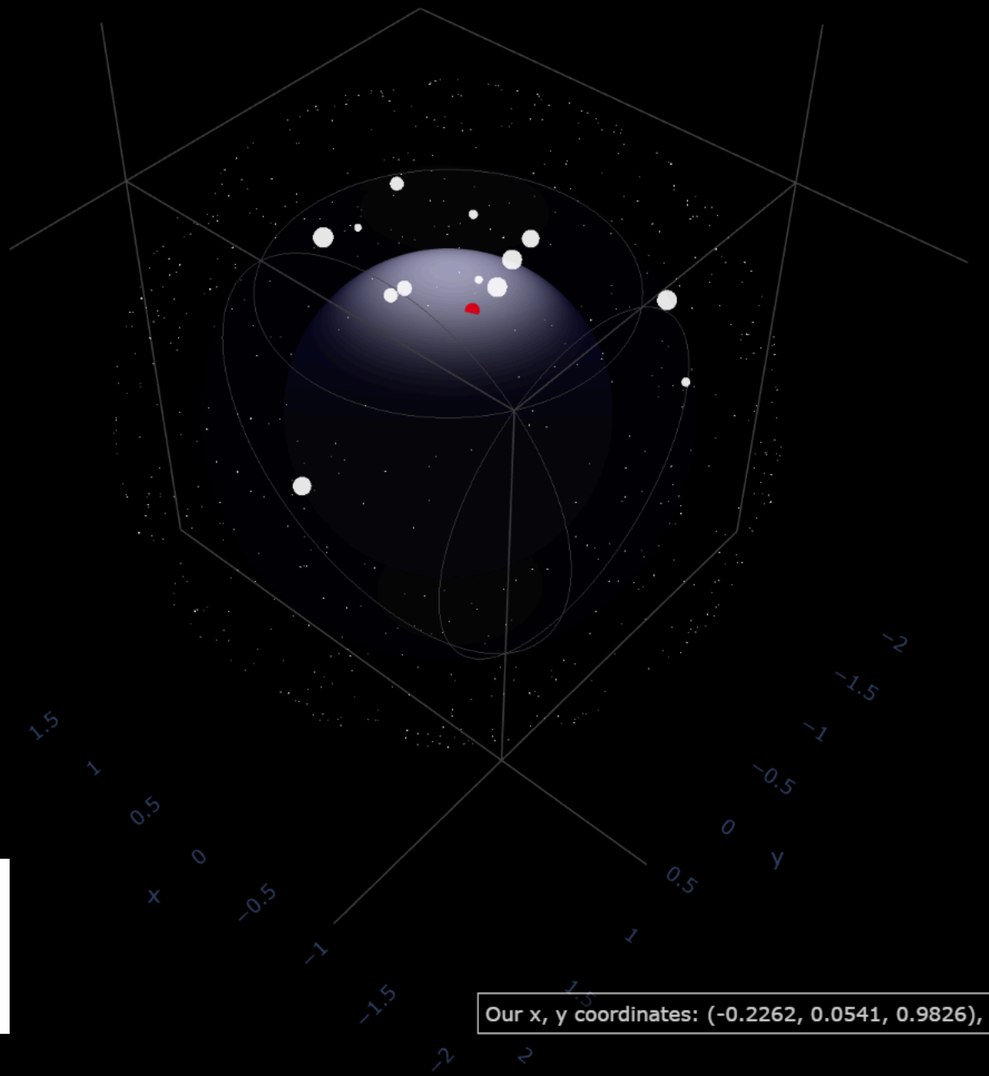
- Modern frontend architecture using Next.js, Three.js for 3D rendering, and React
- Responsive UI powered by Tailwind CSS and ShadcN components
- Real-time data processing and grid visualization system
- Node.js backend supporting high-performance data streams
- Custom-built data transformation engine and generator
- Trained yolov5 on Ukrainian war photos

CELESTIAL NAVIGATION

A backup, satellite-free GNSS.

We just need an image of the sky, and a time point to locate the coordinates of the drone.

Celestial navigation example



1. take picture of sky

2. outline the stars

3. look up pattern in star catalogue

“orion”

5. get local coordinates

HA = LST - RA

= GST + longitude - RA

6. solve altitude equations

$$\sin(Alt)=\sin(\phi)\sin(\delta)+\cos(\phi)\cos(\delta)\cos(HA)$$
$$\sin(Alt)=\sin(\phi)\sin(\delta)+\cos(\phi)\cos(\delta)\cos(HA)$$
$$\sin(Alt)=\sin(\phi)\sin(\delta)+\cos(\phi)\cos(\delta)\cos(HA)$$

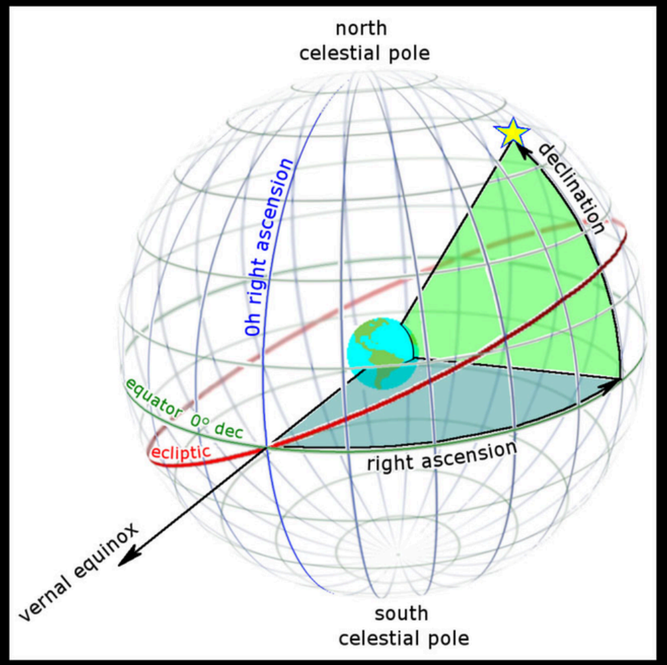
“orion” - betelgeuse

“andromeda” - alpheratz

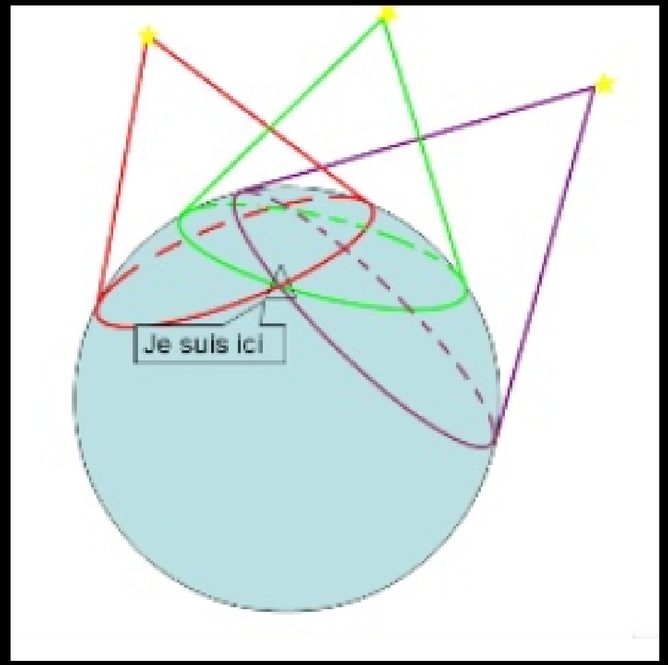
“sagittarius” - nunki

*unknowns

Equatorial coordinates: (r, ra, dec)



Triangulation



USER JOURNEY

1. Pre-flight, calibrate each drone with a single press.
2. Launch drones.
3. Geo- and pose-tagged photos and footage stream back to ground server.
4. Ground server detects and locates objects of interest (OoI) by computer vision + photogrammetry.
5. Drone operators use the Overseer to monitor and dis/confirm OoI in real time.
6. Tactical units use the Overseer for planning.