



# 05/02/24

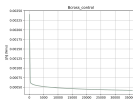
Demonstrating Visual-Inertial A&OD & On-Orbit Edge Computing

# Progress summary

87 days before May 1st

## Updates

- Vision:
  - Data Pipeline automated
  - YOLO Training initial result
  - LD net inference integrated on Jetson
- Estimation:
  - Integrated Transformations and Static batch optimizer module into Vision Inference pipeline
- Testbed:
  - V1 testbed complete setup
  - Logging library for Jetson stats
- Simulator:
  - Preliminary MC analysis for B-cross control completed
- Completed integrated testing for V1



## Blockers

- Camera finalization (avionics?)

## Weekly Plan

- Vision
  - Develop YOLO training plan & retrain LD models
  - Get high-performing LD model for one region w/o overfitting for V2 testbed
  - Complete integration of RC + LD inference on Jetson
- Estimation
  - Evaluate estimation accuracy of batch optimizer module
  - Validate the accuracy of the transformations module by comparing with ground truth.
  - Conduct rigorous integration testing to determine performance of the estimation pipeline

## Interface dependencies

- Intra communication between Jetson and Pycubed
- Driver development for IMU, GPS, magnetorquer h-bridge

# Vision

- **Dataset**

- Data Pipeline (Image download + landmark annotation + convert to Yolo format) automated via shell script on workstation

- **Training**

- **RC:**
  - Training efficientnet-b0 with multiple different hyperparameters, image sizes, dropouts etc.
- **LD:**
  - Trained YOLO models of various sizes on data
  - Issue making detections on val set, likely because landmark boxes are too small/close together + val set too small, mAP close to zero after 100 epochs
  - Future training will be summarized here -> [YOLO Training Report](#)

- **Jetson**

- LD inference class integrated
- Testing script implemented for running inference through sample images + basic accuracy test
- Sample inference data + pretrained 17R\_nadir.pt model for V1

# Automation Script

```

• (myenv) (base) argus-vision@argus-vision:~/vision/VisionTrainingGround/DataPipeline$ ./run_pipeline.sh -h
Usage: ./run_pipeline.sh [options]

Options:
-b BOUNDS      Geographic bounds (format: 'minLon minLat maxLon maxLat'). Default: '-84 24 -78 32'
-i IDATE       Initial date (format: YYYY-MM-DD). Default: '2020-05-01'
-f FDATE       Final date (format: YYYY-MM-DD). Default: '2023-12-31'
-l LANDSAT     Landsat version. Default: 8
-m MAXIMS      Maximum number of images. Default: 50
-s SCALE       Scale. Default: 150
-w BOX_WIDTH   Width of the boxes. Default: 100
-n BOX_COUNT   Number of boxes. Default: 1000
-o OUTPATH     Final output path. Default: '/home/argus-vision/vision/VisionTrainingGround/LD/datasets/17R_dataset'
-h            Display this help and exit

```

```
(myenv) (base) argus-vision@argus-vision:~/vision/VisionTrainingGround/DataPipeline$ ./run pipeline.sh
```

Generate download URLs:	100% ███		50/50 [00:08<00:00, 6.17it/s]
Download Images:	100% ███		50/50 [00:08<00:00, 6.07it/s]
Generate Saliency Maps:	100% ███		50/50 [00:00<00:00, 63.57it/s]

```
Merging saliency maps..  
Maps merged.  
Create Landmark Bounding Boxes::   0%|███████████████████████████████████████████████████████████████| 0/1 [00:00<?, ?it/s]  
Window Size: 100  
Number of Boxes: 1000  
Create Landmark Bounding Boxes:: 100%|███████████████████████████████████████████████████████████████| 1/1 [00:00<00:00, 1.85it/s]  
Processing Images: 100%|███████████████████████████████████████████████████████████████| 50/50 [00:03<00:00, 14.05it/s]  
Generating Label Files: 100%|███████████████████████████████████████████████████████████████| 50/50 [00:00<00:00, 1178.00it/s]
```

Dataset configuration saved to /home/argus-vision/vision/VisionTrainingGround/LD/datasets/17R dataset/dataset.yaml

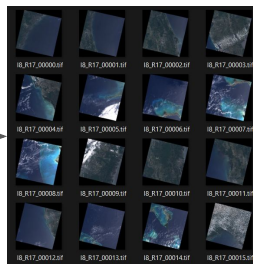
# Data Pipeline Workflow

## Google Earth Engine API

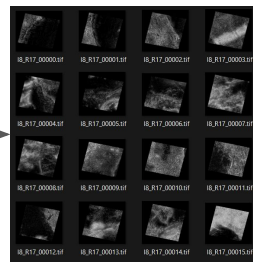
with control of

- Region
- Image count
- GSD / size
- Cloudcover etc..

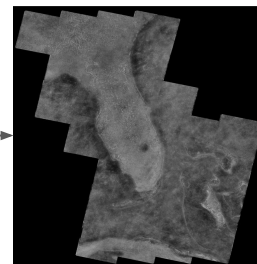
*Download  
images for  
specific region*



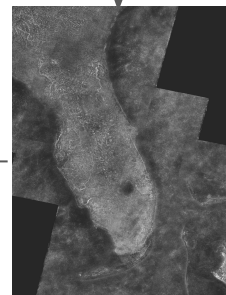
*Create maps of  
local saliency for  
each image in  
region*



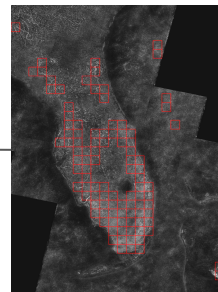
*Merge and  
reproject maps of  
local saliency to  
get global  
saliency (of the  
region)*



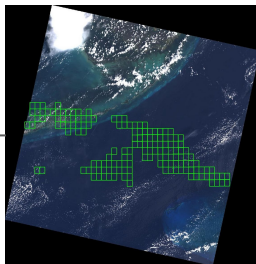
*Crop to remove out  
of bounds area*



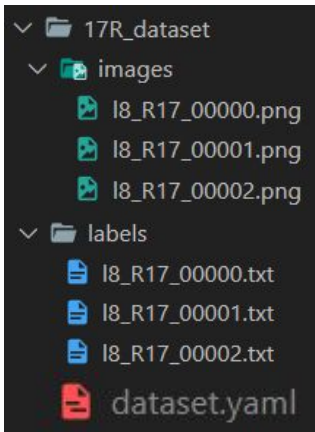
*Scan through  
saliency map to  
find most  
salient boxes  
(landmarks)*



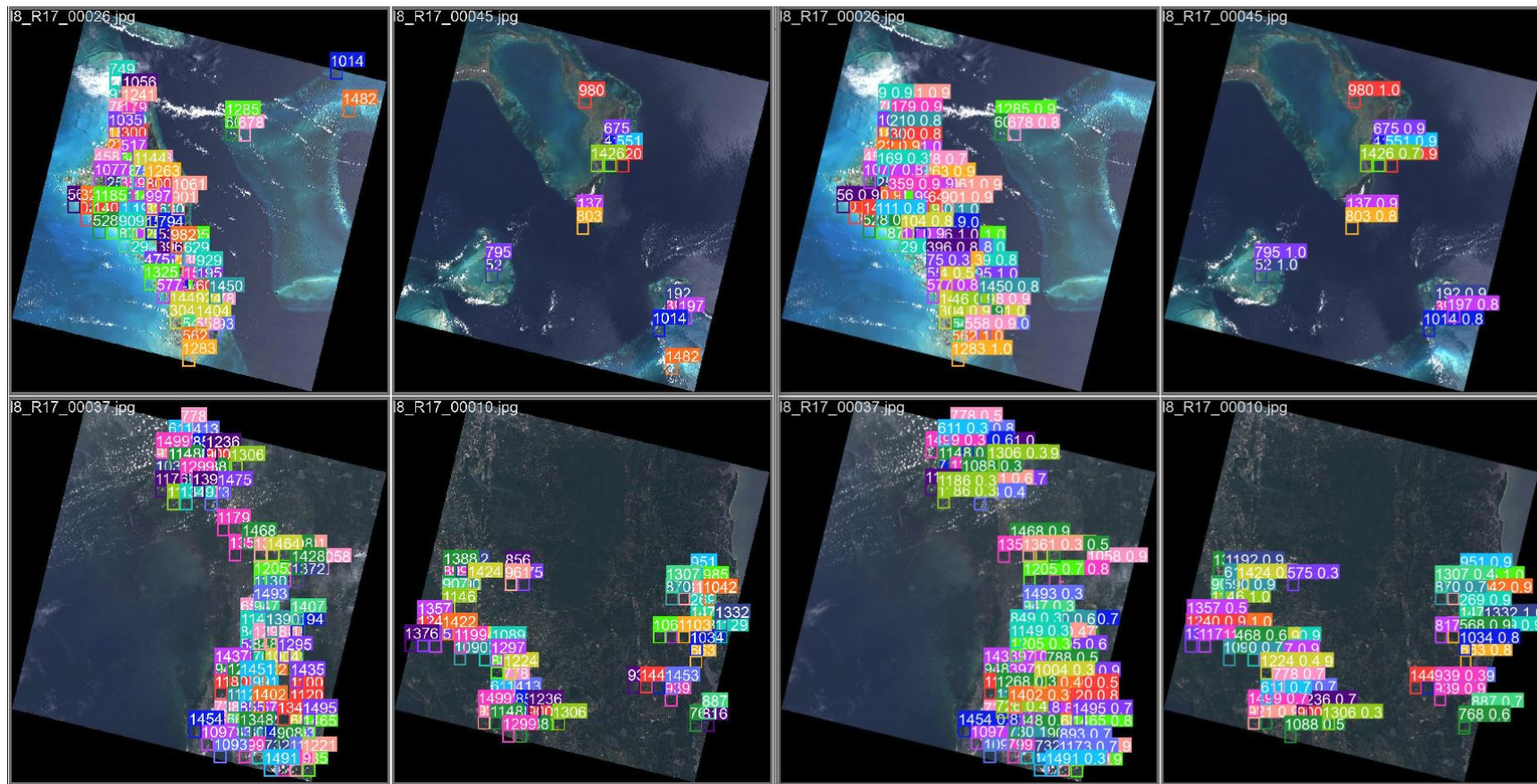
*Assign  
landmarks to  
individual  
image*



- Convert tif to png
- Assign class label for landmark
- Generate label file for each image
- Generate dataset config file



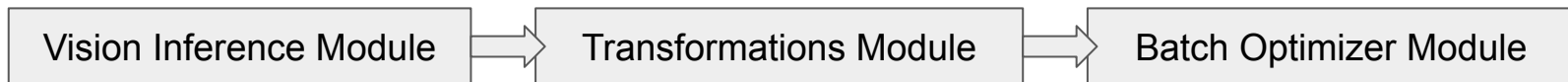
# LD Training Result





# Integrated testing - V1

- Integrated vision/ML, coordinate transformations and batch optimization modules to execute estimation pipeline for V1 prototype
- Inference for landmark detection is done on existing images and result is converted to camera vector and ECI coordinates used for estimation.



## Next steps

- Validate pixel coordinates to camera vector transformation ~ calibration
- Conduct entire pipeline test with images captured by camera
- Implement feature tracking between consecutive images, Batch LS

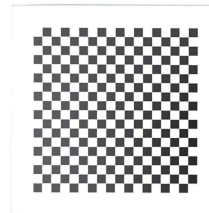
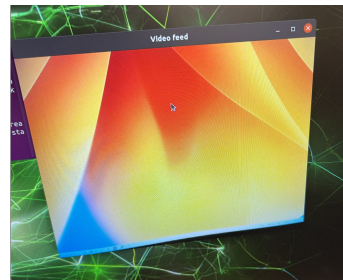


# Dark room testbed

- All components are now relative to the ground truth (left edge of the table)
- Minimized error inside the setup by using spirit level, tape measure and caliper

## Next steps

- Collaborate with vision testings and see if more modifications need to be made
- Merge the current setup as the starting point for V2 testing bed
- Look into how to match the projected image w/ the actual image

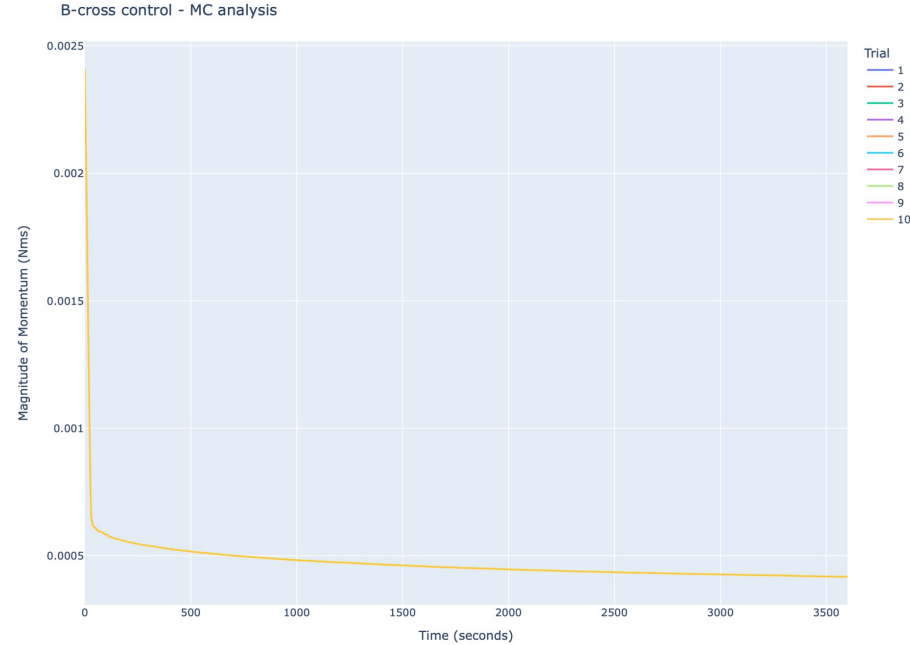


# Simulator

- Features added:
  - Attitude and Orbit Sampling
  - Monte Carlo Analysis Setup
  - Logging and visualization support for MC analysis
- Testing:
  - Preliminary MC analysis of B-cross control
    - 10 runs, 60 minutes duration, 1 Hz control cycle, 10 Hz integrator cycle, Altitude [500 km - 600 km] SSO

Next steps:

- More testing with increased number of runs, different ranges for sun-synchronous orbits, different control and integrator sampling times
- Updates:
  - Sensor transformations based on location
  - Sun sensor model



[B-cross 10runs MC Plot](#)

# Team schedule status → V2

V2	Vision Pipeline Validation and FSW kick-off	Vision dev	Data Engine Pipeline automation (--> Kyle)	Eddie Li																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																	
----	---	------------	--	----------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--