AERO-DETR

Coarse-to-Fine Runway Extraction via Orientation-Normalized Marking Detection

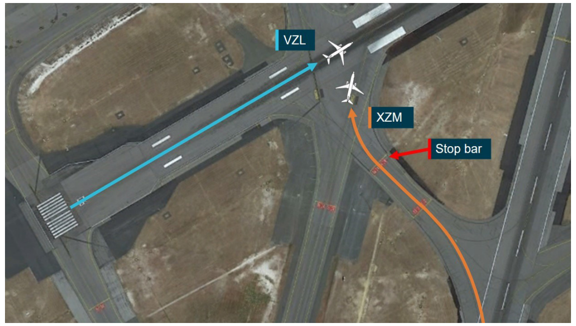
Durga Prasad Dhulipudi[1,2] **Krishnan Sundara Rajan**[1] **Sachin Raja**[1]

[1] Lab for Spatial Informatics, IIIT Hyderabad, India
[2] Honeywell Aerospace

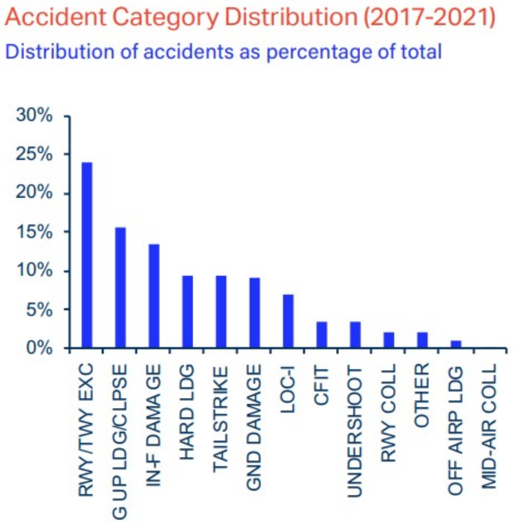
4th Intl. Conference on Machine Intelligence for GeoAnalytics and Remote Sensing (MIGARS 2026)

Rising Complexity and Surface Safety

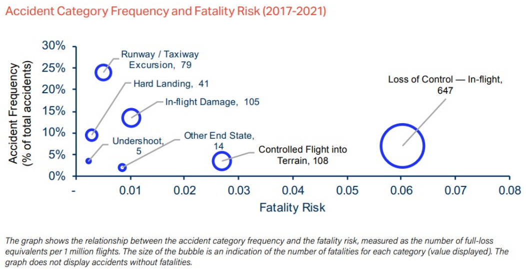
Growing traffic and airport complexity raise **runway-incursion** risk during taxi; Airport Moving Maps improve situational awareness.



Runway incursion



Surface incidents



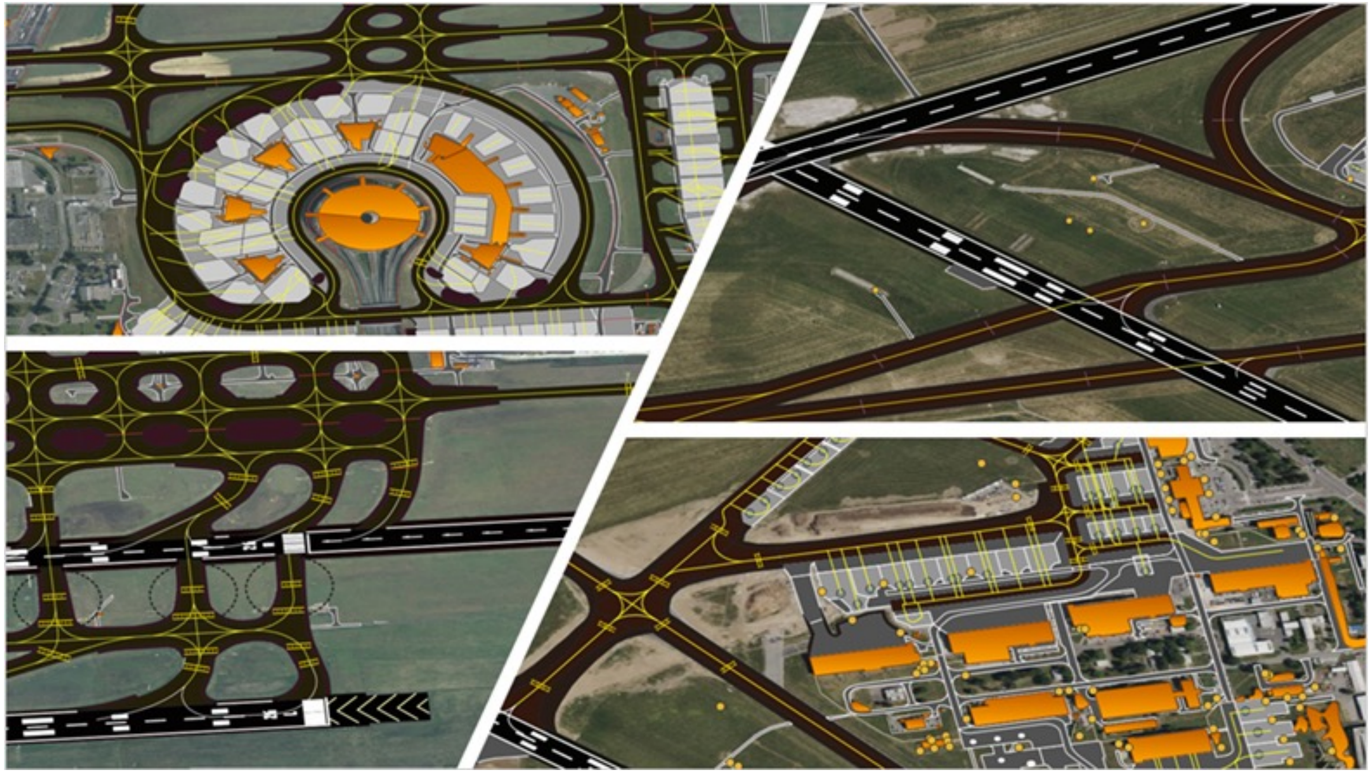
Fatality trend

AMM cockpit view

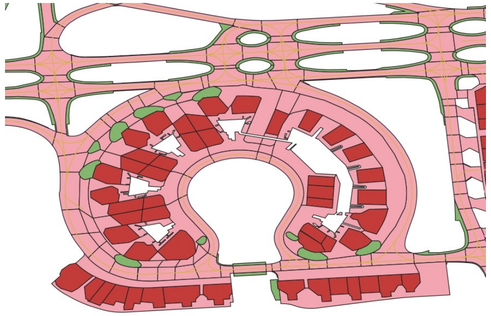
Source: IATA safety reports and aviation AMM practice.

Data Foundation: Aerodrome Mapping Database

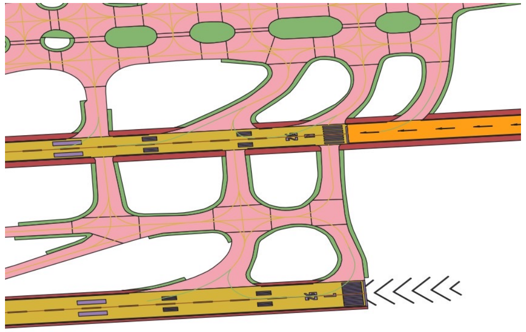
Safety-critical mapping depends on the **Aerodrome Mapping Database (AMDB)** with accurate runway geometry and markings.



Airport scene imagery



Vector features



Overlay and validation

Sliding Window Approach

- **Sliced inference is the standard workflow:**
  - Partition image into overlapping tiles
  - Run the detector per tile
  - Stitch and merge detections back together

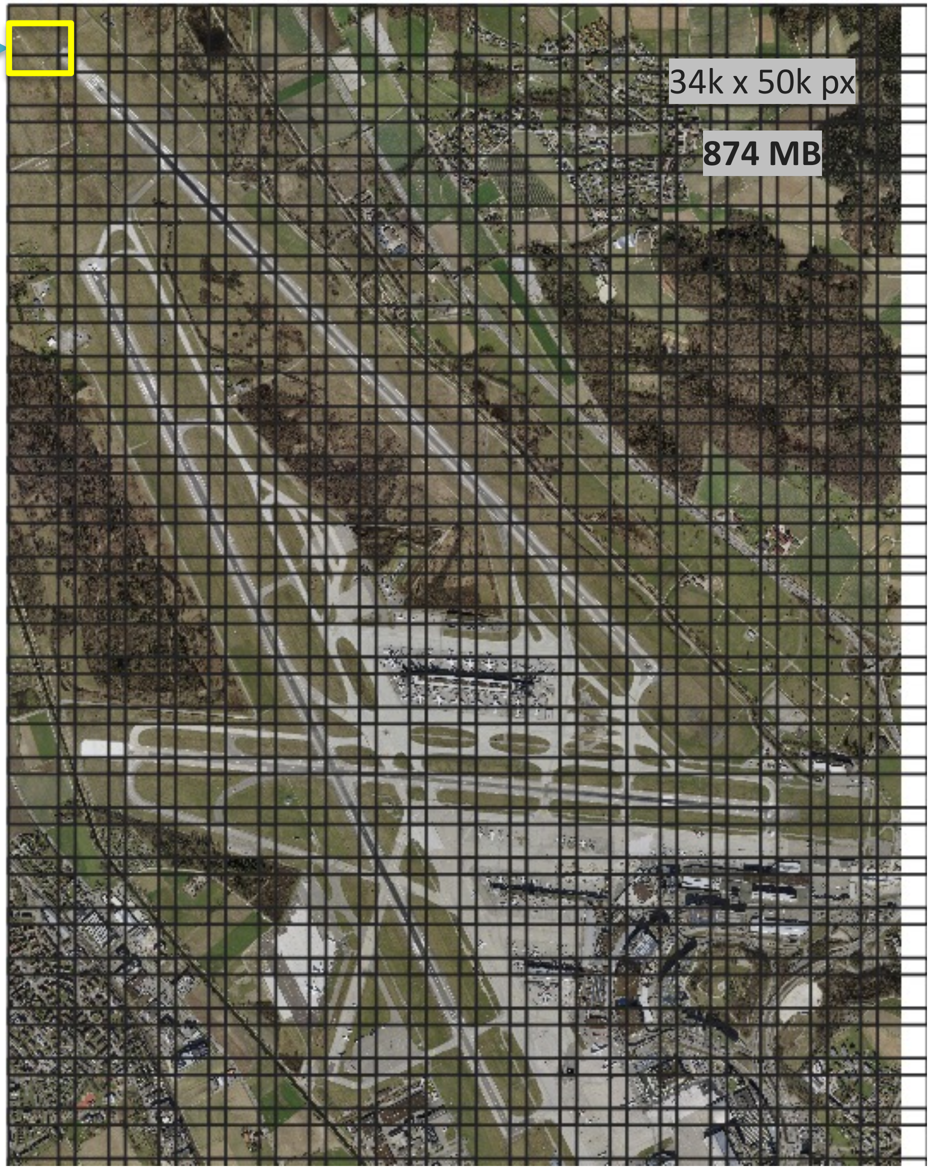
Image Size : 34,000 × 50,000
Tile Size : 512 × 512
Overlap : 128 px
Stride : 512 - 128 = 384 px

Tiles (Width) : [(34,000 - 512) / 384] + 1 = 89
Tiles (Height) : [(50,000 - 512) / 384] + 1 = 130

Total Tiles : 89 × 130 = **11,570**

The Core Challenge

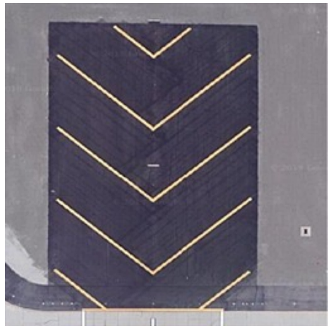
- Detect small runway portions
- Need a dataset + model for these small portions
- Assemble them together
- Form a seamless Runway
- Involves lot of pos-processing

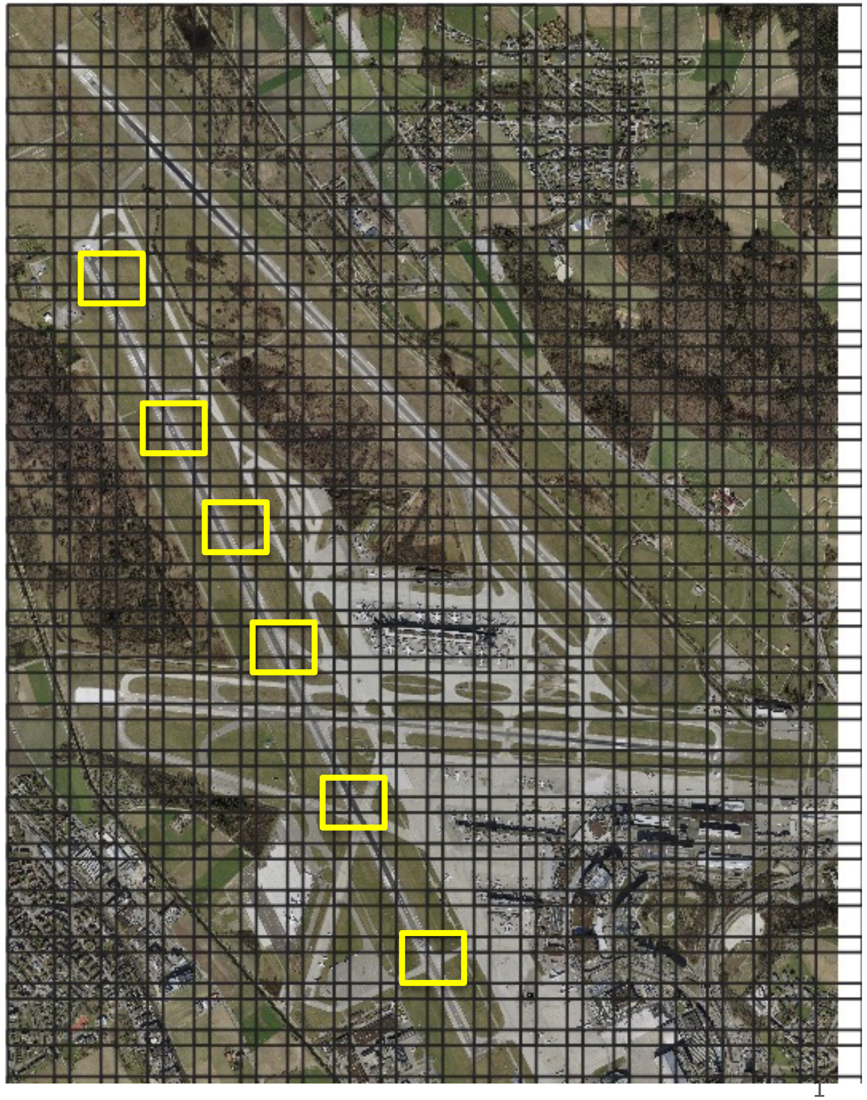


Sliding Window Approach

The Core Challenge

- Detect small runway portions
- ***Need a dataset + model for these small portions***
- Assemble them together
- Form a seamless Runway
- Involves lot of **post-processing**

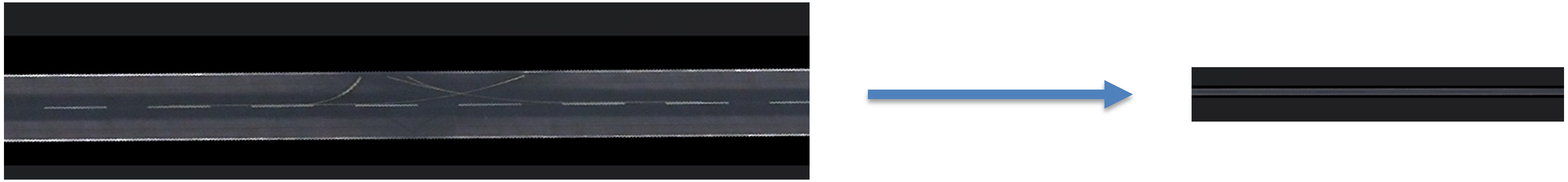




Resizing for Inference – Loss of pattern

The Core Challenge

If we resize the entire 34k × 50k satellite image to a detector input size, the runway-pattern is destroyed before the model even sees it.



Identified Limitations - Coarse & Noisy Detections

Prior Work & Identified Gaps

The following limitations are identified in current methodologies:

Coarse Satellite Images

Features from Sentinel-2 (10 m), Landsat-8 (30 m), and Gaofen-1 (8 m) are too coarse for surface navigation.

Coarse-image detection [Chen et al., 2018]

SAR Images

SAR suits military and disaster use; inherent noise obscures crucial runway features.

SAR runway detection [Lv et al., 2018]

Coarse Runway Boundaries

Only coarse boundaries are recovered: suitable for the landing phase, but not for taxiing.

Coarse boundaries [Men et al., 2020]

Gap: surface navigation requires fine-grained runway geometry and markings, not only coarse runway presence.

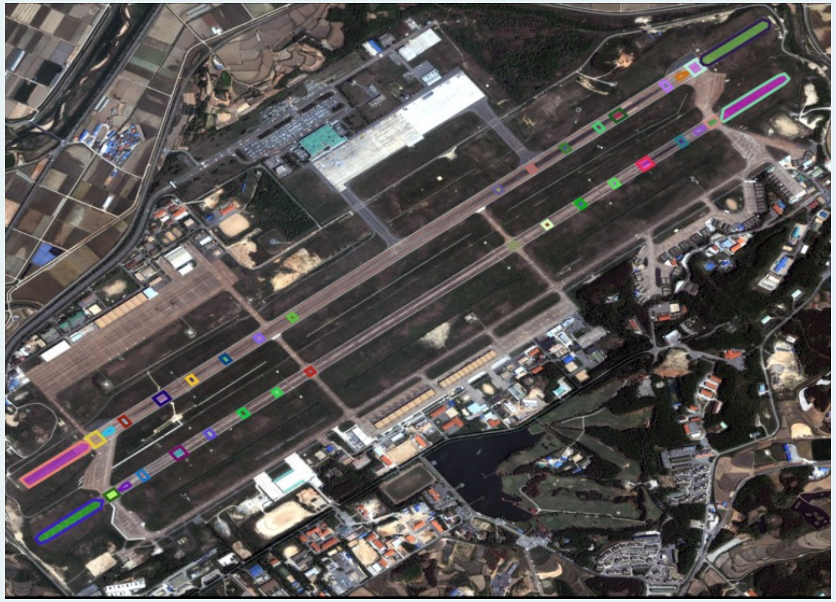
Problem Gap and Opportunity

- Runways are elongated, arbitrarily oriented, and often span very large pixel extents.
- Fine-grained markings are small and sparse relative to full-scene imagery.
- Prior work usually solves only one side: either coarse runway localization or marking details.
- AERO-DETR links both scales in one pipeline for geometry-faithful runway extraction.

Two Purpose-Built Datasets

433 NOAA VHR scenes (RGB+NIR; 1 m pre-2018, 0.6 m from 2018); digitized in GeoJSON/Shapefile -> YOLOv8/COCO.

Top-Down: runway OBBs (1,038 objects)



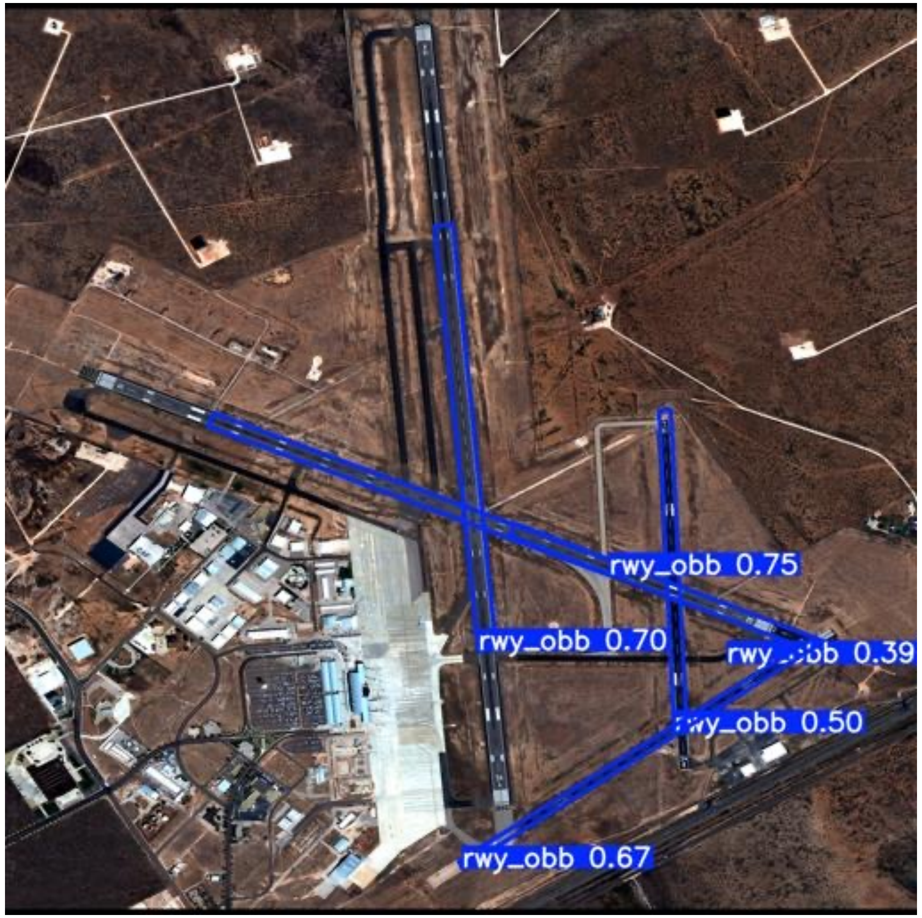
Bottom-Up: 8 marking classes - tdz, aim, desig, rwythr, thrbar, disp, chevron, arrow (12,815 objects)

Top-Down Model: Coarse Runway Corridors

Segmentation models gave patchy runways; YOLOv8-OBB produced clean, axis-aligned corridors.

Model	Metric
Mask-RCNN (Detectron2)	AP 10.6
SegFormer	IoU 0.448
Mask2Former	mIoU 0.383
YOLOv8-OBB	mAP50 0.945; P 0.943; R 0.883

Corridor = medial-axis centerline -> extend to image bounds -> buffer -> clip.

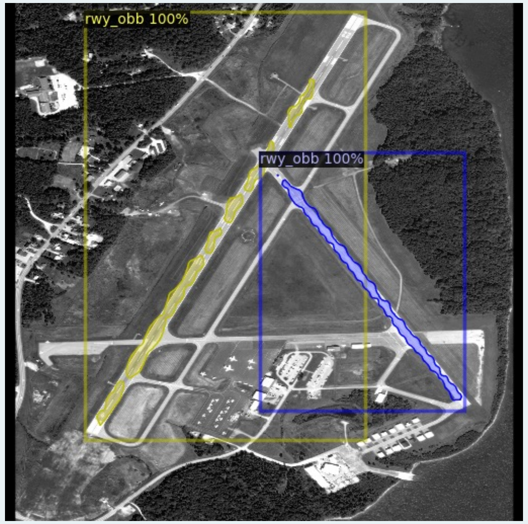


YOLOv8-OBB runway corridor

Baseline Context and Positioning

AERO-DETR combines top-down OBB localization with bottom-up marking geometry to recover precise runway boundaries.

Mask R-CNN



Strong instance segmentation baseline.

SegFormer

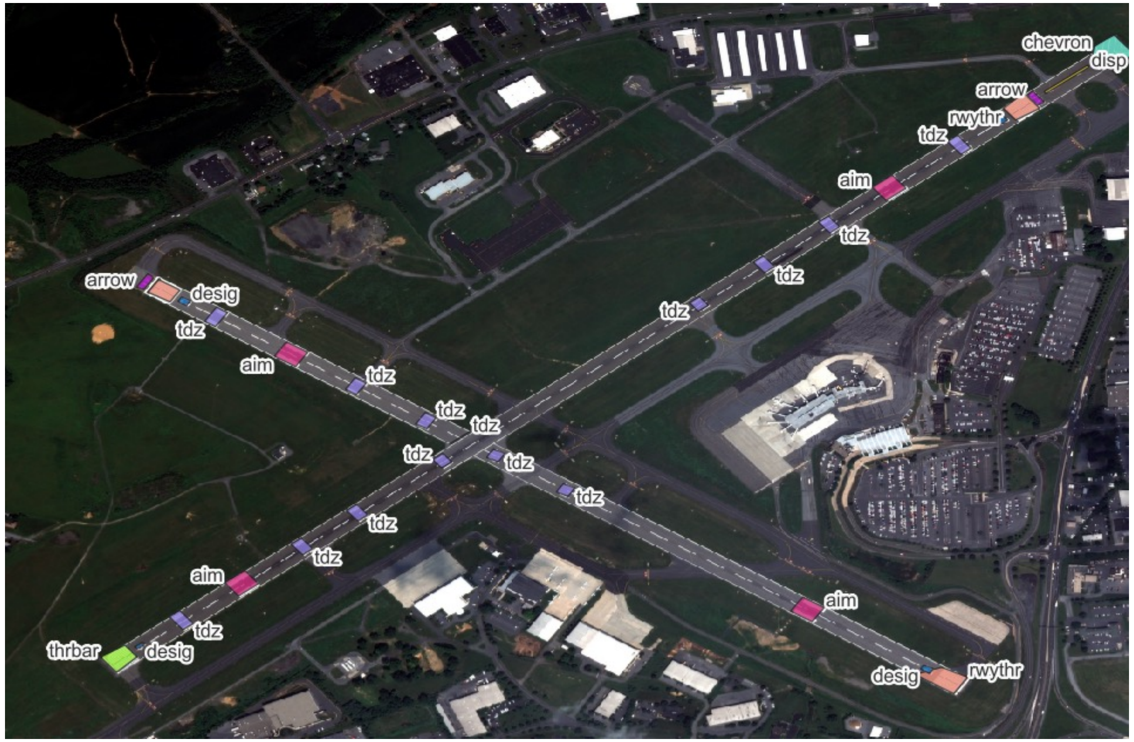
Transformer segmentation for runway extents.

YOLOv8-OBB

Top-down oriented localization baseline.

Bottom-Up Model: Orientation Normalization

Each corridor's markings are rotated to a horizontal canonical orientation, standardizing input for the marking detector.



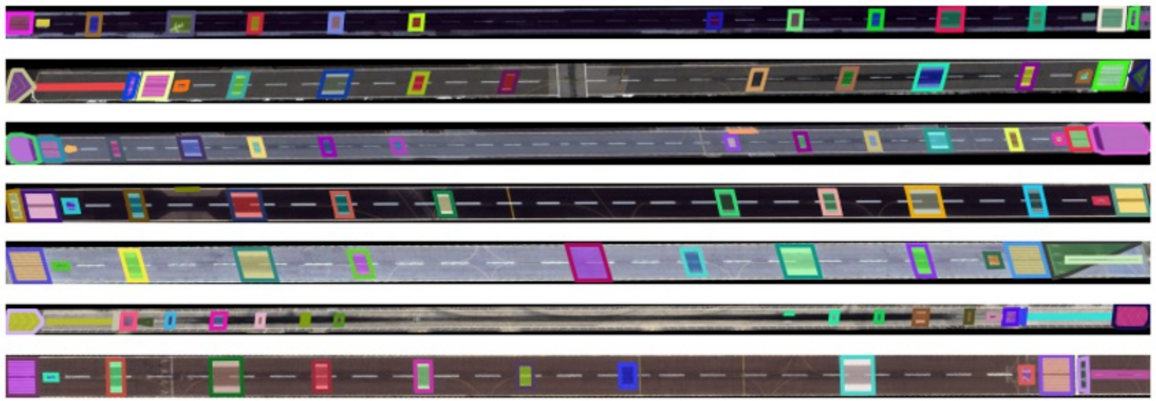
Original orientations

Rotated to horizontal

Marking Taxonomy (DO-272) & Samples

Class	DO-272 Element	Count
tdz	Touchdown Zone	5,680
aim	Aiming Point	1,764
desig	Designation Number	1,931
rwythr	Runway Threshold	1,484
thrbar	Threshold Bar	856
disp	Displaced Threshold	320
chevron	Chevron / Overrun	538
arrow	Arrow	243
Total	**8 classes**	**12,816**

Natural long-tail: touchdown-zone bars dominate; chevrons & displaced thresholds are rare.



Orientation-normalized runway crops with fine-grained marking polygons.

System Architecture in Five Modules

1 Top-down OBB

Predict coarse runway hypotheses (center, size, angle, confidence).

2 Normalize

Rotate by -theta and crop a canonical horizontal runway chip.

3 Marking detector

Detect 8 runway marking classes with geometry and confidence.

4 Geometry-aware pairing

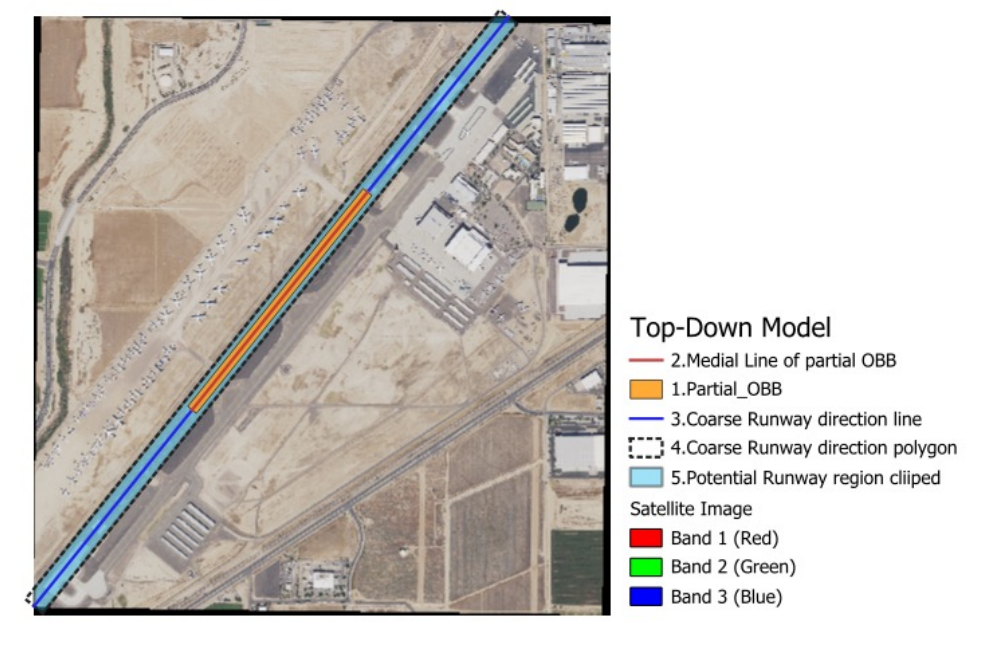
Pair runway-end markings and aggregate supporting marks.

5 Inverse transform

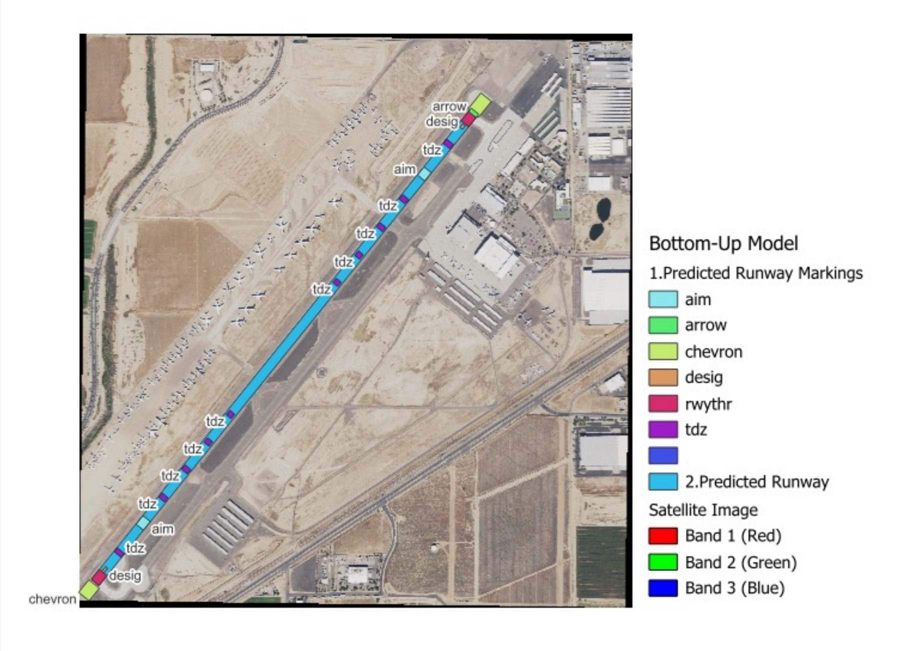
Map reconstructed polygon back to original image coordinates.

AERO-DETR: Dual-Perspective Coarse-to-Fine Pipeline

Top-down runway localization



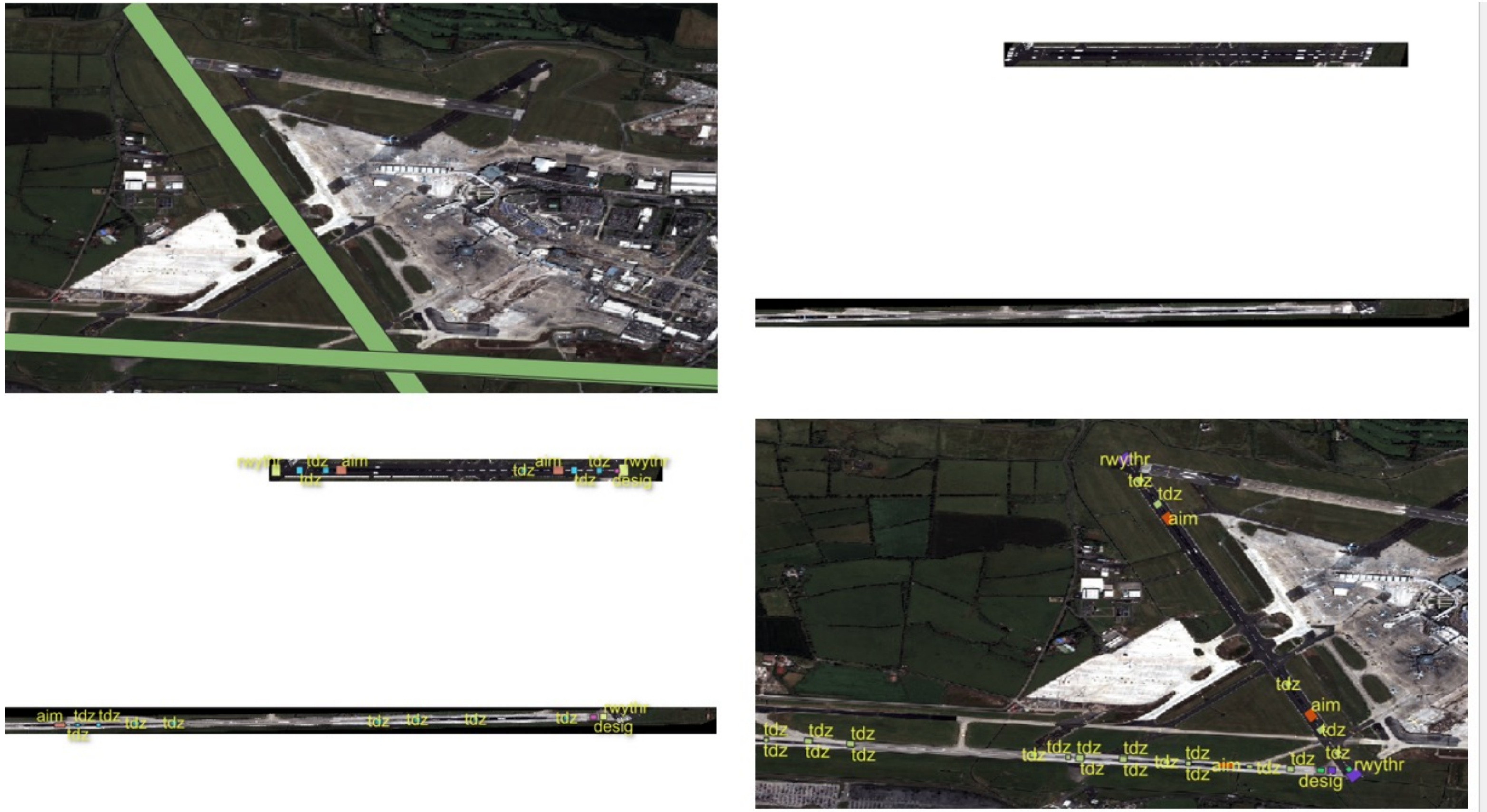
Bottom-up marking detection



Top-down OBB gives location and heading; bottom-up classes and geometry drive final runway polygon.

Precise Runway Extraction - Result

Assembling rwythr + desig yields tight, georeferenced runway polygons, validated by IoU against manually digitized ground truth (airport-wise mean IoU ~= 0.89).



Accurate runway detection from the AERO-DETR pipeline.

Full-Image AERO-DETR Pipeline: Inference Speed

End-to-end runtime (top-down OBB -> horizontalize -> DETR markings -> geoprocessing) measured on 126 full-airport rasters.

Category	Airports	Min (s)	Max (s)	Avg (s)
Single	25	1.10	9.18	3.96
Parallel	25	0.65	99.27	15.11
Intersection	25	3.24	64.07	13.96
Mixed	26	0.90	85.93	10.48
Complex	25	2.71	260.13	39.17
Overall	126	0.65	260.13	16.49

Runtime scales with airport complexity and raster size: most configurations finish within a few seconds; large complex scenes take longest.

AERO-DETR Marking-Detection Metrics

+17 mAP points over DETR while remaining the fastest; sets a practical benchmark for satellite runway analysis.

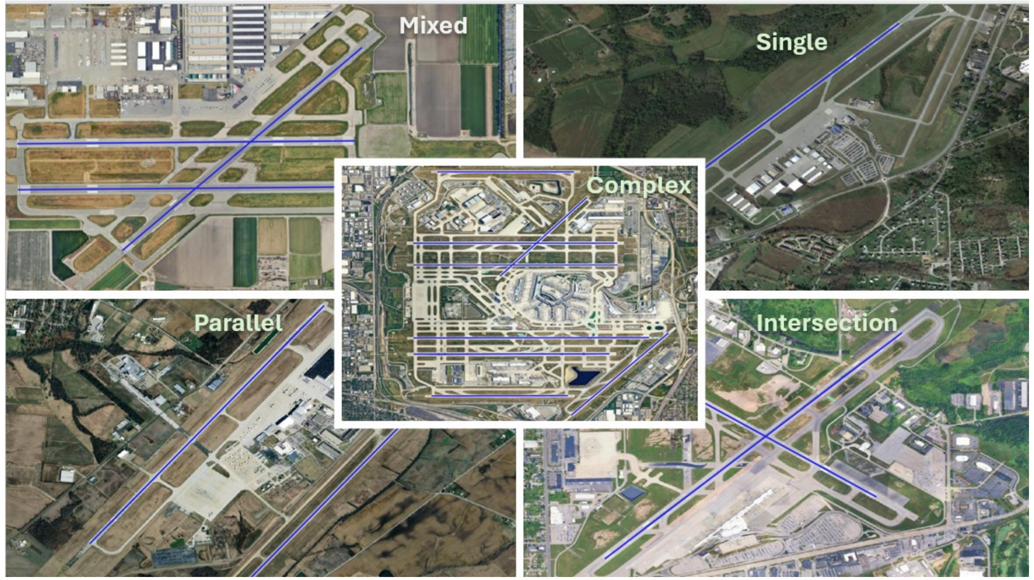
Algorithm	Precision	Recall	mAP50	mAP50-95	Speed (ms)
Mask-RCNN	-	-	23.15	11.05	836.3
YOLOv8	0.568	0.554	0.477	0.22	130
Segformer	-	-	-	0.0441	769.4
Mask2former	-	-	-	0.14488	1535
DETR	0.694	0.586	0.565	0.369	84.3
AERO-DETR	0.771*	0.768*	0.741*	0.572*	70.6*

Runway surface markings detection (orientation-normalized).

AERO-DETR Runway Benchmark (MIGARS'26 dataset)

A public, georeferenced benchmark (Zenodo), validated by spatial IoU per runway configuration.

Category	Airports	IoU mean	# IoU > 0.8
Single	25	0.9064	23
Parallel	25	0.9066	22
Intersection	24	0.8614	5
Mixed	26	0.8874	14
Complex	25	0.9018	17
Average	125	0.8927	81



Five runway configurations evaluated

NAIP + NOAA public domain (GSD 0.42-0.72 m); dual OBB (1,038) + 8-class markings (12,816); 995 orientation-normalized crops; airport-disjoint splits; DO-272 / FAA AC 150/5340-1M.

Key Technical Differentiators

- Dual-perspective architecture: top-down runway localization + bottom-up marking detail.
- Orientation normalization applied during both training-data creation and inference.
- Marking-driven deterministic reconstruction using runway-end pairing constraints.
- Hybrid learned + geometric reasoning to improve robustness on elongated structures.
- Inverse mapping restores geospatially consistent runway polygons in original frame.

Dataset Link

https://doi.org/10.5281/zenodo.21094439

Thank You

Questions and Discussion

Durga Prasad Dhulipudi[1,2] **Krishnan Sundara Rajan**[1] **Sachin Raja**[1]

[1] Lab for Spatial Informatics, IIIT Hyderabad, India
[2] Honeywell Aerospace