



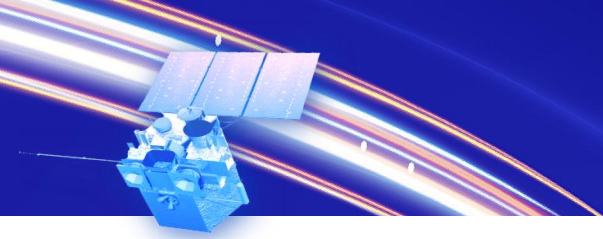
AOMSUC-15 2025 FYSUC

THE 15TH ASIA-OCEANIA METEOROLOGICAL SATELLITE USERS' CONFERENCE (AOMSUC-15)
2025 FENGYUN SATELLITE USER CONFERENCE (2025 FYSUC)

A Hybrid pySTEPS and TrajGRU Model for Precipitation Nowcasting Using Samut Songkhram (SKM) Weather Radar Data

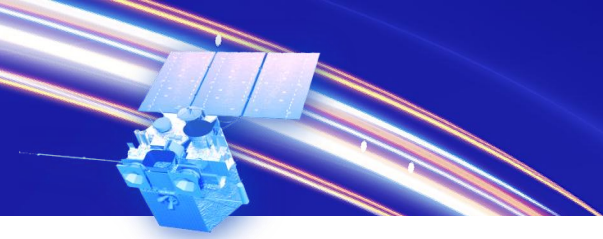
Saowapak Buphu
Meteorologist, Thai Meteorological Department (TMD)



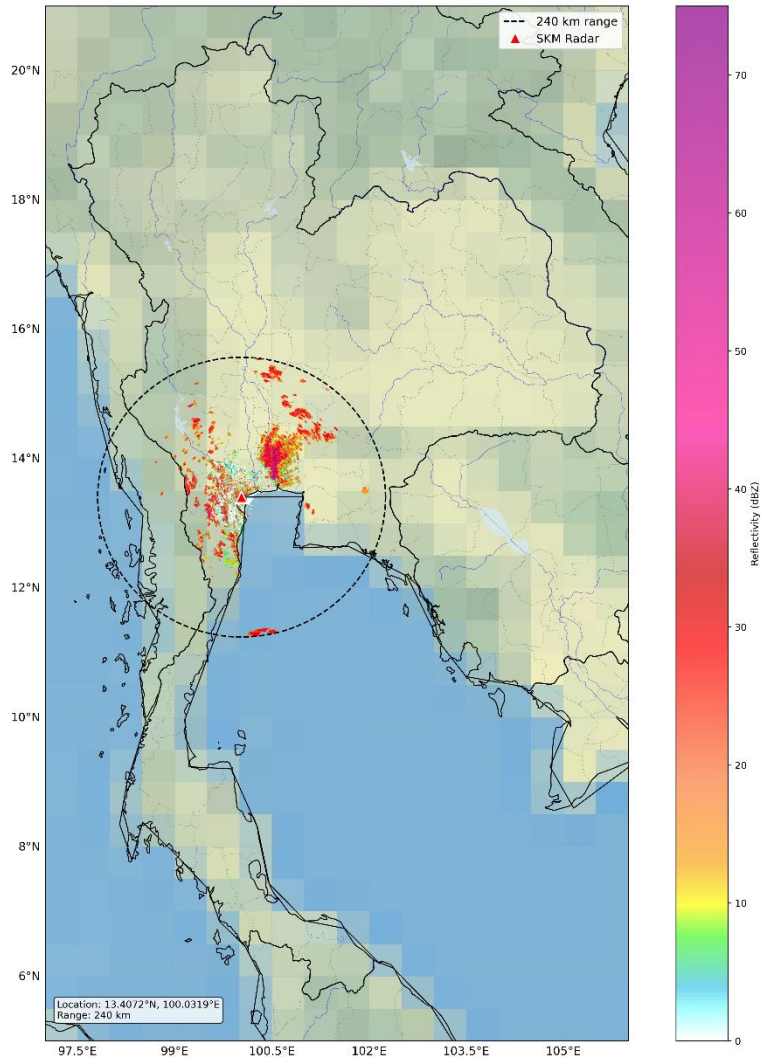


Outline

- 1. Radar Data Overview for Site SKM under TMD**
- 2. Methodology**
- 3. Evaluation Metrics**
- 4. Results**



SKM Radar - Samut Songkhram, Thailand
2025-09-13 00:00:08 UTC



Radar Data Overview for Site SKM under TMD

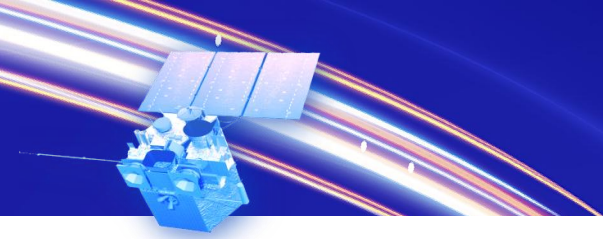
Format: UF file, updated every 15 minutes

Period: 2022-01-01 to 2024-12-31 (3 years)

Nsweeps: 4 Sweeps (**Degree:** 0.8 | 1.48 | 2.39 | 3.48)

Nrays: 240 Gates

Fields: reflectivity, velocity, spectrum_width, corrected_reflectivity, corrected_differential_reflectivity, differential_phase, cross_correlation_ratio



Applies Quality Control (QC) on Radar Reflectivity Data using Gate Filtering and Median Filtering.

Step A (Conditional Azimuthal filter, if no `cross_correlation_ratio` data)

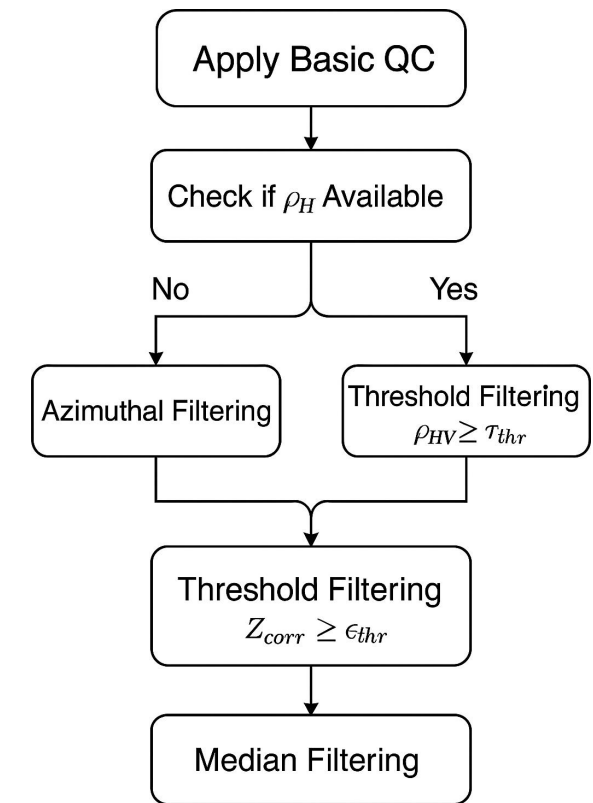
- If `cross_correlation_ratio` exists in the radar data, apply clutter filtering (**Cross-correlation threshold = 0.8**).
- If **not**, apply 1D median filtering in the **azimuthal direction only** to reduce radial streaks. (**Azimuthal window = 20 azimuths**)

Step B (Threshold filtering)

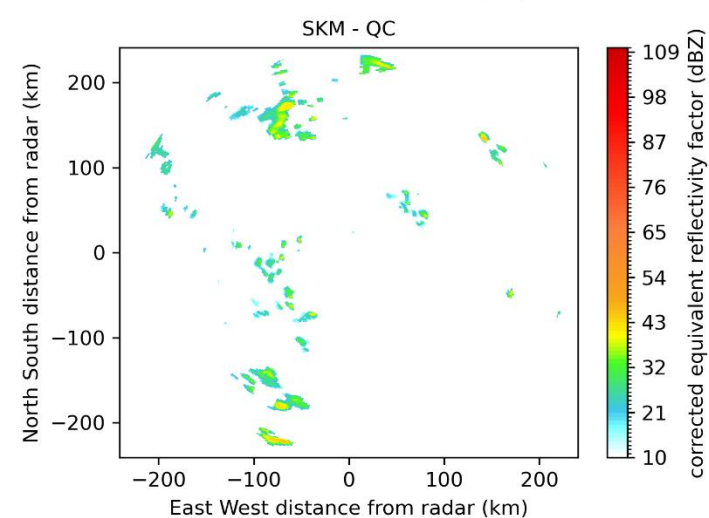
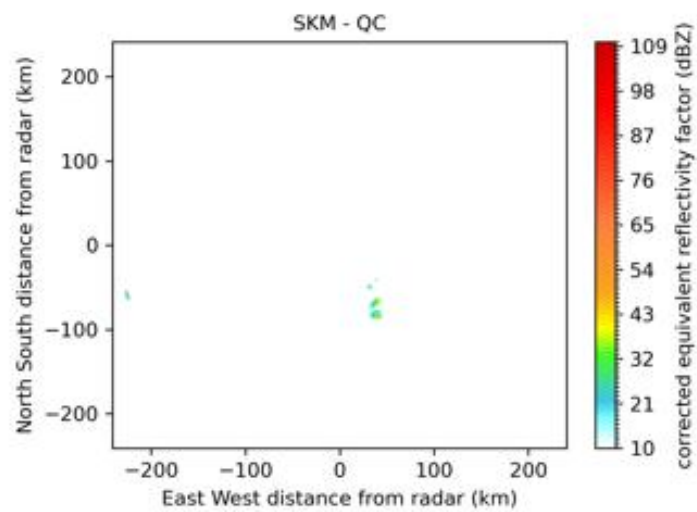
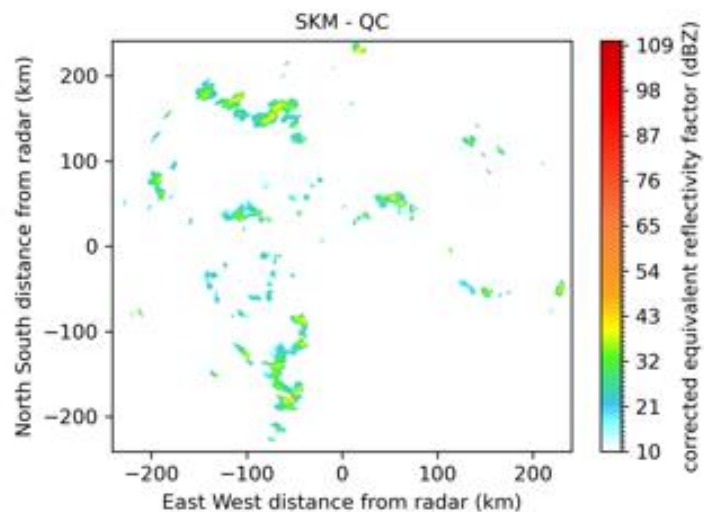
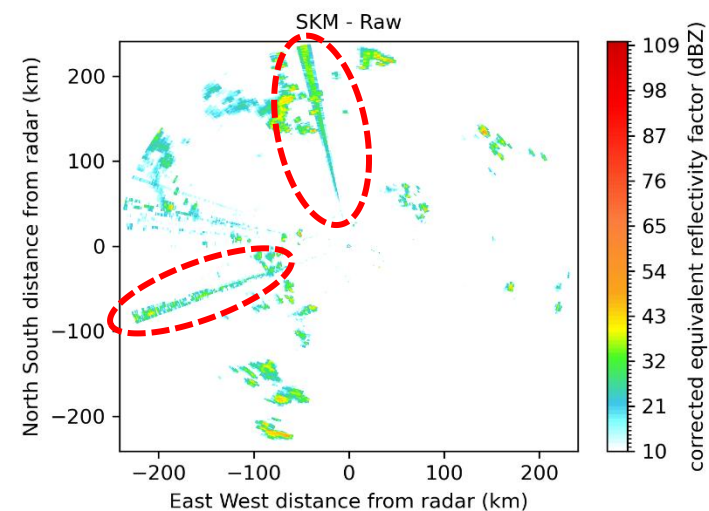
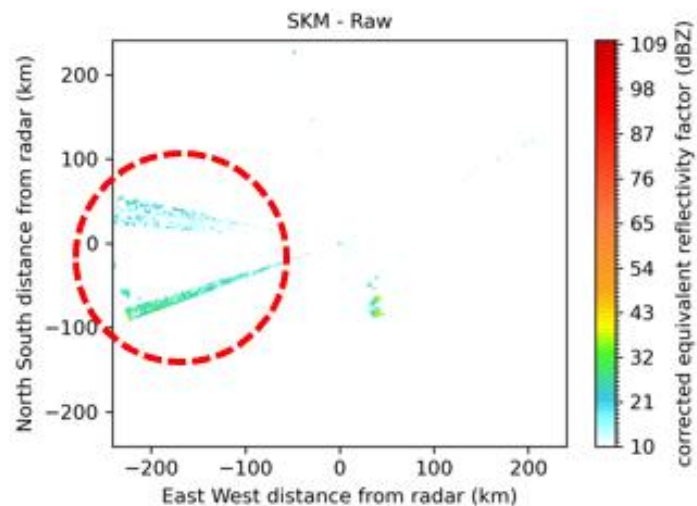
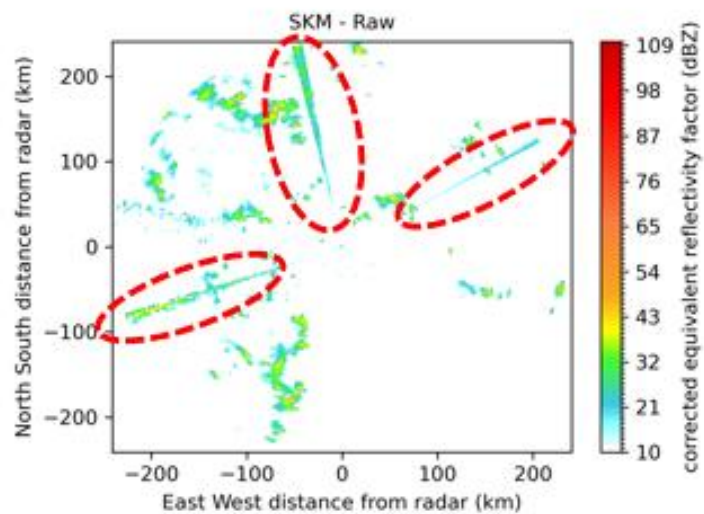
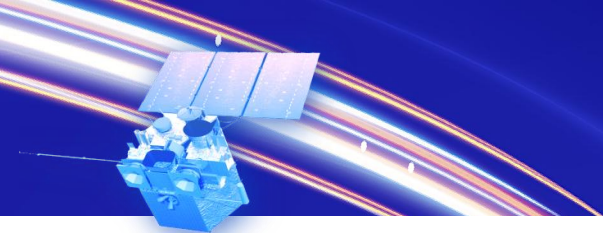
- Apply basic reflectivity gate filtering. Remove gates where **Z < 10 dBZ**.

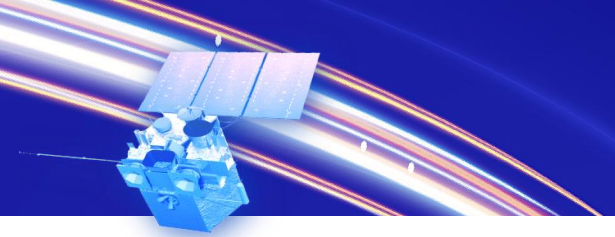
Step C (Median filter)

- Apply 2D median filtering over neighboring gates to reduce speckle **noise** (**The window is a square of size (3 × 3)**)



AOMSUC-15 2025 FYSUC

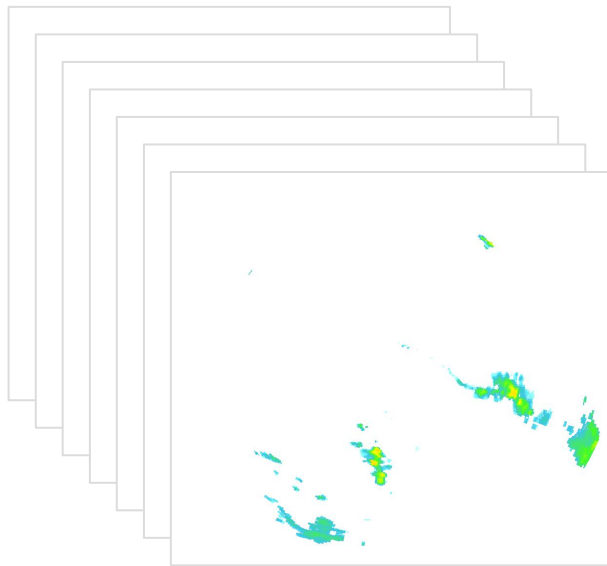




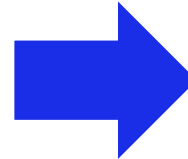
Methodology

Method 1: Pysteps (Physics-Based)

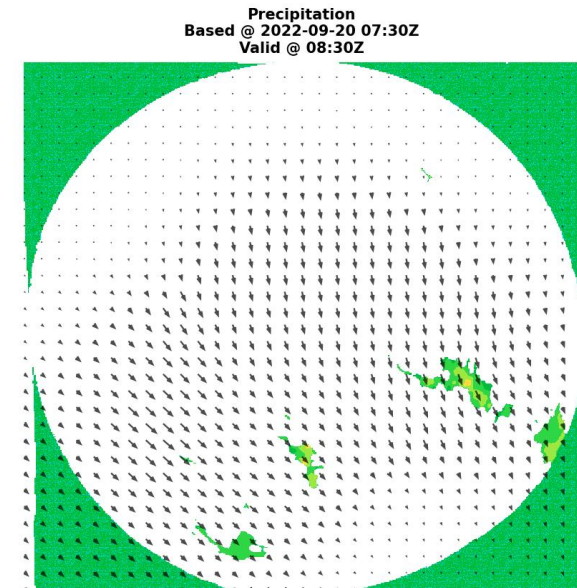
This method forecasts the *motion* of the radar echo.



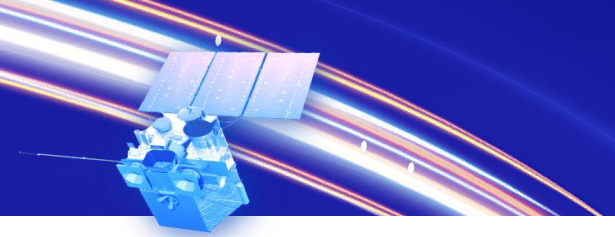
Input: Load 7 Frames (**Z**) (Each frame is 15m)



Motion: Calculate Motion
Optical Flow (Lucas-Kanade)



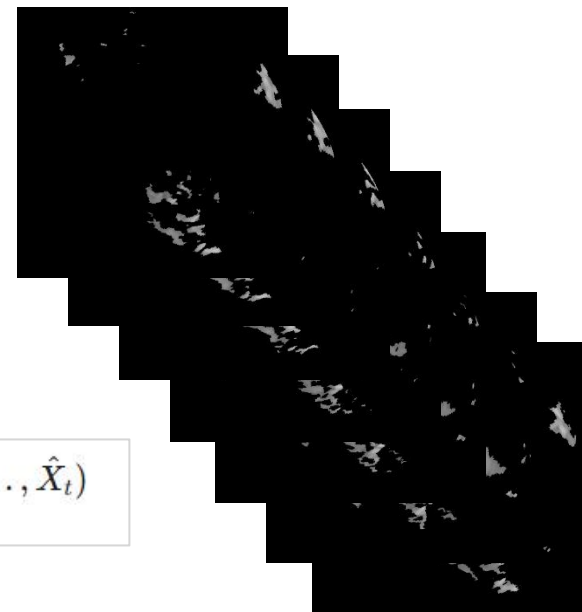
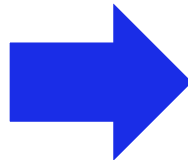
Forecast: Extrapolate Advection (Semi-Lagrangian) ->
Generate Forecast (**Z**) (20 frames) ->
Apply **Z-R Relationship** ($a=300$, $b=1.4$)



Methodology

Method 2: TrajGRU (Deep Learning)

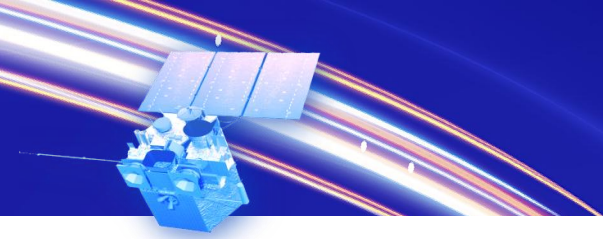
TrajGRU dynamically learns the motion of the weather patterns. It predicts a storm's future by looking at its "trajectory" where it *came from* rather than just its previous fixed-grid position, and is better at handling **moving, rotating, and deforming** (non-stationary) weather systems.



$$\tilde{X}_{t+1}, \dots, \tilde{X}_{t+K} = \arg \max_{X_{t+1}, \dots, X_{t+K}} p(X_{t+1}, \dots, X_{t+K} \mid \hat{X}_{t-J+1}, \dots, \hat{X}_t)$$

Input: Load 7 Frames (**Z**) (Each frame is 15m)

Output : 20 Frames (**Z**) ->
Apply **Z-R Relationship (a=300, b=1.4)**



Evaluation Metrics

Both methods produce a **Forecast (R)**, which is then compared to the **Observation (R)**.

1. Ground Truth:

1. Observation (**Z**) → Apply **Z-R Relationship (a=300, b=1.4)** → Observation (**R**)

2. Compare:

1. Forecast (**R**) [from pysteps] **vs.** Observation (**R**)
2. Forecast (**R**) [from TrajGRU] **vs.** Observation (**R**)

3. Calculate Metrics:

1. Calculate CSI, HSS, for both methods using the mm/h thresholds (# Thresholds: 0.5, 2.0, 5.0, 10.0, 30.0 mm/h).

CSI (Critical Success Index)

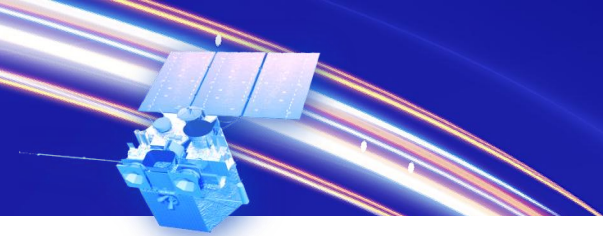
$$CSI = hits / (hits + misses + false_alarms)$$

HSS (Heidke Skill Score)

$$HSS = (po - pc) / (1 - pc)$$

po = proportion correct

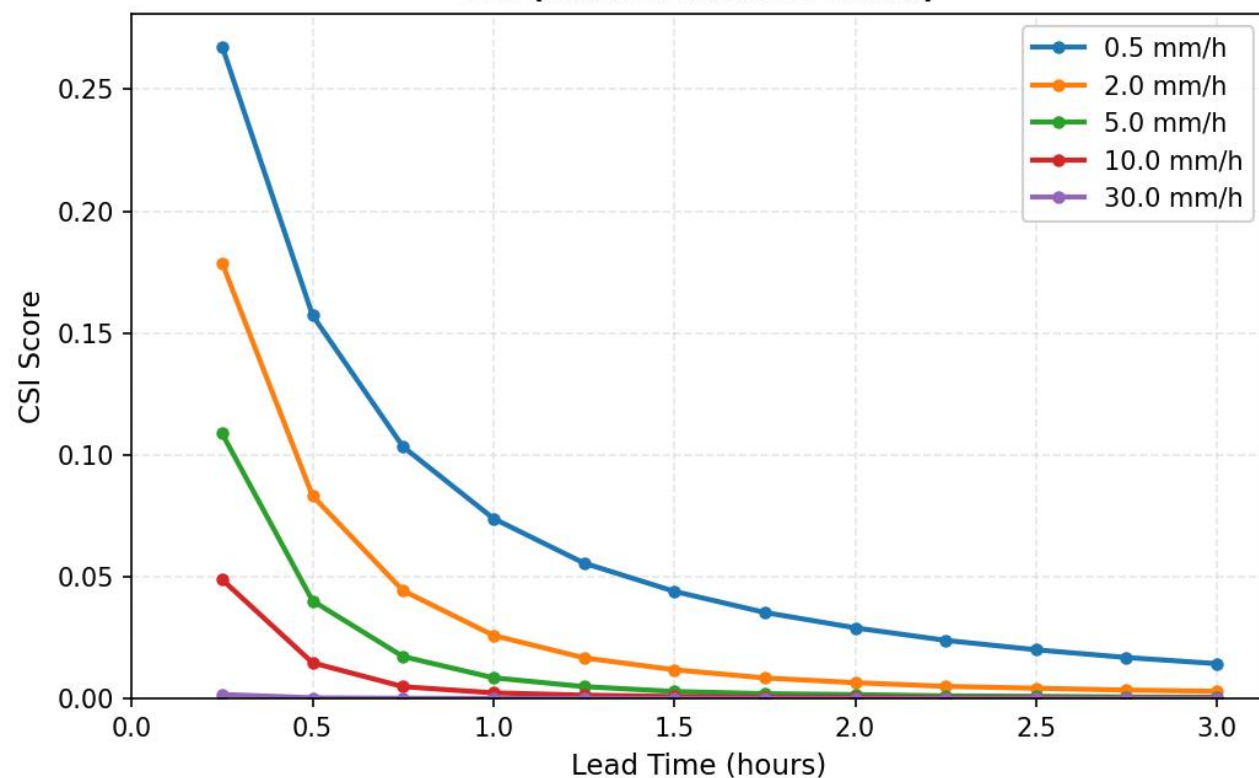
pc = expected correct by chance



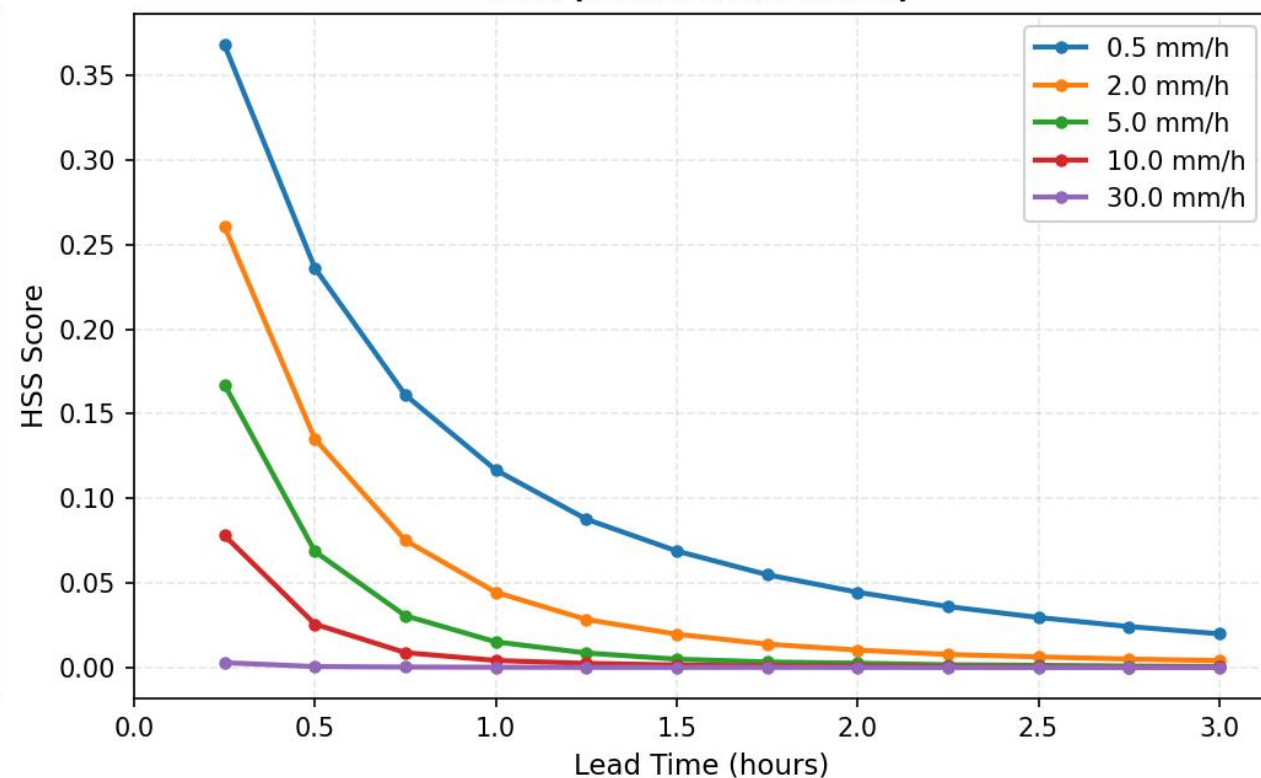
Result

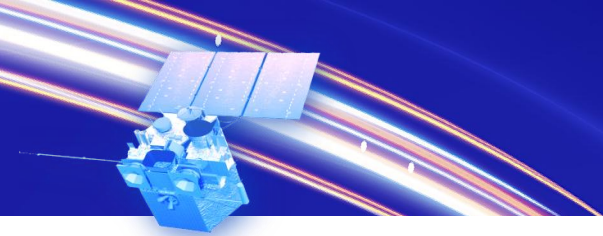
PySteps: Optical Flow (Lucas-Kanade) + Extrapolation (Semi-Lagrangian)

CSI (Critical Success Index)



HSS (Heidke Skill Score)

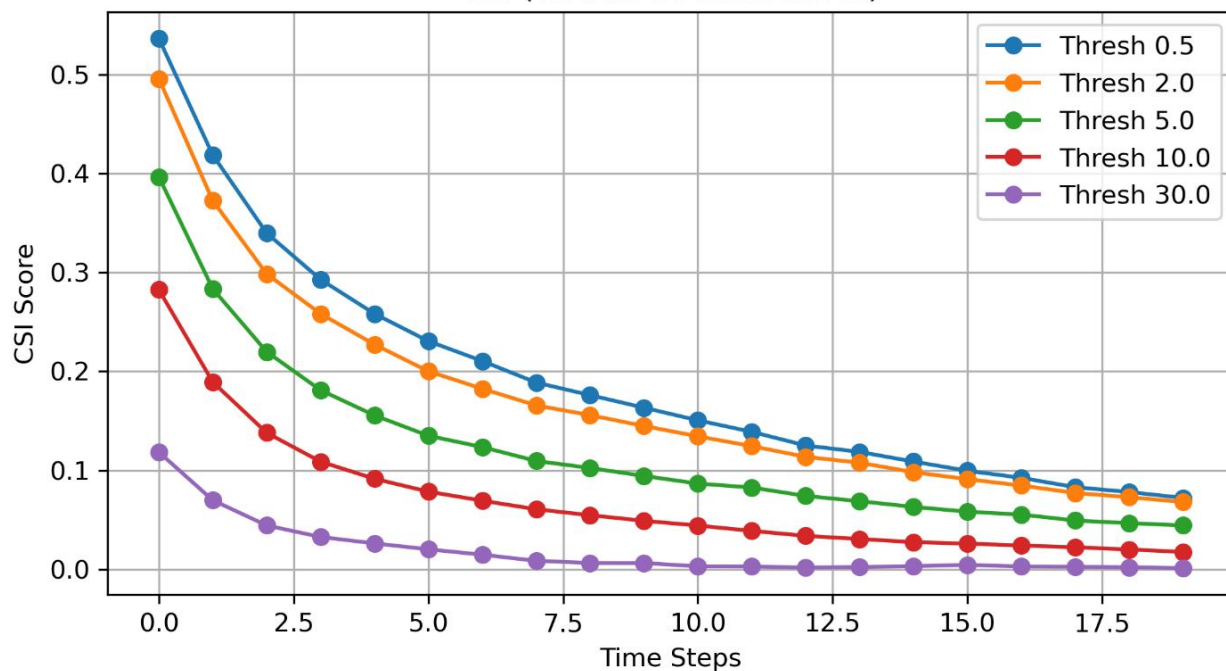




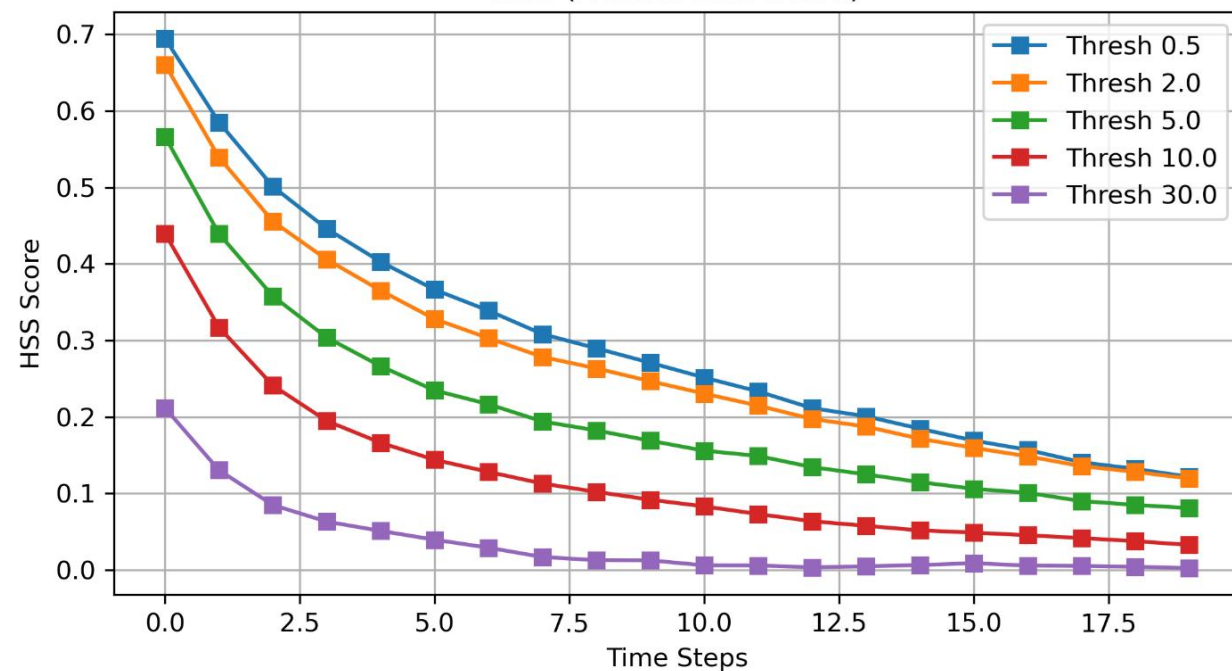
Result

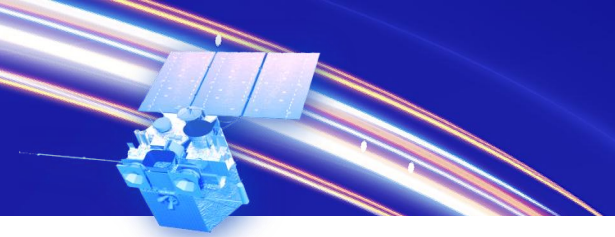
TrajGRU Model: Load Weight from HKO7 (encoder_forecaster_45000.pth)
and Inference with SKM dataset

CSI (Critical Success Index)



HSS (Heidke Skill Score)





Thank you for your attention