

Learning to Generate 3D Atmospheric Winds from Geostationary Imagery with Deep Optical Flow

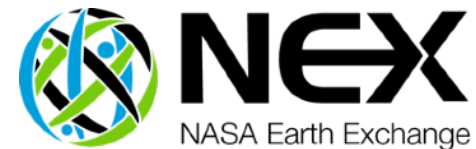
16th International Winds Workshop

May 11, 2023

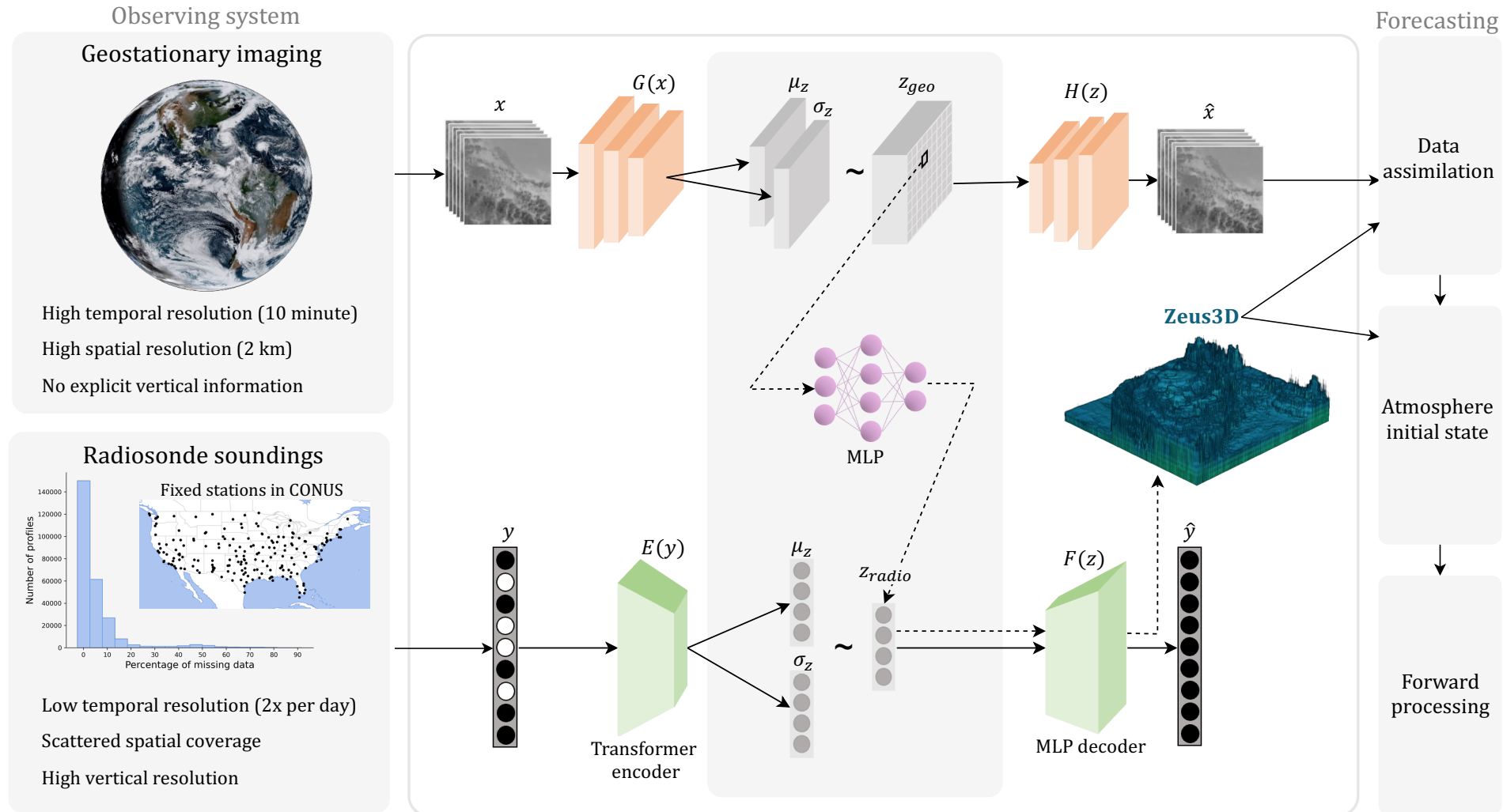
Speaker: TJ Vandal^{1,2,3}

Collaborators: Kate Duffy¹, Dan Holdaway⁴, Daniel McDuff¹, Rama Nemani³

¹ Zeus AI, Inc., ² NASA Ames Research Center, ³ Bay Area Environmental Research Institute, ⁴ NASA GMAO at Goddard Space Flight Center



Overview



Geostationary satellites

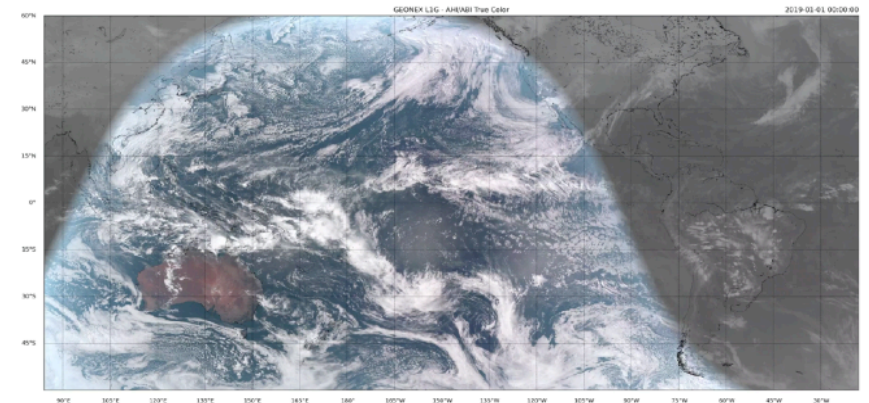
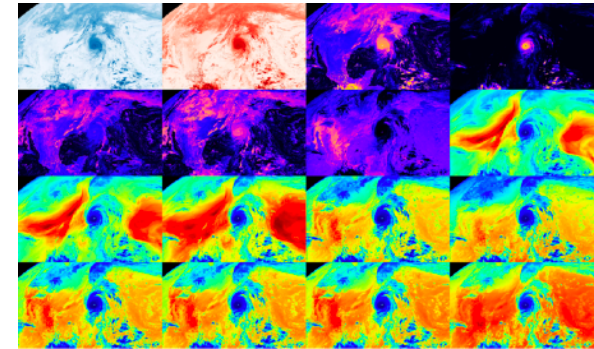
Specifications

- GEO Imaging sensors produce high-frequency multi-spectral data
 - GOES-16/17/18, Himawari-8/9, GeoKompSat-2a, FY-4
- Spatial: 0.5-2 km resolution
- Temporal: 1-10 minutes
- 5-16 spectral bands
- Global coverage from multiple satellites

Some Applications

- Atmospheric motion vectors (Velden 1997; Apke 2022; Carr 2019, ...)
- Quantitative precipitation estimation (Sadeghi 2019)
- Land surface temperature (Duffy 2022)
- Convective storm patterns, anvil plumes (Bedka 2021)
- Air quality (Kondragunta 2020)
- Vegetation (Hashimoto 2021)
- Wildfire detection (Xu 2017)
-

16 Visible and Infrared bands

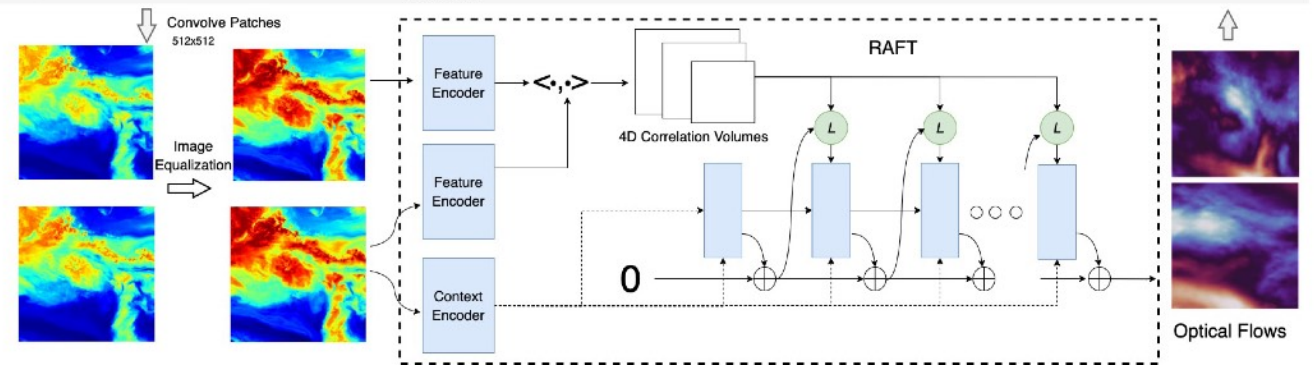
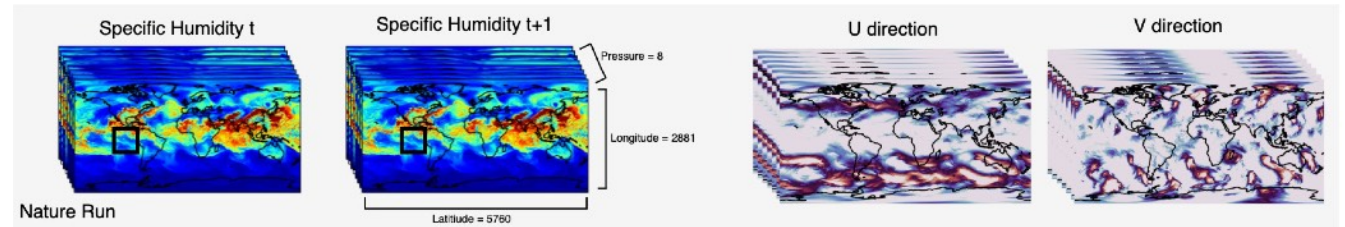


GeoNEX L1G dataset - Near global, 10 minute, 2km

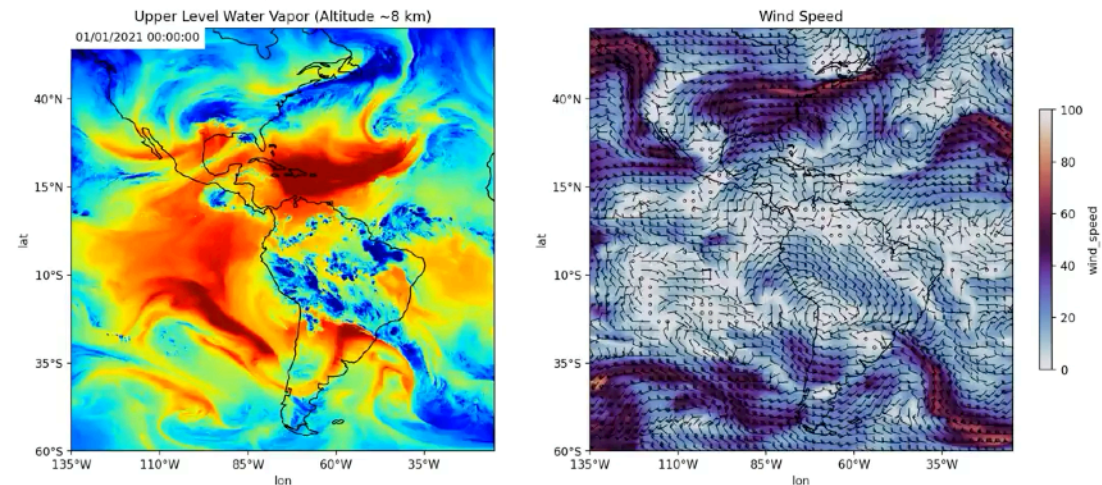
WindFlow

- ML for Atmospheric Motion Vectors (AMVs)
- Extract horizontal motion from sequences of humidity.
- Learn from Numerical weather simulations which are widely available as *synthetic* data. Apply to GEO.
- Flexible choice of optical flow architectures. Results compare four networks where we found RAFT to be the best performing.

$$Loss = ||Y-F(I_0, I_1)||_1$$



Apply to
Geostationary
satellite data.



GEOS-5 Nature Run: Flows across dimensions and scales

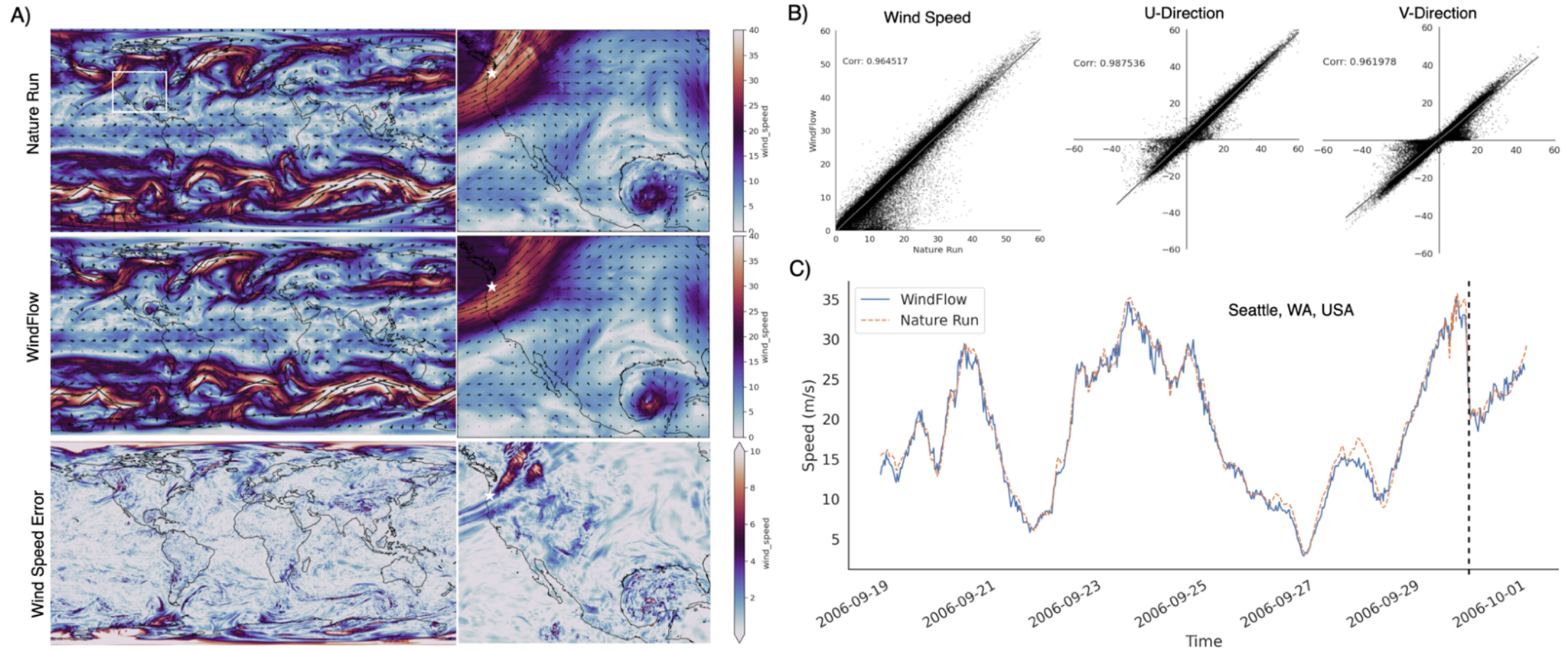
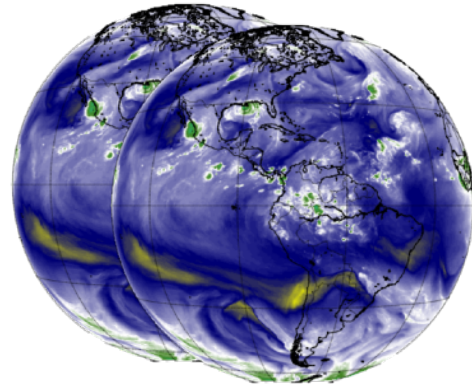


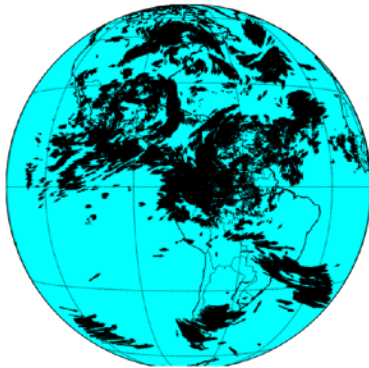
Figure: Results on nature run test set. A) Global and mesoscale comparison, B) U, V, and wind speed scatter plots, C) Time-series

Satellite based winds - Rawinsonde Comparison

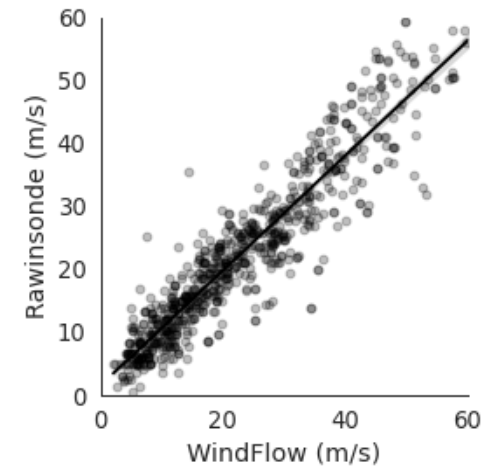
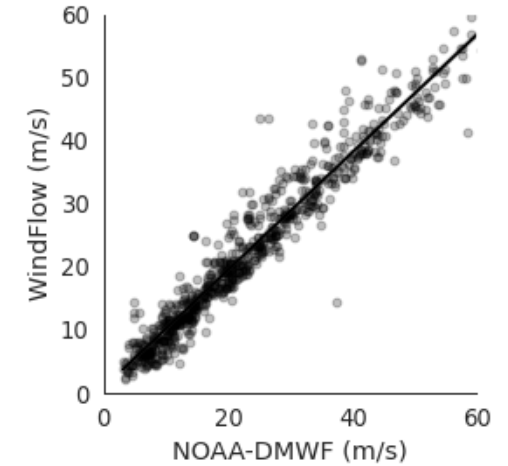
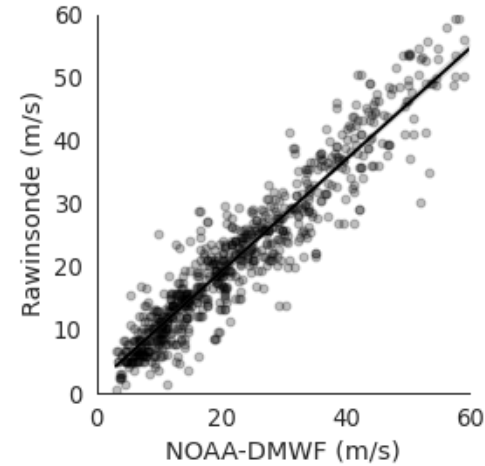
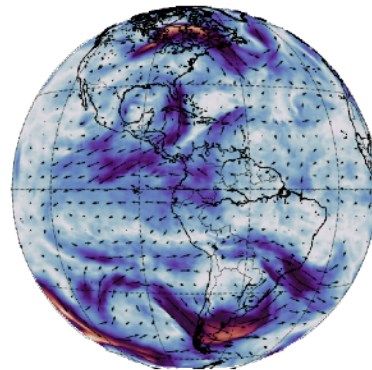
Input: Pair of GOES-16
Band 8 Infrared Water
Vapor



NOAA Derived
Motion Winds



NEX-AI WindFlow



Combine models for 3D Structure

Can we reconstruct vertical profiles from GEO imagery?

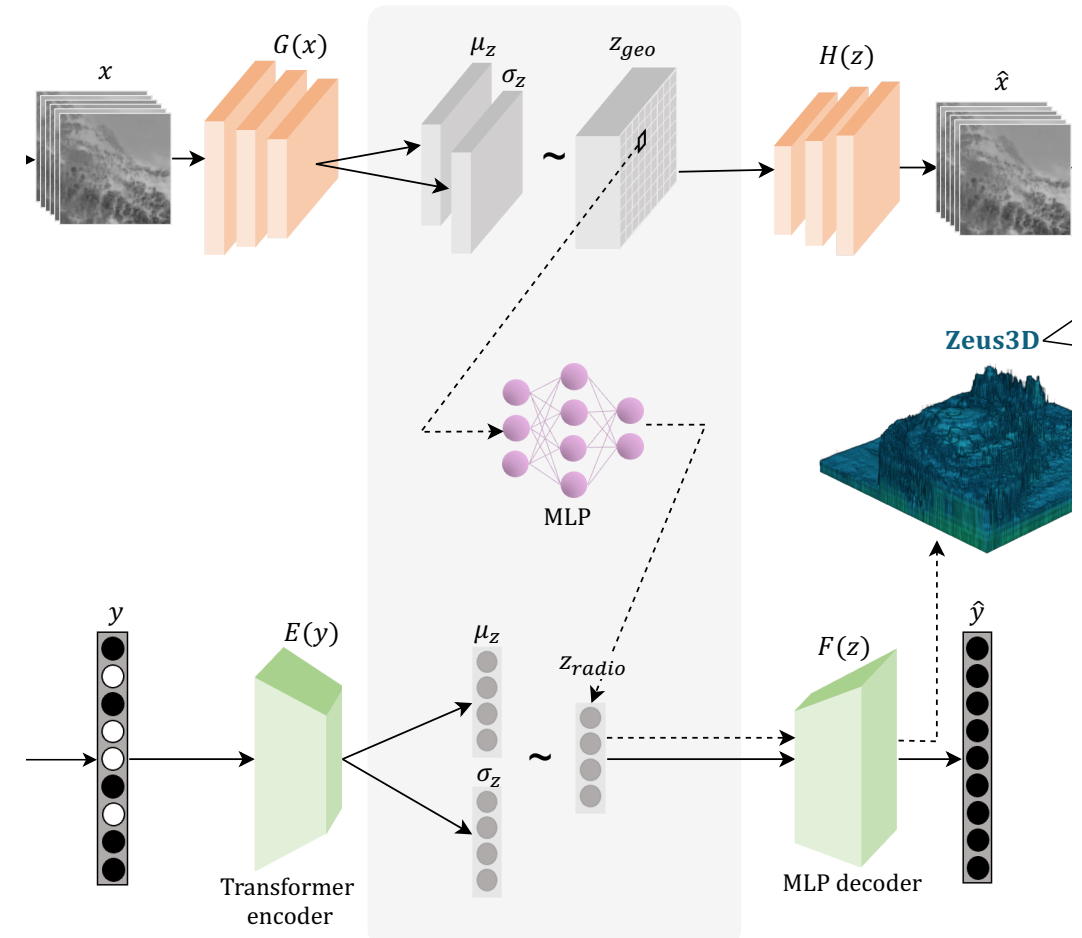
Scientific observations

1. Atmospheric profiles can be categorized into “clusters” representing different, well defined weather conditions.
2. Meteorologists can visually identify features in GEO TIR imagery for most, if not all weather conditions.

Hypothesis - A generative model can be learned to translate GEO TIR to radiosonde profiles

Approach - Learn a translation between low dimensional representations of GEO TIR and radiosondes

Variational Autoencoders (VAEs) compress high-dimensional data to a probabilistic latent representation, preserving uncertainty and extracting useful features

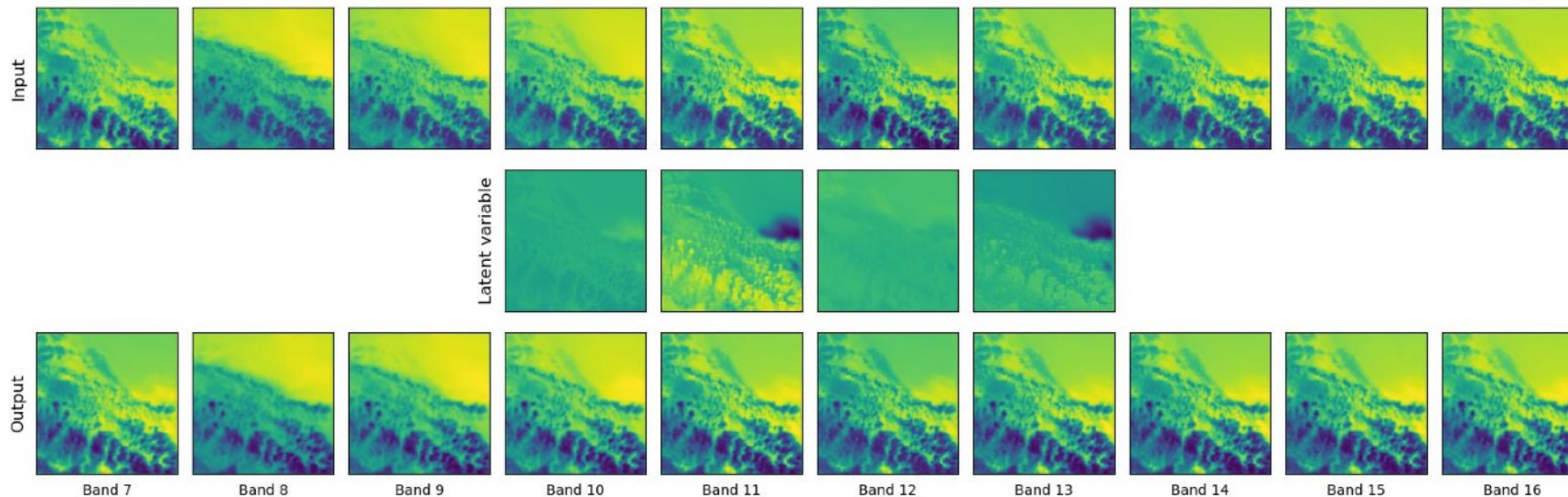
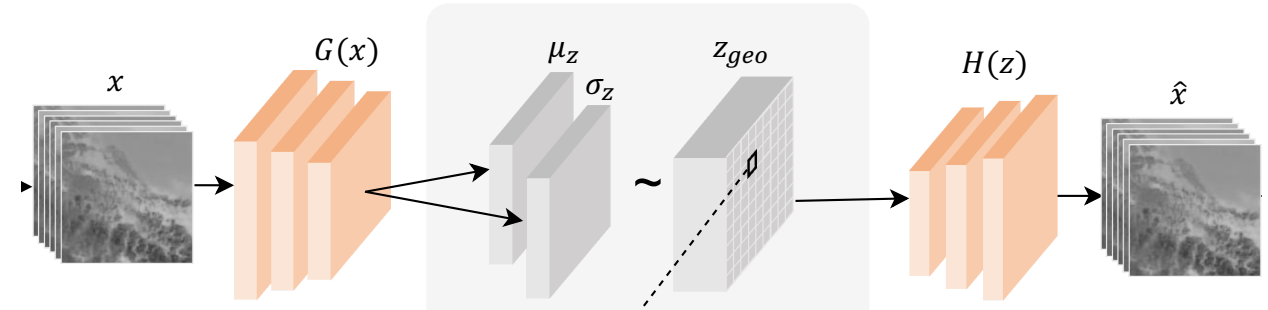


Geostationary VAE

High-fidelity VAE to learn a latent representation of GEO TIR

Compressed representation can reduce storage requirements

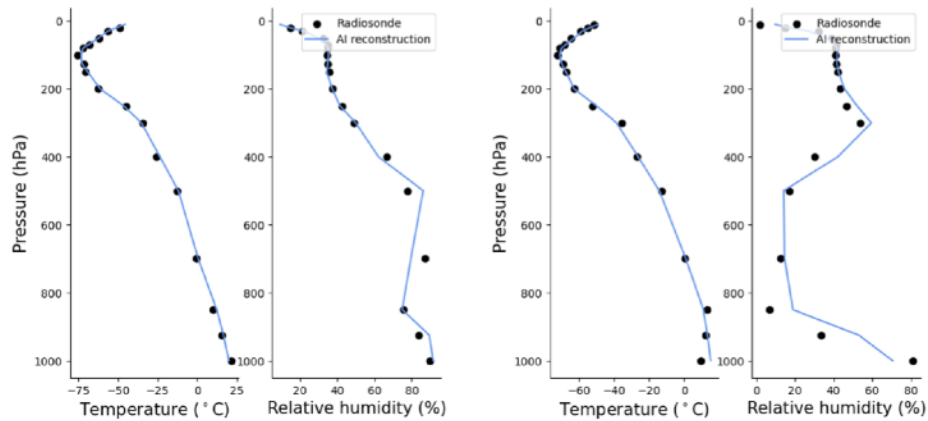
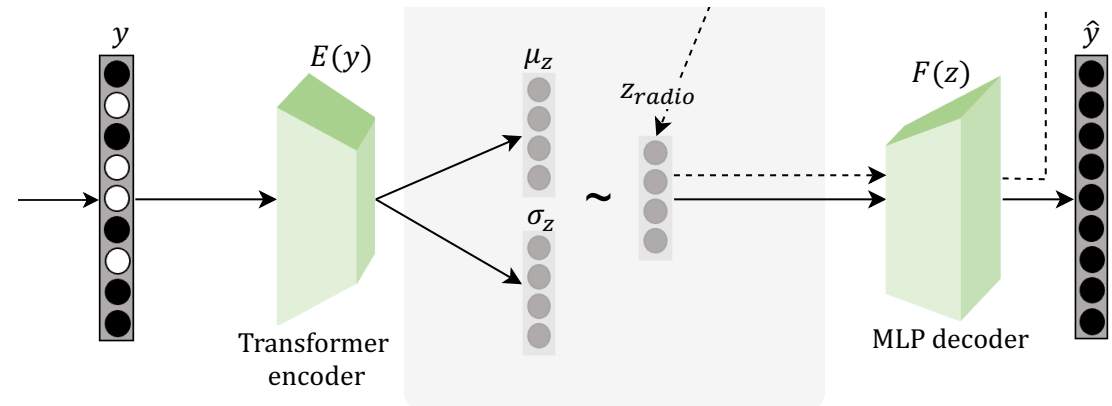
Trade-off between accuracy and compression ratio



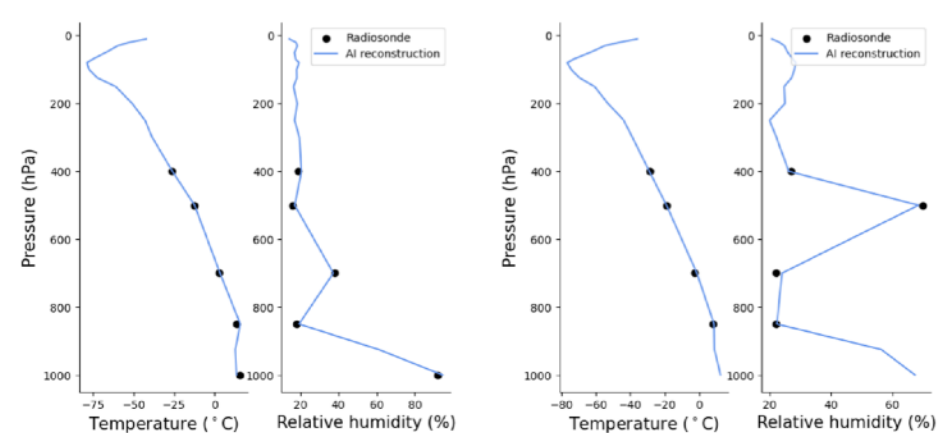
Radiosonde data reconstruction - VAE

- o VAE with transformer encoder layers and MLP decoder
- o Automatically fills missing values
- o Uncertainty aware sampling
- o Errors (dropping 20% of obs)

Temperature = 1.5° C, RH = 2.7%



Reconstructions



Gap Filled Profiles

Geostationary to 3D Profile

Predict radiosonde latent from GEO latent

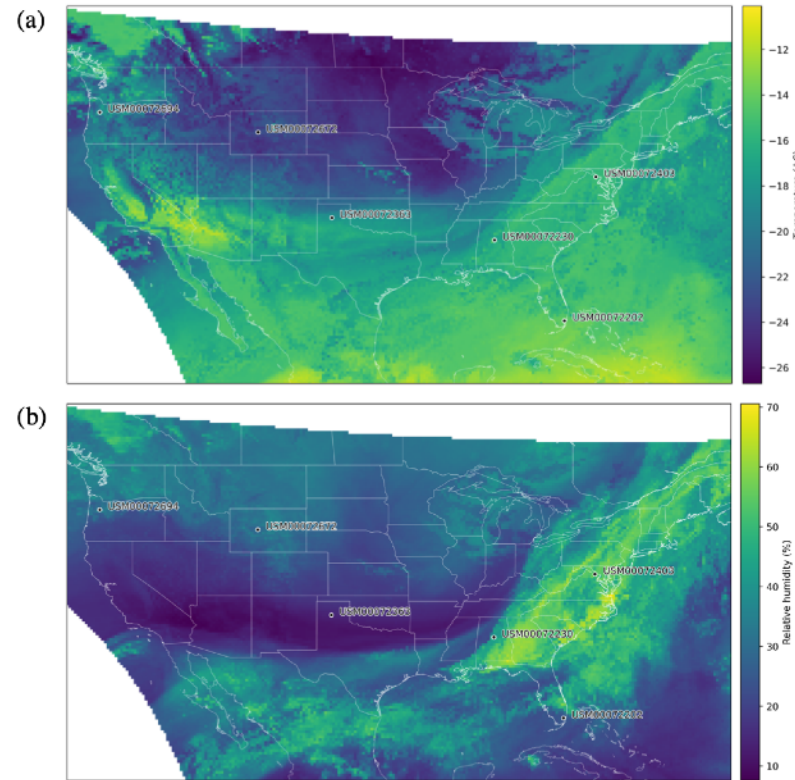
Train a model:

$$F(z_{geo}) = z_{rad}$$

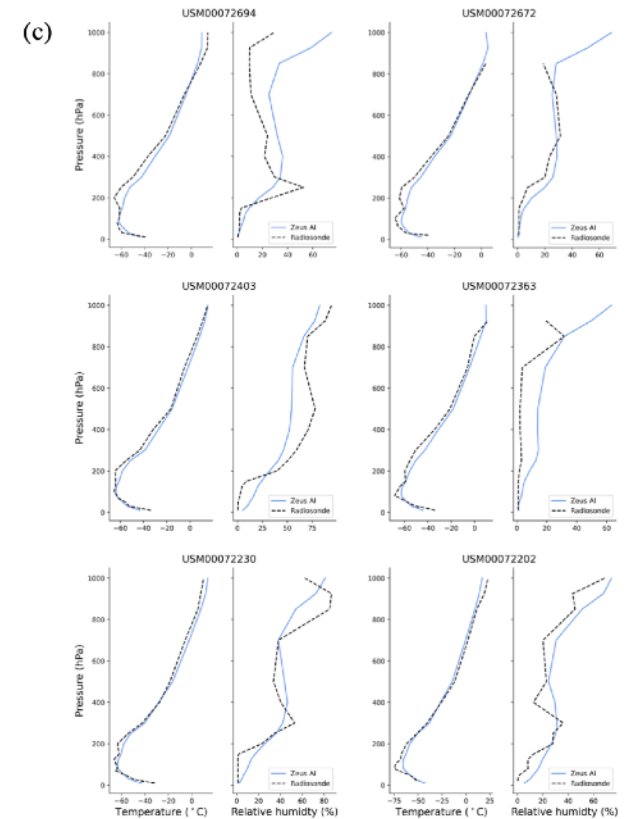
such that

$$z_{geo} = E_{geo}(GEO)$$

F is defined as an MLP with 4 input features and 9 output features



500 hpa a) Temperature and b) Relative Humidity Maps



c) Vertical profiles vs Radiosonde

Preliminary Analysis: Zeus AI vs ERA5

- + Improved spatial resolution and high-res atmospheric features
- + High-refresh rate every 10-minutes
- Underestimates warm core temperature

1:1 comparisons

Test set: January-April 2021

N = 1101

Relative humidity

RMSE(zeus, igr) = 15.54%

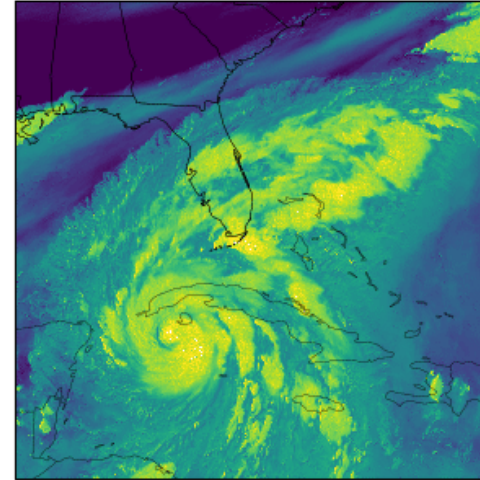
RMSE(era5, igr) = 18.980%

Temperature

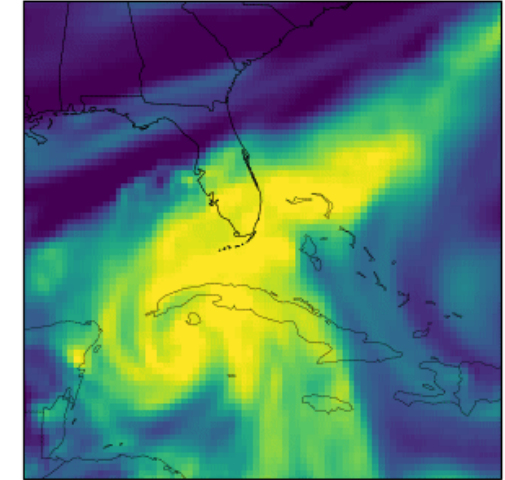
RMSE(zeus, igr) = 5.460 C

RMSE(era5, igr) = 3.364 C

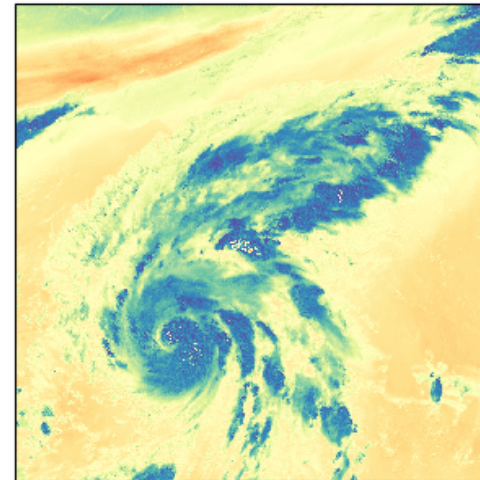
zeus relative humidity (%)



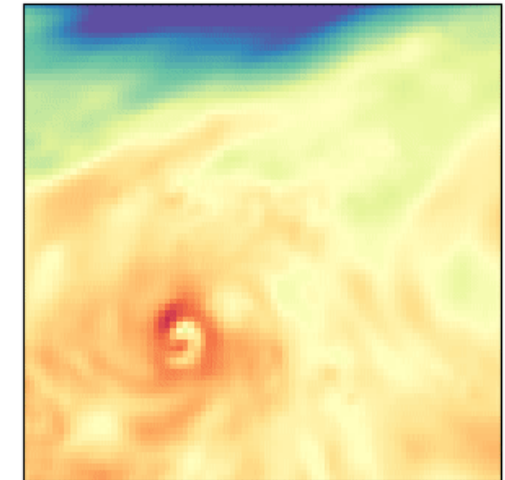
era5 relative humidity (%)



zeus temperature (C)



era5 temperature (C)



Preliminary Analysis: Zeus AI vs ERA5

1:1 comparisons

Test set: January-April 2021

N = 1101

Relative humidity

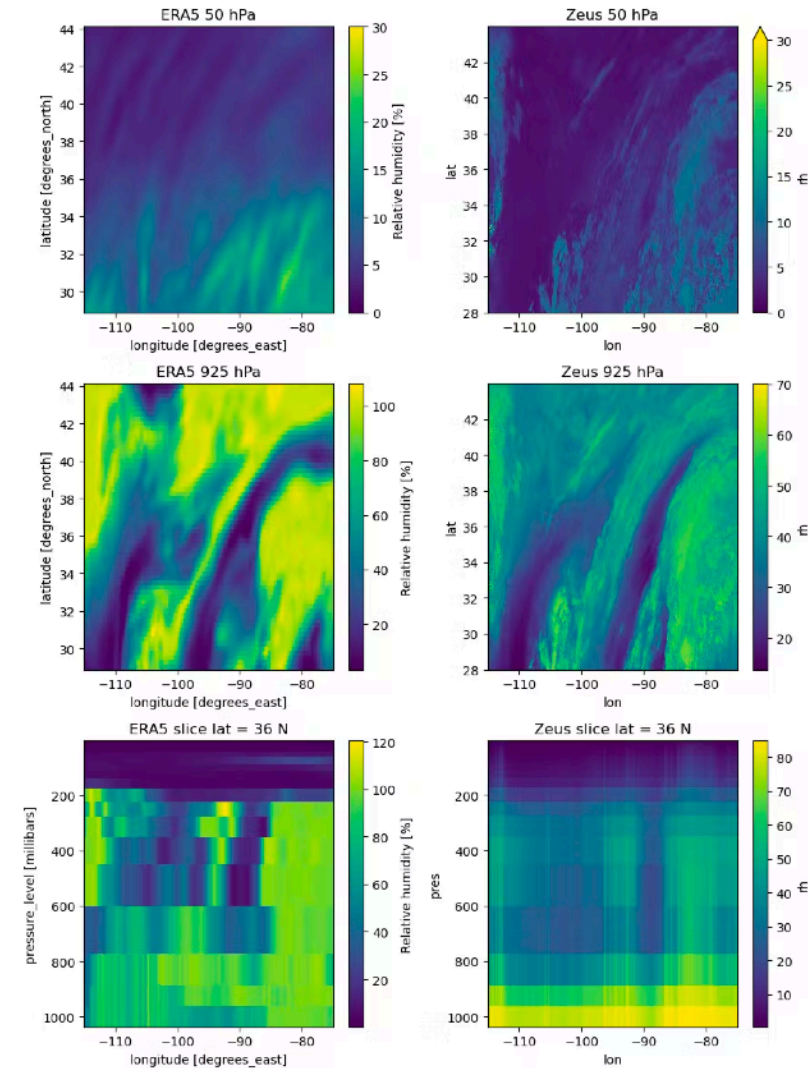
RMSE(zeus, igr) = 15.54

RMSE(era5, igr) = 18.980

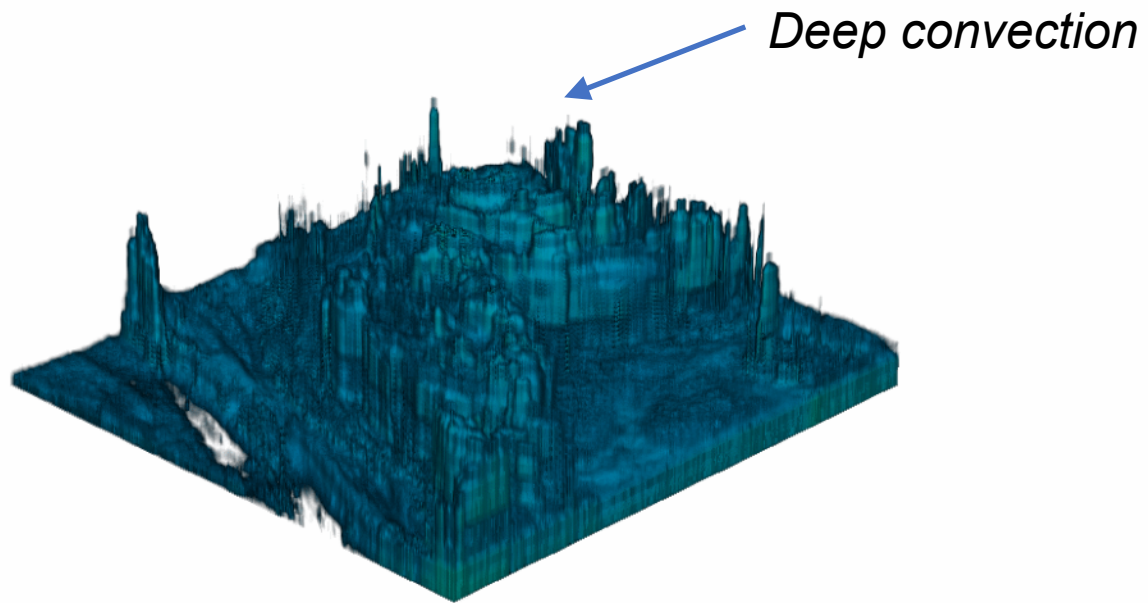
Temperature

RMSE(zeus, igr) = 5.460

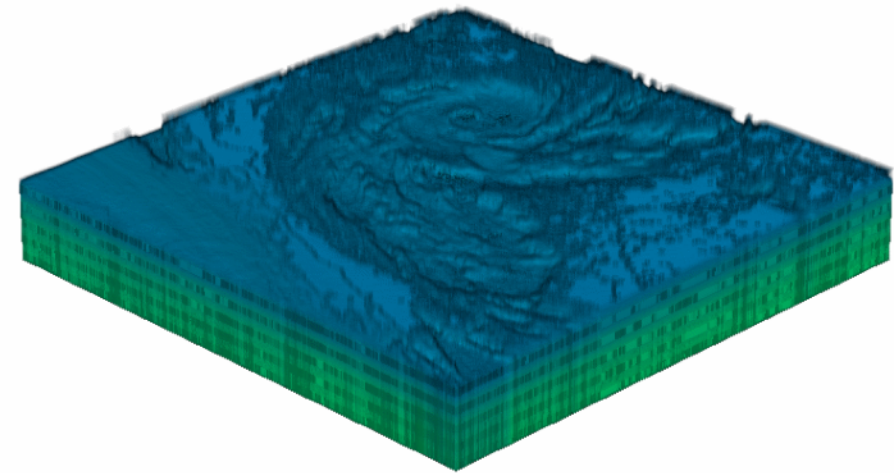
RMSE(era5, igr) = 3.364



3D Hurricane Ian



Humidity



Temperature

Future Work

- Apply WindFlow to retrieved from multi-level relative humidity sequences to produce 3D winds.
- Evaluate data products using triple collocation analysis against models and observations.
- Evaluation performance for extreme events and case studies.
- Develop Neural Ordinary Difference Equations (NODEs) to effectively assimilate and smooth our 3D product.
- Apply NODE into future as a novel forecast.

Open source software

Windflow

<https://github.com/tjvandal/windflow>

Includes code for a suite of optical flow models, training, and dependencies

Windflow-light

<https://github.com/tjvandal/windflow-light>

Minimal dependencies

Efficient inference, runs on a laptop

CALIPSO cloud height prediction (In progress)

<https://github.com/tjvandal/calipso-cloud-height-prediction>

Temporal Interpolation

<https://github.com/tjvandal/geostationary-superslomo>

```
import torch
import datetime as dt

from windflow import inference_flows
from windflow.datasets import goesr

# load model runner
checkpoint_file = 'model_weights/windflow.raft.pth.tar'
inference = inference_flows.FlowRunner('RAFT',
                                        overlap=128,
                                        tile_size=512,
                                        device=torch.device('cpu'),
                                        batch_size=1)

inference.load_checkpoint(checkpoint_file)

# load data
file1 = 'data/OR_ABI-L1b-RadC-M6C10_G16_s20222751101170_e20222751103554_c20222751103590.nc'
file2 = 'data/OR_ABI-L1b-RadC-M6C10_G16_s20222751106170_e20222751108554_c20222751108596.nc'

g16_1 = goesr.L1bBand(file1).open_dataset()
g16_2 = goesr.L1bBand(file2).open_dataset()

# Perform inference
_, flows = inference.forward(g16_1['Rad'].values,
                             g16_2['Rad'].values)
```

Acknowledgements and contact

Funding:

NASA Small Business Innovation Research Phase I and II, GSFC (2022-2025)

NASA ROSES Operational Geostationary Satellite program (2020-2023)

NASA Earth eXchange (NEX), GeoNEX project (2018-2022)

Data provided by the NOAA GOES-R Team

Computing partially provided by NASA's Pleiades Supercomputer

Email: tj@myzeus.ai

Website: www.myzeus.ai

