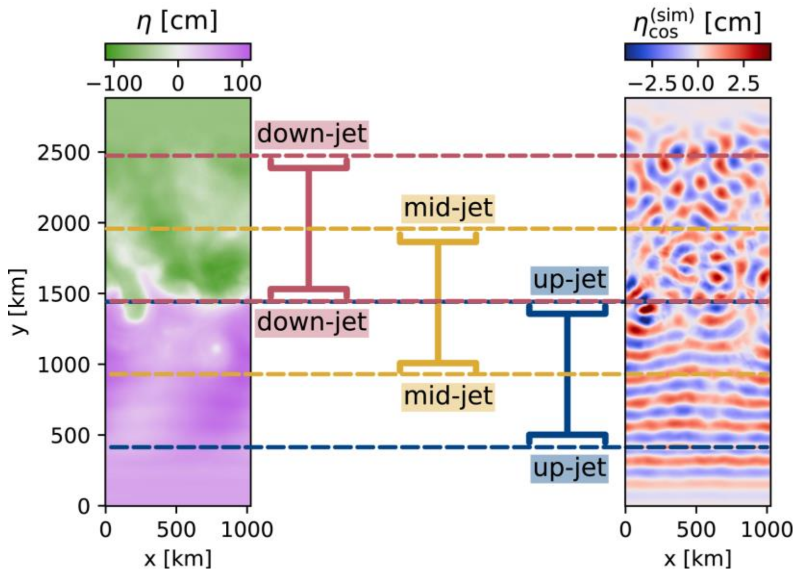
Can Deep Learning Disentangle Waves and Balanced flows from Sea Surface Snapshots?

UNIVERSITY OF TORONTO

Jeffrey Uncu, Nicolas Grisouard, Han Wang*

Motivation

The ocean sea surface height (SSH) η is a combination of balanced flows in the form of eddies and currents, and wave flows in the form of inertial waves and inertia gravity waves. Amongst the most energetic wave flows are the tides. The barotropic tide is phase locked to the astronomical forcing, allowing us to easily extract its signal from the SSH. However ,the baroclinic tide , or the internal Tide (IT) can become incoherent (not phase locked) near generation sites and in regions of strong balanced flow. The combination of this incoherence and the overlapping length scales over which eddy flow and ITs span (~150km in mid-latitudes) makes it challenging to distinguish them from satellite altimetry missions, such as the Surface water and Ocean topography Mission (SWOT), where temporal sampling will be too low to separate the two. The ability to separate the wave and balanced flow is crucial to infer ocean circulation from SWOT. **We use a generative deep learning algorithm on idealized data to disentangle wave and balanced flow given only a snap shot of SSH**.



Wang, H., Grisouard, N., Salehipour, H., Nuz, A., Poon, M., & Ponte, A. L. (2022). A deep learning approach to extract internal tides scattered by geostrophic turbulence. *Geophysical Research Letters*, 49, e2022GL099400. https://doi.org/10.1029/2022GL099400

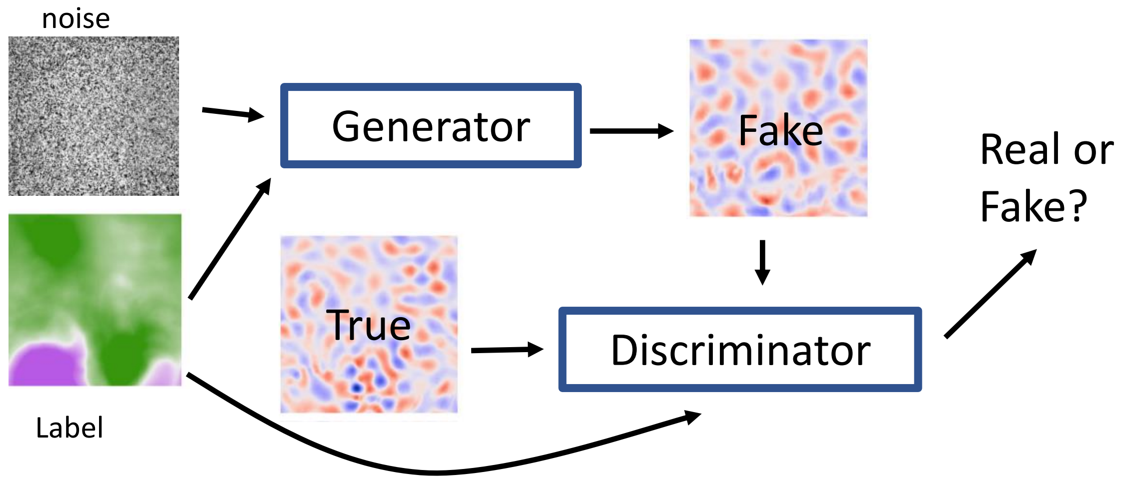
Data

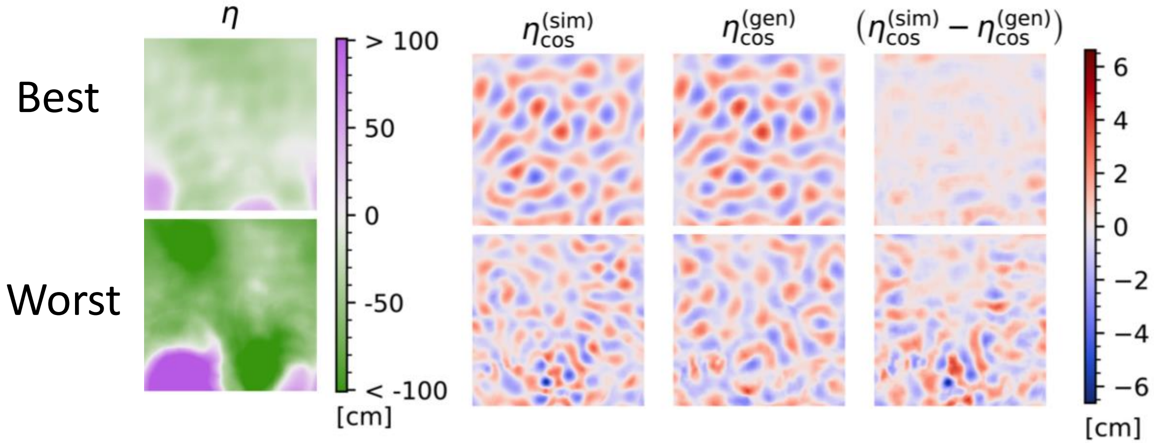
- Boussinesq simulations with 4 km horizontal resolution, 50 vertical levels.
- Incoming wave is forced through baroclinically unstable flow in the center of the domain.
- 5 different turbulence levels with 200 simulations per level

PixtoPix

- PixtoPix is conditional Generative Adversarial Network (cGAN)
- The domain is divided into top, middle, and bottom panel for
- Train on 80 %of the data, and test on 20%

PixtoPix cGAN Architecture





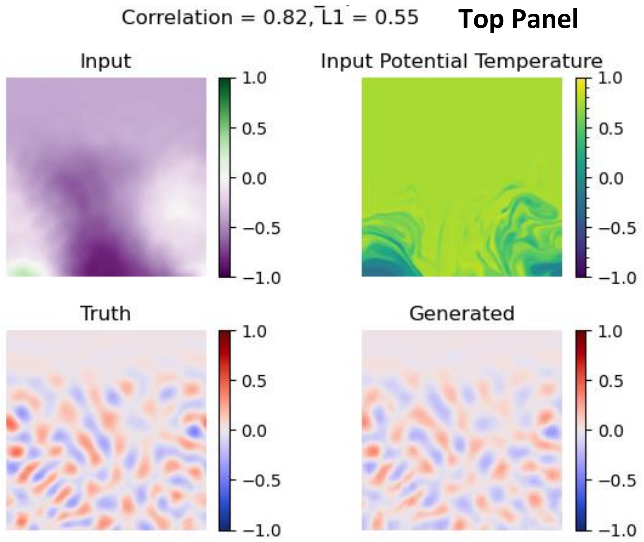
Base Performance $\eta \rightarrow \eta_{cos}$

Average correlation when training on all energy levels is 0.87. The best run is 0.95 and worst is 0.68. Generally algorithm does well for low energy levels but struggles with the highest of energy levels

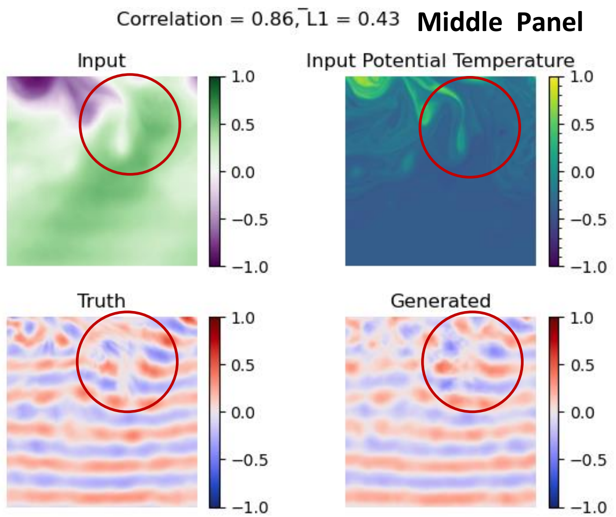
Improvements

- Normalize wave and vortex separately improves average correlation to **0.91**
- We add the temperature as an input into the algorithm
- Modifying the loss function to include overestimation of the spectrum

Worst Performers

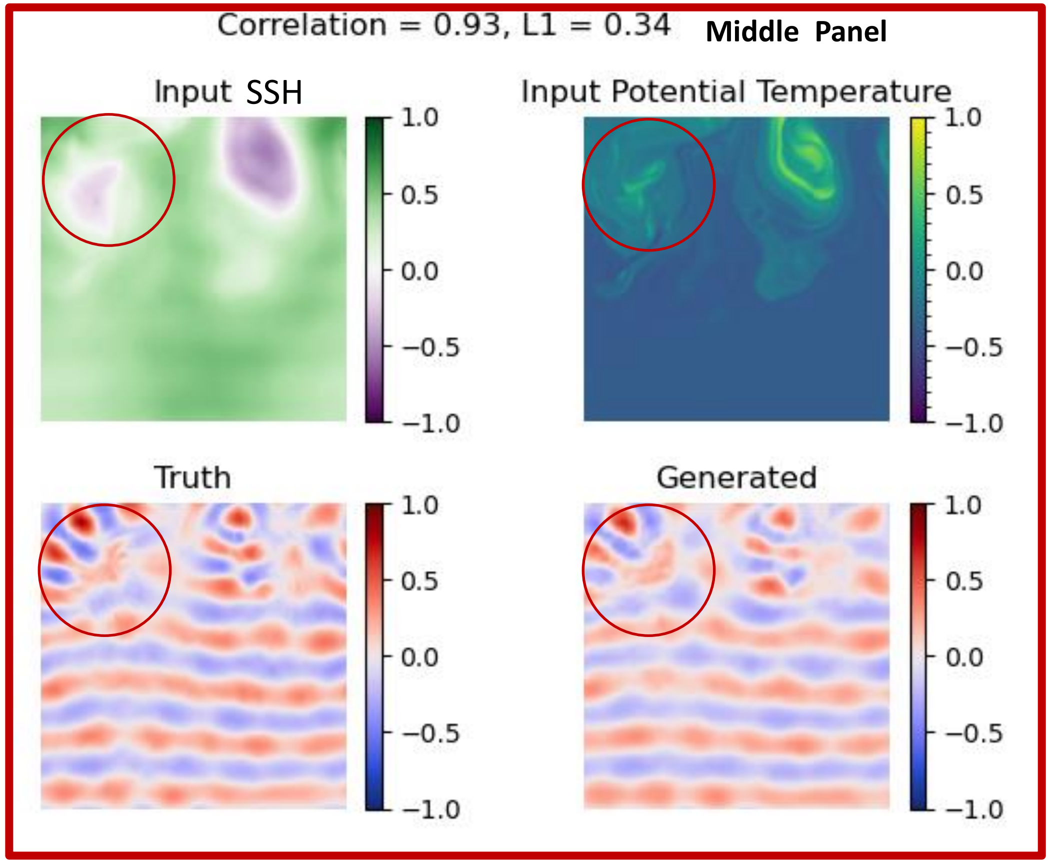


Highest turbulence levels cause the worse performance. Regions of strong balanced flows cause strong wave scattering. Sometimes these can look like dislocation in the wave front, which I believe is what the algorithm struggles with.



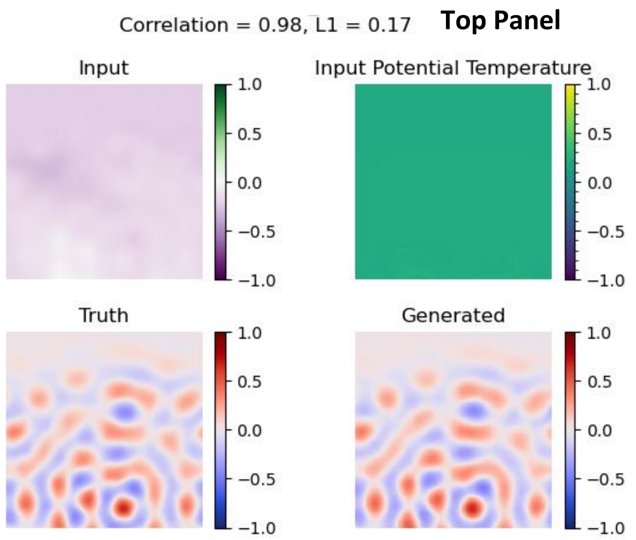
How Can Temperature Data Help Disentangle Balanced Flow?

- Eddy flows can be considered Quasi-Geostrophic. QG flow variables are uniquely determined by the potential vorticity.
- Ponte et al 2017 devised a method to approximate interior potential vorticity with sea surface density with fair results. We use temperature to approximate density changes, and add it as an input to our algorithm to improve results
- This method assumes IT signature on SST is weak

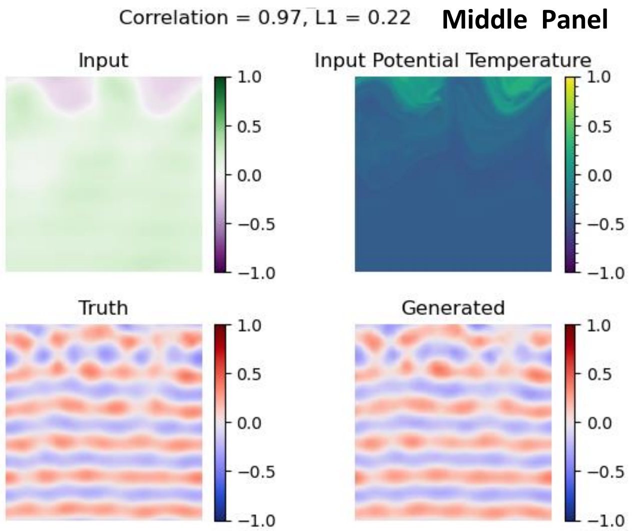


- **Overall correlation of 0.95**
- The middle and top panels are the most challenging given the increase in turbulent flow in the center, and the ensuing scattering pattern in the top panel.
- This shows better performance than the approximation of Ponte et al 2017 meaning there is more information than expected?
- Limitations are that the eddies need to form due to temperature gradients. Is attaining SST to this resolution realistic?

Best Performers



In the top of top panel, the turbulent flow is mostly gone, yet we see that the algorithm is able to predict the scattering pattern very well from the balanced flow at the bottom of the panel.

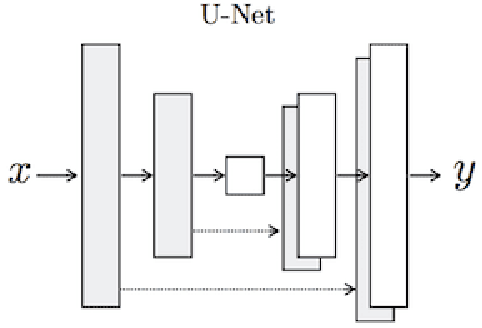


Looks good… But how does the spectrum look?

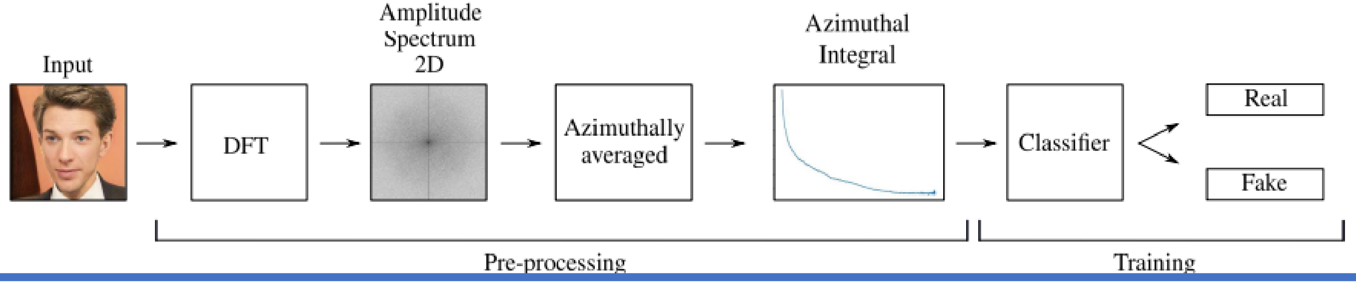
The generator is a U-net architecture which works by reducing the high dimensionality of the input space into a low dimensionality (down-sampling), and then using these few parameters to up-sample to the generated image. This up-sampling process has a by-product of introducing high frequency noise. This lead to a common problem where cGANs to overestimate the power spectrum of an image. This is especially important in physical applications where we would like to have conservation of energy and a fair representation of the cascade of energy among scale. In order to do so we follow a process outline by (Durall et al 2020) in which add a term to the loss function which penalizes the algorithm to obey the original spectrum .

G_L: Gan Loss , S_L: Spectral Loss, λ: spectral hyperparameter **Total Loss:** $\boldsymbol{T_L = G_L(1 + \lambda S_L)}$

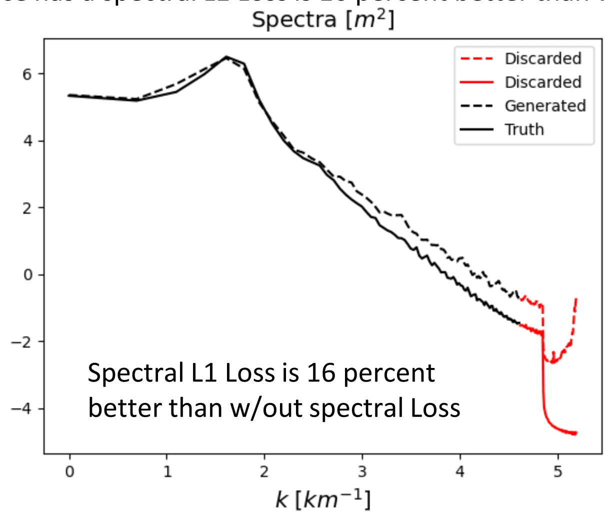
A Gentle Introduction to Pix2Pix Generative Adversarial Network - MachineLearningMastery.com

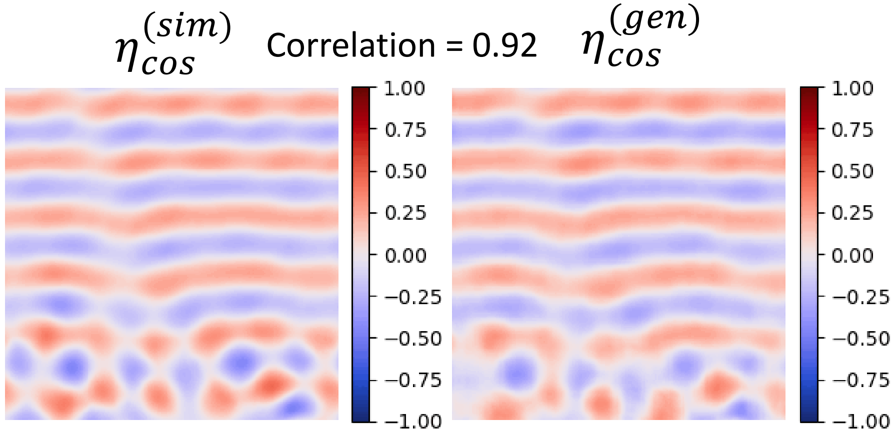


Spectral Loss Algorithm



Below are pictures of the generated spectra and images of the same simulation with and without the spectral loss function. This specific instance has a spectral L1 Loss is 16 percent better than w/out spectral Loss. This produce also is expected to increase stability of the algorithm.





	$\eta \rightarrow \eta_{cos}$	$\eta + T \rightarrow \eta_{cos}$	$\eta \rightarrow \eta_{cos}$ (Spectral loss)
Correlation	0.91	**0.95**	0.92
Spectral	-	?	**5 % better**

Conclusion

- cGANs have shown to have incredible generative power to disentangle waves and balanced flows even in regions of strong incoherence
- We have improved on past performance by introducing new normalization, adding spectral loss function, and including temperature in the input.
- We have found (not shown here) that the algorithm is able to extrapolate past the energy levels it is trained on reasonable well (correlation = 0.89).

Future Work

- Combining spectral loss and temperature field methods
- Change spectral loss calculation to to conserve energy
- Diffusion Learning (alternative Generative Models)
- Han Wang is working to apply this algorithm to HYCOM

Estiamte PV from surface density.
Inverty PV to geostrophic stream function

$$q=\nabla^2\psi+\partial_z\left(\frac{f_0^2}{N^2}\partial_z\psi\right),$$

$$q=\partial_y\bar{v}-\partial_x\bar{u}-\partial_z\left(\frac{gf_0(\bar{\rho}-\rho_{\text{ref}})/\rho_0}{N^2}\right).$$

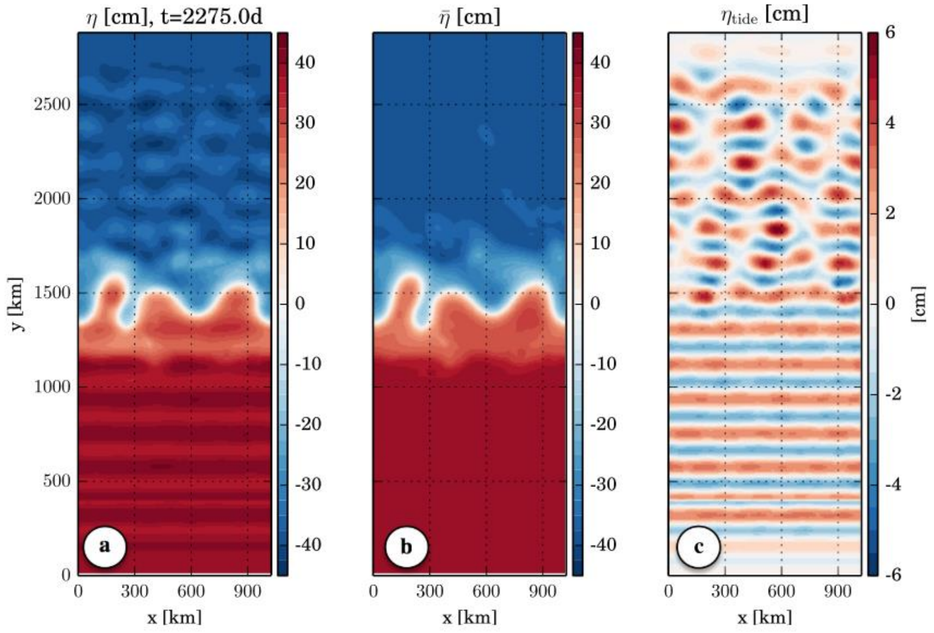
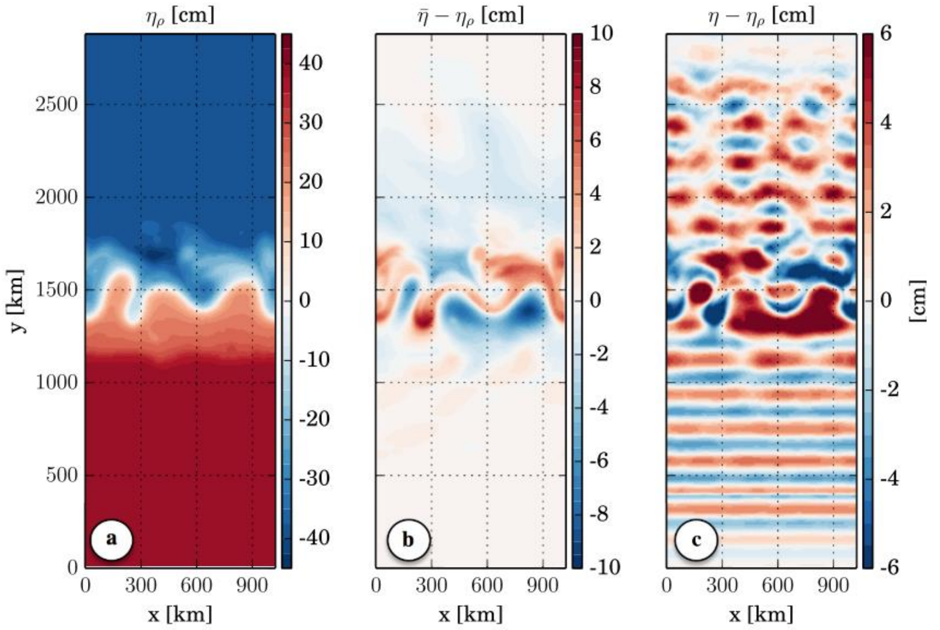
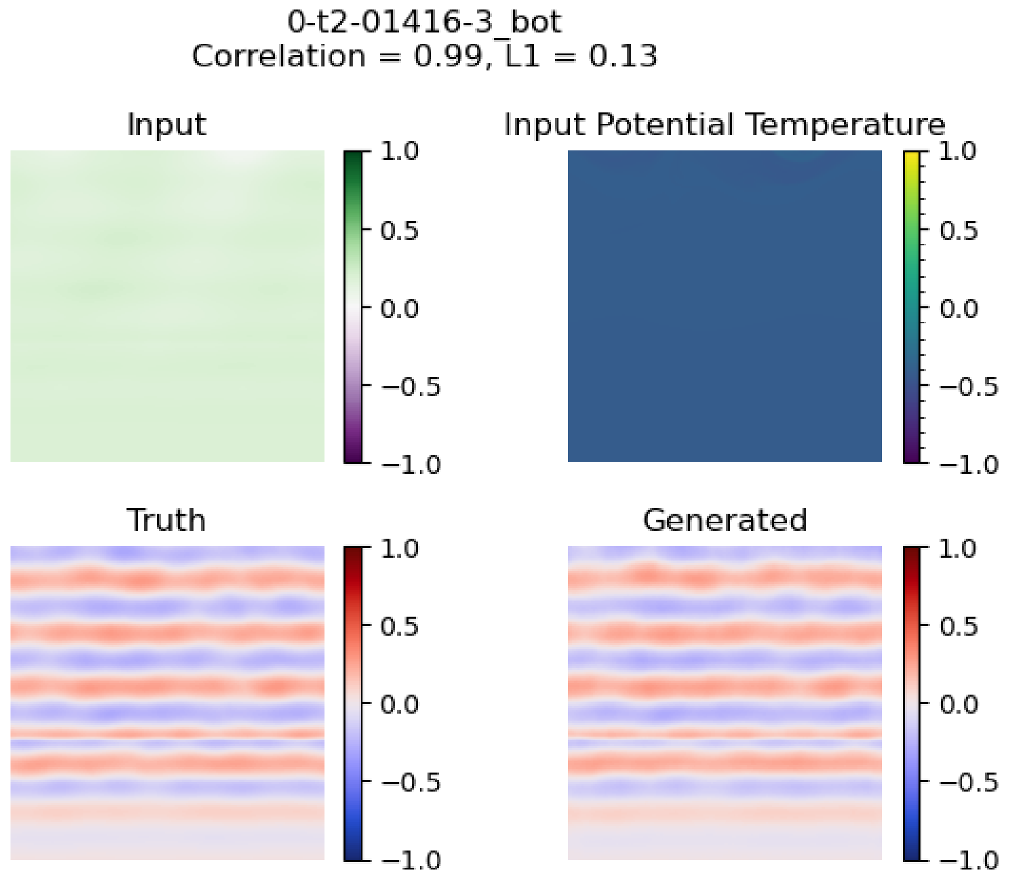


Figure 2. (a) Instantaneous, (b) 1 day averaged, and (c) tidal sea level.




0-t2-01416-3_bot
Correlation = 0.99, L1 = 0.13
Input
Input Potential Temperature
Truth
Generated
1.0
0.5
0.0
−0.5
−1.0