A Hybrid Deep Learning Model for Improved Wind and Wave Forecasts

Yuval Yevnin[1]**,Yaron Toledo**[1]

[1]School of Mechanical Engineering, Tel-Aviv University, Tel-Aviv 69978, Israel

Key Points:

- A deep learning recurrent-convolutional model to improve wind and wave forecast is designed. The model is trained to improve wind forecast based on past reanalysis data. The resulting improved wind field prediction is used as an input for the wave forecasting model.
- Even without prior physical knowledge, the model manages to improve both wind and wave forecasts RMSE by ∼10% over the Mediterranean, and ∼35% over the Aegean Sea during Etesian wind.
- The presented model has negligible additional computational costs, and can be generalized to a global grid or specialized to a high-resolution local grid.

Corresponding author: Yuval Yevnin, `yuval.yevnin@gmail.com`

Corresponding author: Yaron Toledo, `toledo@tau.ac.il`

Abstract

The paper presents a hybrid numerical - deep learning (DL) approach for improving wind and wave forecasting. First, a DL model is trained to improve wind velocity forecast by using past reanalysis data. The improved wind forecast is used as a forcing for WAVEWATCH III numerical wave forecasting model. This novel approach to combine physics-based and data-driven models was tested over the Mediterranean. It resulted in root mean squared error (RMSE) of ∼10% lower in both wind velocity and significant wave height forecasts over standard operational models. This significant improvement is even more substantial when examining the local region of the Aegean Sea during May to September, when the Etesian wind is dominant, improving wave height forecasts RMSE by over 35%. The additional computational costs of the new DL model are negligible compared to the costs of either numerical models. This work has the potential to revolutionize the weather forecasting models used nowadays by tailoring models to localized seasonal conditions, with negligible additional computational costs. The derived methodology can also be applied to various other fields, where the deep learning model can learn to predict measured or simulated results from an initial, less accurate model.

Plain Language Summary

Modern wave forecasting originated in the D-Day invasion, while attempting to predict the optimal date for departure. In the decades since it has come a long way, and currently forecasting models are sets of complicated, physics-based equations. Similar, and even more complex models are used to make wind forecasts, which are needed as inputs for the wave models. This work presents a deep learning model improving the wind forecast, and consequently improving also the wave forecast. The novel approach of combining deep learning and classical forecasting models was tested over the Mediterranean Sea, and resulted in ∼10% improvement in both wind and wave forecasts over current operational model. This significant improvement is even more substantial when examining the local region of the Aegean Sea during May to September, when the Etesian wind is dominant, improving wave height forecasts by over 35%. This work has the potential to revolutionize the weather forecasting models used nowadays by tailoring models to localized seasonal conditions, with negligible additional computational costs. The derived methodology can also be applied to various other fields, where the deep learning model can learn to predict measured or simulated results from an initial, less accurate model.

1 Introduction

Wind velocity accuracy has been established as one of the most significant factors in achieving an accurate ocean waves forecast (Bidlot et al., 2002). For this reason, operational wave forecasting models aim to use the most accurate wind fields available, with a high resolution in both space and time. The models producing the wind fields are highly computationally expansive, simulating many layers in the atmosphere. The results of the wind models are later reanalyzed to assimilate measurements taken by various instruments such as satellites, radars and point measurement devices. The reanalysis data is used to assess, study and improve the forecast ability (Hersbach et al., 2020).

Traditionally, wave forecasting models, such as WAM (Hasselmann et al., 1988), WAVEWATCH III (Tolman, 1991) or SWAN (Booij et al., 1999), use wind forecast as an input. Although the driving force for wave generation is surface wind, the parameter used by most models is wind velocity at $10m$ above the sea surface (U10), as this property is easier to measure and predict. This means only a single property at a single level of the atmospheric model actually affects the wave model. A semi-empirical source term is used by wave models to convert U10 to wave action forcing (Janssen & Janssen, 2004; Ardhuin et al., 2010). Optimizing atmospheric models is highly complex, both in terms of computational costs and in terms of improved physical equations accounting for multiple flow parameters. Thus a model which can optimize U10 independently, decoupled from the physics-based model and with low computational costs is very desirable.

In the last few years deep learning (DL) models have been used in multiple fields to solve complex, highly nonlinear problems (Wang et al., 2019; Brunton et al., 2020). These DL models are data-driven, meaning they generally do not possess any prior physical knowledge, but are instead trained to predict a given "ground-truth" data. After the model is trained using a training dataset to achieve good performance, it is verified over an independent test dataset. The training process usually requires more significant computational resources, though still relatively small compared to numerical models. Afterwards, the resulting model can be used to produce accurate predictions at very minimal computational cost. As was shown in (Reichstein et al., 2019), these methods are highly relevant for geophysical problems, and have already been used to make independent, data-driven wind forecasts (Scher & Messori, 2019; Weyn et al., 2019, 2020; Rasp & Thuerey, 2020; Rasp et al., 2020; Arcomano et al., 2020).

The presented paper uses a DL model with a U10 wind velocity forecast as an input, and predicts the reanalysis data, considered "ground-truth". This improved wind prediction is used as an input to a numerical wave model. Unlike previous works, the current model focuses on improving forecast produced by a numerical atmospheric model, and thus is able to achieve much higher accuracy. To the best of our knowledge this is the first attempt to create such a hybrid numerical - deep learning model.

2 Model Database - ECMWF Wind Velocity

The datasets used in this paper are ECMWF Era5 single-level forecast (FC) and reanalysis (REAN) databases (Hersbach et al., 2020), with the parameters of wind velocity vectors in the zonal and meridional directions at $10m$ height $(u10, v10)$. The FC data was used as the deep learning model's (DLM) input and the REAN as the "ground-truth". The FC is initiated from a wind analysis every 12 hours at 06:00 and 18:00, and consists of 18 hourly steps. This means there is an overlap between consecutive forecasts. In this work the time steps 7-18 were chosen, as these had the largest errors. The REAN data is an hourly high-resolution model incorporated with measurements.

The spatial grid chosen was of the Mediterranean region, with longitude between $30.2-45.7N$ and steps of $0.5N$, and latitude between $-2.1-36.0E$ and steps of $0.3E$. This results in a base 2 grid of dimensions 32×128, making it efficient for processing with a DLM.

3 Recurrent-Convolutional Model

In Roitenberg and Wolf (2019) a general DLM architecture for spatio-temporal forecasting problems was introduced and tested for public transportation demand. This model was used as a base for a new DLM, by removing the encoder and making several adjustments to the decoder part (Fig. 1). The new DLM begins with an input sequence of FC instances. Next is an encoder comprised of convolutional layers with gradually increasing width and dilation. Increasing the width allows each layer to capture more information, while larger dilation allows a wider receptive field taking into account the effects of farther spatial information. Using dilation instead of more traditional approaches of strided convolution or pooling layers keeps the original input dimensions, and thus prevents spatial information loss (Yu & Koltun, 2015).

Following the encoder, Convolutional Gated Recurrent Unit (CGRU) (Ballas et al., 2015) layers were used. These layers combine the ability of the GRU layer (Chung et al., 2014) to learn temporal connections with the convolutional layer capability of spatial modelling. This is done by replacing the matrix multiplication of a GRU with a convolution, and the parameter matrices and vectors with smaller kernels. Each instance of the input sequence is introduced separately to the encoder and to the following CGRU, and the last output of the CGRU is concatenated with the last input instance into it. This forms a skip connection over the CGRU, allowing both to bypass it where needed, and to add a residual to improve it. Using residual connections was shown to be extremely effective in improving the learning ability of the neural networks compared to modelling absolute values (Littwin & Wolf, 2016).

Finally, the new decoder consists of convolution layers mirroring the structure of the encoder in width and dilation. The output of the decoder was summed with the last input instance to the model, forming another residual connection.

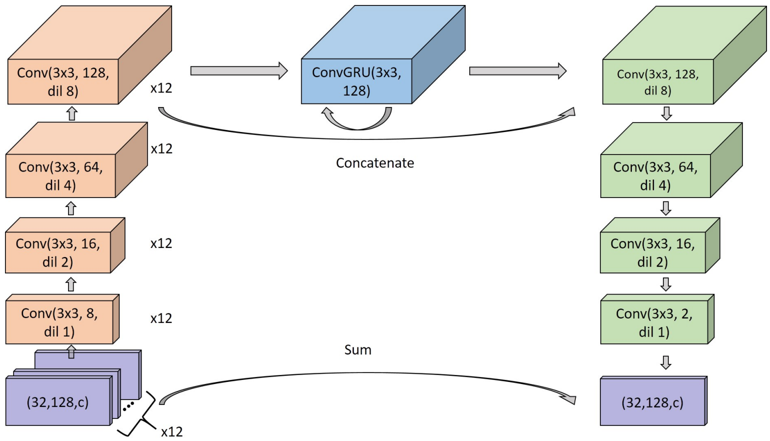


Figure 1. Model architecture from bottom left: input (purple) in the form of a sequence of FC instances with c channels (variables) is passed one at a time to the encoder (orange), comprised of convolutional layers with increasing filters and dilation. The output of the encoder is fed to a CGRU (blue). The last output of the resulting sequence is concatenated with the last input into it, and introduced to the decoder (green), comprised of convolutional layers mirroring the encoder. The final result is summed with the last instance of the input sequence to form a residual connection (purple).

4 Deep Learning Wind Prediction Experiments

Four types of wind input to the DLM were tested for effectiveness in producing a better wind input for wave forecasting. The input data for all experiments consisted of 12 consecutive hourly time steps from the FC dataset. The target was the REAN at the time of the last input. This effectively means improving the wind field at a given time t by using time steps $(t-11, t)$. The network hyper-parameters were initially set to those of Roitenberg and Wolf (2019). A short training period of the years $2010 - 2011$ and validation period of the year 2012 was used to test changes to the architecture. Due to long run times even for these short periods, an extensive architectural grid search was not conducted. The chosen architecture (shown in Fig. 1) consisted of a four convolutional layers encoder with $(8, 16, 64, 128)$ filters and a dilation of $(1, 2, 4, 8)$, followed by a single CGRU layer with input and output dimensions of 128. The decoder consisted of four convolutional layers with $(128, 32, 16, 2)$ filters and $(8, 3, 2, 1)$ dilations. The datasets were split to a train / validation / test sets with the following temporal range:

1. A training set between the years $2001 - 2016$

Table 1. *Wind velocity RMSE*

Model	Property	DLM RMSE	FC RMSE	RMSE improved
UMag, sec. 4.1	$U[m/s]$	0.5999	0.6673	10.1%
	$u10[m/s]$	0.7075	0.7291	2.97%
	$v10[m/s]$	0.7065	0.7278	2.88%
UVec, sec. 4.2	$U[m/s]$	0.615	0.6673	7.8%
	$u10[m/s]$	0.6616	0.7291	9.26%
	$v10[m/s]$	0.6594	0.7278	9.39%
UDir, sec. 4.3	$\cos\theta$	0.2307	0.2469	6.55%
	$\sin\theta$	0.229	0.2463	7.04%
	$u10[m/s]$	0.6906	0.7291	5.28%
	$v10[m/s]$	0.69	0.7278	5.19%
UFrc, sec. 4.4	$U[m/s]$	0.6162	0.6673	7.65%
	$u10[m/s]$	0.663	0.7291	9.06%
	$v10[m/s]$	0.6613	0.7278	9.14%

2. A validation set of the year 2000
3. A test set of the year 2017

The DLM was trained and evaluated using an NVidia GeForce GTX 2080 Ti GPU with 12GB memory. The Fastai API (Howard & Gugger, 2020) was used with Pytorch API as a base. The model was optimized using ADAM (Kingma & Ba, 2015). Weight decay was set to $1E-3$, and the mini-batch size was 16. A changing learning rate with the 1-cycle approach of (Smith, 2018) was used, and each model was trained for 8 cycles of 2 epochs. The max learning rate started at $1E-3$, and divided by the cycle number as learning progressed. After training, the validation set was used to identify the cycle with best performance. The weights of this cycle defined the new DLM, and its performance was evaluated on the test set. The resulting RMSE in space and time of all wind input types are shown in Table 1 and compared to the original FC data.

4.1 Input type 1: Wind Velocity Magnitude

The first experiment optimized prediction of wind velocity magnitude (UMag), defined as $U = \sqrt{u_{10}^2 + v_{10}^2}$. The magnitude was chosen as it seemed easier to predict, being always positive, non-directional and independent property in space. This resulted with input and output tensors with dimensions of $(time = 12, c = 1, lat = 32, lon = 128)$. The resulting U was also transformed back to the form of $u10$ and $v10$ using the original FC direction. As expected, U improved significantly, as it is the main objective of the UMag DLM. It is interesting that the resulting $u10$ and $v10$ are improved by a much smaller percentage.

4.2 Input type 2: Wind Velocity Vector

The second experiment was performed to test the DLM's ability to improve the wind velocity vector (UVec) directly. The input was set as the FC $u10$ and $v10$, and the output as the matching prediction, resulting with $(12, 2, 32, 128)$ tensors. Although the improvement in the main objective of each DLM is smaller, the resulting wind vector improvement is almost three times as much as that of the UMag model.

4.3 Input type 3: Wind Direction Vector

The third experiment was predicting the direction of the wind velocity vector (UDir). The directional unit vector was defined as

$$\begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix} = \begin{pmatrix} u_{10}/U \\ v_{10}/U \end{pmatrix},$$

and was set as both the input and output of the DLM. The test set output was multiplied by U to produced a wind velocity vector. Examining the results of this DLM found it similar to the UVec model with smaller improvement.

4.4 Input type 4: Wind Friction Velocity Vector

Lastly, an experiment was done to try and make a connection between a physical wave forecasting model and the DLM for wind prediction. The wave model uses the wind input through a source term (ST) which converts it to wave energy. Such ST combine analytical and empirical derivations, with a varying degree of complexity. The relatively simple wind friction velocity vector (UFrc) of WAM 3 (WAMDI Group, 1988)

$$\mathbf{u}_* = \begin{pmatrix} u_{10}\sqrt{0.8 + 0.065u_{10}} \\ v_{10}\sqrt{0.8 + 0.065v_{10}} \end{pmatrix},$$

was used in the DLM cost function. which should make it better fitting as an input to the ST. This still lacks the local wave action spectrum used in the source term, but as they are the result of an independent model with high computational cost, such a coupled model was not tested. This DLM's results were almost identical to the UVec model.

5 Wave forecasting with deep learning wind prediction

The effects of the new DLM output (the wind velocity prediction) on ocean waves forecasting was examined by using it as a forcing of the WAVEWATCH III v6.07 (WW3) model. WW3 ran with an unstructured grid of the eastern (Levant) area of the Mediterranean Sea, using 36 directions, 36 frequencies in the range $0.04{-}0.427Hz$ and a time step of $dt_{global} = 10min$. The wind forcing source term of Ardhuin et al. (2010) was used, alongside a linear wind interpolation. Six input configurations were tested: ECMWFs FC and REAN, and the four DLM outputs. WW3 ran separately with each forcing for the year 2017. The resulting wave forecast mean field parameters of significant wave height (H_s), mean wave direction (dir) and mean wave period ($T_{m0,-1}$) are shown in Table 2. All DLM outperformed the FC, as expected. Surprisingly, UMag had the best performance for both wave height and period, while UVec results with a better mean direction. UDir was outperformed by the other models and UFrc was almost identical to UVec with slightly worse results. Thus, only UMag and UVec are shown in the following analysis.

A spatial map of H_s time-mean RMSE differences can be seen in Fig. 2. The RMSE difference was taken as $RMSE_{FC} - RMSE_{DLM}$, meaning the new DLM has better performance where positive and vice versa. It is immediately apparent that both DLM outperform the original FC in the eastern part of the basin, especially in the Aegean Sea where the local improvement is $\sim$20%. The FC slightly outperforms the DLM at the western part. The better performance of UMag can be attributed to more accurate results

Table 2. *Model wave mean parameters RMSE*

Property	FC	UMag (%improved)	UVec (%)	UDir (%)	UFrc (%)
$H_s[m]$	0.0765	**0.0676 (11.6%)**	0.0698 (8.7%)	0.0762 (0.4%)	0.0705 (7.8%)
$Dir[deg]$	44.4	42.8 (3.4%)	**42.2 (4.9%)**	43.8 (1.3%)	42.5 (4.3%)
$T_{m0,-1}[sec]$	0.309	**0.283 (8.4%)**	0.286 (7.4%)	0.307 (0.05%)	0.287 (7.1%)

over the western half, as well as better performance along the coastal area. This spatial deviation suggests that applying a mask during the training process or combining the prediction with FC might be beneficial.

A temporal comparison of spatial-mean RMSE of the DLM and FC is given in Fig. 3. This shows that the main improvement of both DLM was during the spring to autumn period, most prominently during the summer months (implying correction of the Etesian wind). The current model can be used as is, or as a seasonal model, alongside a separate seasonal model trained specifically for the winter season or even for stormy conditions. Such models can work as an ensemble to produce better results.

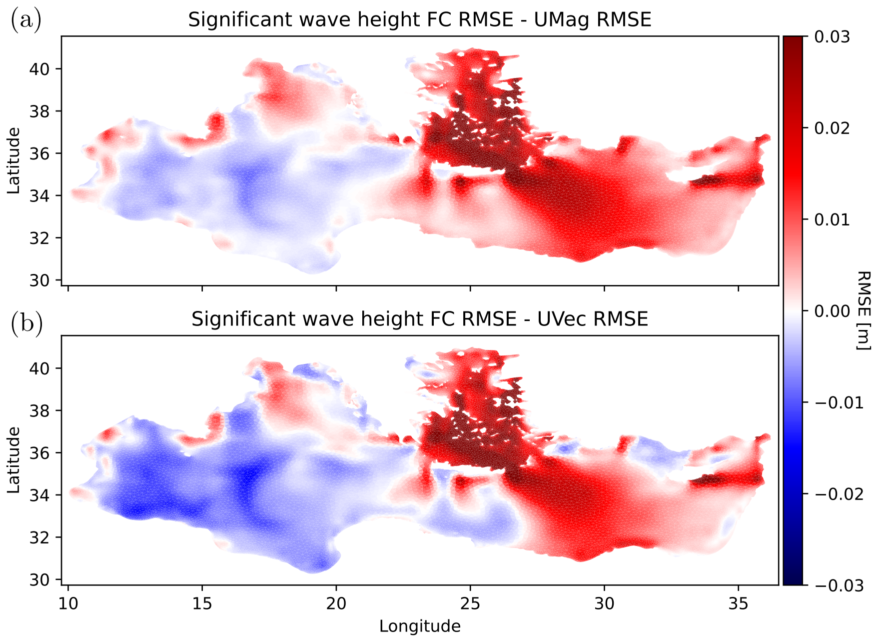


Figure 2. Time-mean RMSE difference map of significant wave height H_s for: (a) FC RMSE - UMag RMSE; (b) FC RMSE - UVec RMSE. FC with larger error in red, DLM in blue.

6 Summary and discussion

In this work a novel hybrid model was presented combining numerical, physics-based models with a deep learning, data-driven model (DLM) to improve wind and waves fore-

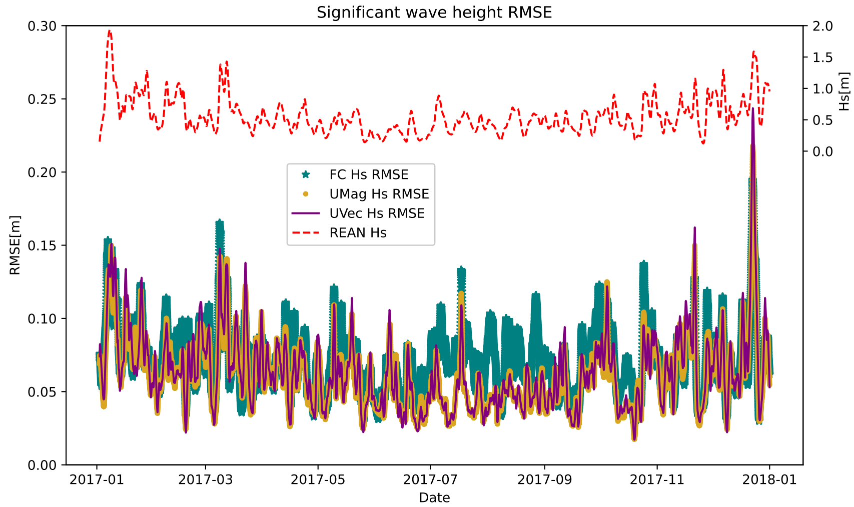


Figure 3. Spatial-mean RMSE of significant wave height H_s $24hrs$ moving average of: FC (thick teal); UMag (medium orange); UVec (thin purple). The right axis is the REAN H_s in dashed red line, for reference.

casting accuracy. The DLM's input was ECMWF's Era5 forecast (FC), which was fitted to the matching reanalysis (REAN) data. This model consisted of convolutional encoder and decoder, with a convolutional gated recurrent unit in between. The DLM's output was used as a forcing for a wave forecasting model (WAVEWATCH III), and the resulting significant wave height, mean wave direction and mean wave period were examined. The new model showed significant improvement in all wind and wave parameters.

The presented DLM was used to improve wind velocity, but could easily be trained to improve any other parameter of the atmospheric model, such as geopotential height or temperature. It could also be trained over different locations, or as a global model. Furthermore, another very interesting usage is training towards seasonal localized models. These could be optimized over specific time periods and locations where weather conditions are hard to predict, and make significant improvement. One such example is shown in this work at the Aegean Sea, where the Etesian wind is dominant during mid-May to mid-September. Even without training specifically for this task, the presented model improves the significant wave height forecast over the Aegean Sea at this period by $\sim$35%.

Another benefit of the new model is very minimal computational cost, which is negligible when compared to either the numerical wind or wave forecasting models. Furthermore, it could easily be implemented, as it does not require any adjustment to any of the currently used operational models, while providing significant improvement in forecasting results.

Acknowledgments

This research was supported by: the ISRAEL SCIENCE FOUNDATION [grant No. 1601/20] and The Gordon Center for Energy Studies. The authors would like to thank Prof. Lior Wolf for his insights and helpful discussion.

Author Contribution

Y.Y. conceived the research, created the deep learning model and performed the experiments. Y.T. supervised the work. Both authors wrote the manuscript

Data Availability

The ERA5 wind datais available from ECMWF and the Copernicus Climate Data Store database at https://cds.climate.copernicus.eu/. The WAVEWATCH III wave model is available at https://github.com/NOAA-EMC/WW3/. The fasi.ai repository used to create the deep learning model is available at https://www.fast.ai/.

References

Arcomano, T., Szunyogh, I., Pathak, J., Wikner, A., Hunt, B. R., & Ott, E. (2020, may). A Machine Learning-Based Global Atmospheric Forecast Model. *Geophysical Research Letters*, *47*(9).

Ardhuin, F., Rogers, E., Babanin, A. V., Filipot, J.-F., Magne, R., Roland, A., ... Collard, F. (2010). Semiempirical Dissipation Source Functions for Ocean Waves. Part I: Definition, Calibration, and Validation. *Journal of Physical Oceanography*, *40*(9), 1917–1941.

Ballas, N., Yao, L., Pal, C., & Courville, A. (2015, nov). Delving Deeper into Convolutional Networks for Learning Video Representations. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*.

Bidlot, J. R., Holmes, D. J., Wittmann, P. A., Lalbeharry, R., & Chen, H. S. (2002, apr). Intercomparison of the performance of operational ocean wave forecasting systems with buoy data. *Weather and Forecasting*, *17*(2), 287–310.

Booij, N., Ris, R. C., & Holthuijsen, L. H. (1999, apr). A third-generation wave model for coastal regions. 1. Model description and validation. *J. Geophys. Res.*, *104*(C4).

Brunton, S. L., Noack, B. R., & Koumoutsakos, P. (2020, jan). Machine Learning for Fluid Mechanics. *Annual Review of Fluid Mechanics*, *52*(1), 477–508.

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014, dec). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling.

Hasselmann, K., Hasselmann, S., Bauer, E., Janssen, P. A., Komen, G. J., Bertotti, L., ... Ewing, J. A. (1988, dec). *The WAM model - a third generation ocean wave prediction model.* (Vol. 18; Tech. Rep. Nos. 12 , Dec. 1988).

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., ... Thépaut, J. N. (2020, jul). The ERA5 global reanalysis. *Quarterly Journal of the Royal Meteorological Society*, *146*(730), 1999–2049.

Howard, J., & Gugger, S. (2020, feb). fastai: A Layered API for Deep Learning. *Information (Switzerland)*, *11*(2).

Janssen, P., & Janssen, P. A. E. M. (2004). *The interaction of ocean waves and wind.* Cambridge University Press.

Kingma, D. P., & Ba, J. L. (2015, dec). Adam: A method for stochastic optimization. In *3rd international conference on learning representations, iclr 2015 - conference track proceedings.* International Conference on Learning Representations, ICLR.

Littwin, E., & Wolf, L. (2016, nov). The Loss Surface of Residual Networks: Ensembles and the Role of Batch Normalization.

Rasp, S., Dueben, P. D., Scher, S., Weyn, J. A., Mouatadid, S., & Thuerey, N. (2020, feb). Weatherbench: A benchmark dataset for data-driven weather forecasting.

Rasp, S., & Thuerey, N. (2020, aug). Data-driven medium-range weather prediction with a Resnet pretrained on climate simulations: A new model for WeatherBench.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, *566*(7743), 195–204.

Roitenberg, A., & Wolf, L. (2019). Forecasting traffic with a convolutional gru decoder conditioned on adapted historical data. *ICML 2019 Time Series Workshop*.

Scher, S., & Messori, G. (2019, jul). Weather and climate forecasting with neural networks: Using general circulation models (GCMs) with different complexity as a study ground. *Geoscientific Model Development*, *12*(7), 2797–2809.

Smith, L. N. (2018, mar). A disciplined approach to neural network hyper-parameters: Part 1 – learning rate, batch size, momentum, and weight decay.

Tolman, H. L. (1991, jun). A Third-Generation Model for Wind Waves on Slowly Varying, Unsteady, and Inhomogeneous Depths and Currents. *J. of Phys. Oceanogr.*, *21*(6), 782–797.

WAMDI Group, T. (1988). The WAM model - A third generation ocean wave prediction model. *Journal of Physical Oceanography*, *18*(12), 1775–1810.

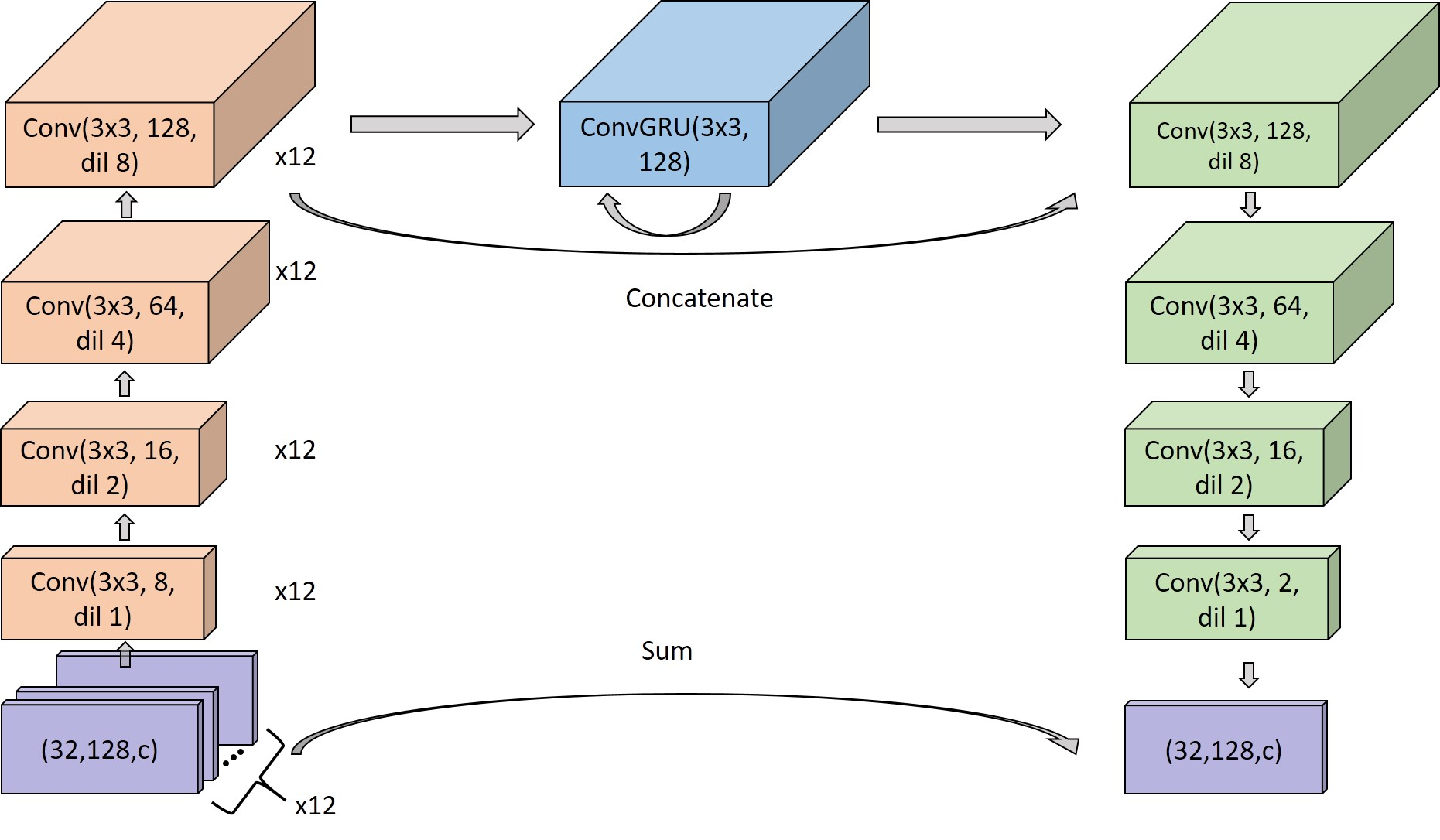
Wang, S., Cao, J., & Yu, P. S. (2019, jun). *Deep learning for spatio-temporal data mining: a survey.* arXiv.

Weyn, J. A., Durran, D. R., & Caruana, R. (2019, aug). Can Machines Learn to Predict Weather? Using Deep Learning to Predict Gridded 500-hPa Geopotential Height From Historical Weather Data. *Journal of Advances in Modeling Earth Systems*, *11*(8), 2680–2693.

Weyn, J. A., Durran, D. R., & Caruana, R. (2020, mar). Improving data-driven global weather prediction using deep convolutional neural networks on a cubed sphere. *Journal of Advances in Modeling Earth Systems*, *12*(9).

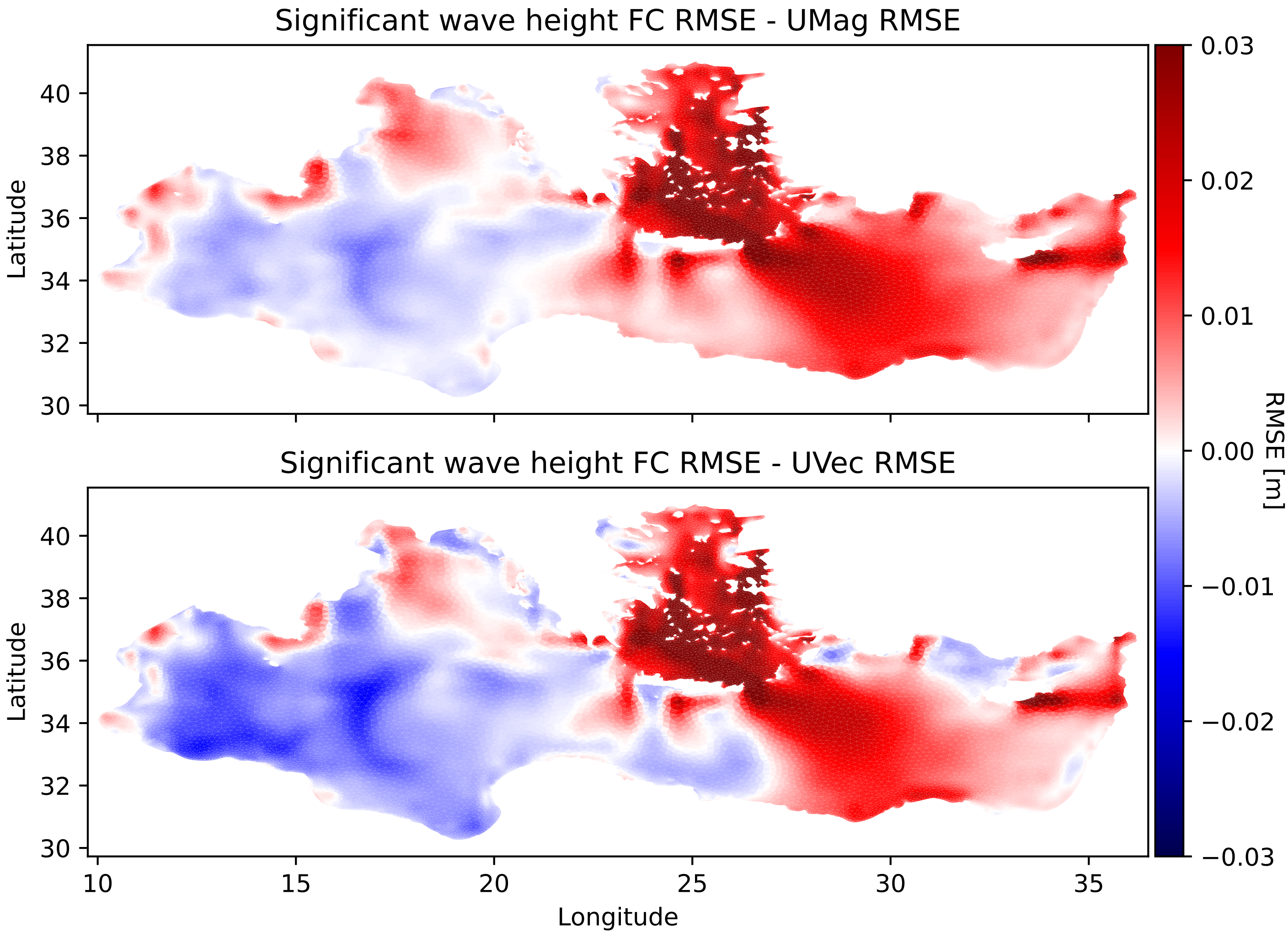
Yu, F., & Koltun, V. (2015, nov). Multi-Scale Context Aggregation by Dilated Convolutions. *4th International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings*.

Figure1.



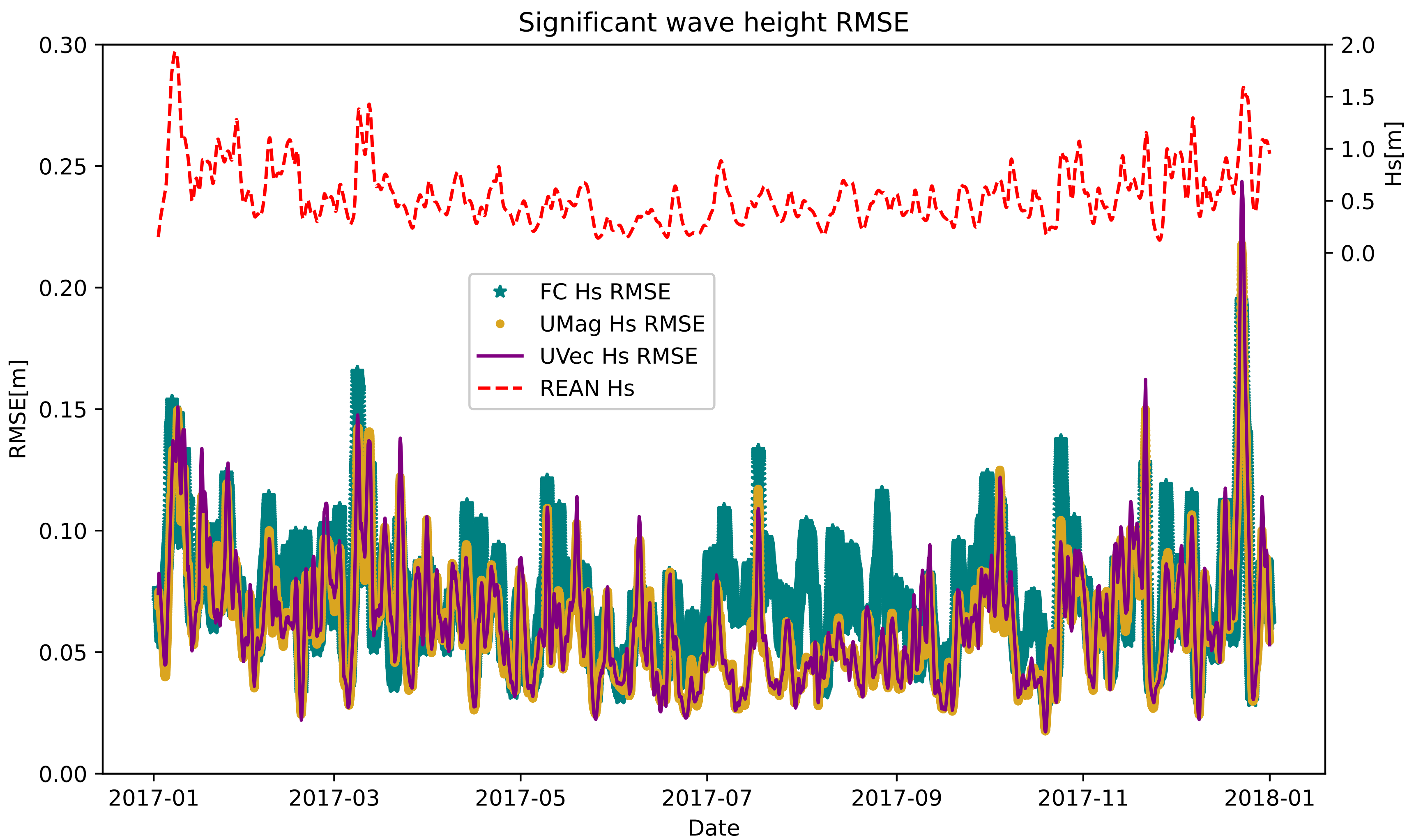
Conv(3x3, 128, dil 8)
x12
ConvGRU(3x3, 128)
Conv(3x3, 128, dil 8)
Conv(3x3, 64, dil 4)
x12
Concatenate
Conv(3x3, 64, dil 4)
Conv(3x3, 16, dil 2)
x12
Conv(3x3, 16, dil 2)
Conv(3x3, 8, dil 1)
x12
Conv(3x3, 2, dil 1)
Sum
(32,128,c)
x12
(32,128,c)

Figure2.



Significant wave height FC RMSE - UMag RMSE
Significant wave height FC RMSE - UVec RMSE
Latitude
Longitude
RMSE [m]
0.03
0.02
0.01
0.00
−0.01
−0.02
−0.03
40
38
36
34
32
30
10
15
20
25
30
35

Figure3.


Significant wave height RMSE
0.30
0.25
0.20
0.15
0.10
0.05
0.00
RMSE[m]
2.0
1.5
1.0
0.5
0.0
Hs[m]
FC Hs RMSE
UMag Hs RMSE
UVec Hs RMSE
REAN Hs
2017-01
2017-03
2017-05
2017-07
2017-09
2017-11
2018-01
Date