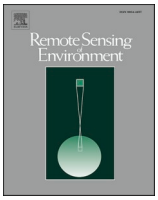
Remote Sensing of Environment 269 (2022) 112801

Contents lists available at ScienceDirect

Remote Sensing of Environment

journal homepage: www.elsevier.com/locate/rse



GNSS reflectometry global ocean wind speed using deep learning: Development and assessment of CyGNSSnet

Milad Asgarimehr [a,*], Caroline Arnold [b], Tobias Weigel [b], Chris Ruf [c], Jens Wickert [a,d]

[a] German Research Centre for Geosciences GFZ, Potsdam, Germany
[b] Deutsches Klimarechenzentrum DKRZ, Hamburg, Germany
[c] Climate and Space Department, University of Michigan, Ann Arbor, MI, USA
[d] Institute of Geodesy and Geoinformation Science, Technische Universität Berlin, Germany

ARTICLE INFO

Edited by Dr. Menghua Wang

Keywords:
Wind speed
Ocean
GNSS reflectometry
CyGNSS
Deep learning
CNN
Fully connected layers

ABSTRACT

GNSS Reflectometry (GNSS-R) is a novel remote sensing technique for the monitoring of geophysical parameters using reflected GNSS signals from the Earth's surface. Ocean wind speed monitoring is the main objective of the recently launched Cyclone GNSS (CyGNSS), a GNSS-R constellation of eight microsatellites, launched in late 2016. In this study, the capability of deep learning, especially, for an operational wind speed data derivation from the measured Delay-Doppler Maps (DDMs) is characterized. CyGNSSnet is based on convolutional layers for the feature extraction from bistatic radar cross section (BRCS) DDMs, along with fully connected layers for processing ancillary technical and higher-level input parameters. The best architecture is determined on a validation set and is evaluated over a completely blind dataset from a different time span than that of the training data to validate the generality of the model for operational usage. After a data quality control, CyGNSSnet results in an RMSE of 1.36 m/s leading to a significant improvement by 28% in comparison to the officially operational retrieval algorithm. The RMSE is the lowest among those seen in the literature for any conventional or machine learning-based algorithm. The benefits of the convolutional layers, the advantages and weaknesses of the model are discussed. CyGNSSnet offers efficient processing of GNSS-R measurements for high-quality global ocean winds.

1. Introduction

Remote sensing is a key component for monitoring and modeling Earth systems. The increasing remote sensing datasets coupled to innovative deep learning techniques for geo-information extraction have led to understanding and quantifying the Earth's complexity to an unprecedented extent (Zhu et al., 2017; Ma et al., 2019). However, deep learning has been mostly limited to the processing of optical remote sensing data.

GNSS Reflectometry (GNSS-R) refers to the exploitation of GNSS signals, after reflection off the Earth's surface, as a bistatic radar technique for the remote sensing of a variety of geophysical properties (Zavorotny et al., 2014). Low-mass receivers onboard small satellites track multiple reflected GNSS signals simultaneously. Substantially lower development costs, and higher sampling frequency and spatial coverage are the main advantages of GNSS-R missions compared to other remote sensing techniques. The data volume is expected to increase in the near future with the emergence of GNSS-R CubeSats (Munoz-Martin et al., 2020). With the availability of large data sets, deep learning becomes a viable and effective method in GNSS-R. In this study, we show that deep learning can be used to retrieve high-quality global ocean wind speed. We demonstrate the general applicability and the potential benefits of employing machine learning in GNSS-R ocean monitoring. As we show in the subsequent sections, deep learning can outperform conventional methods that determine wind speeds from GNSS-R signals, and we encourage to embrace machine learning techniques as the key processing approach to GNSS-R.

1.1. GNSS-R basics and challenge to address

The capability of the spaceborne GNSS-R in monitoring ocean state and surface wind was well demonstrated using the measurements of the UK TechDemoSat-1 (TDS-1), launched in 2014 (Foti et al., 2015; Asgarimehr et al., 2018a). The first spaceborne mission fully dedicated

* Corresponding author.
E-mail address: milad.asgarimehr@gfz-potsdam.de (M. Asgarimehr).

https://doi.org/10.1016/j.rse.2021.112801
Received 11 June 2021; Received in revised form 18 October 2021; Accepted 10 November 2021
Available online 23 November 2021

to GNSS-R, Cyclone GNSS (CyGNSS), was launched in late 2016. The CyGNSS is a constellation of eight microsatellites with the primary science objective of ocean wind speed monitoring especially in hurricanes and tropical cyclones (Ruf et al., 2018a).

The most common spaceborne GNSS-R observable is the Delay Doppler Map (DDM) which refers to a map of the cross-correlation power at a range of signal delay and Doppler frequency shift. The cross-correlation is carried out between the received reflected signal and a replica of it locally generated at the receiver. Fig. 1 shows exemplary Bistatic Radar Cross Section (BRCS) DDMs at different wind speeds (according to ERA5 estimates) and how the BRCS decreases with the increase in wind speed. In the conventional approach, the conversion of the measurements to the wind data products relies on the extraction of a quantity from the DDMs, which is necessarily controlled by surface roughness and consequently wind speed. After correcting non-wind-derived effects such as receiver transmitter antenna gains and signal power variations due to the ranges between the transmitter and receiver satellites and the specular point, this quantity is then fitted to surface wind speed using a match-up dataset, either from other instruments or reanalysis model estimates (Clarizia et al., 2014). Conventionally, the Normalized BRCS (NBRCS) of a 3 delay × 5 Doppler bin box that includes the specular point bin, or/and the Leading Edge Slope (LES) of the same box, are obtained from CyGNSS DDMs. The derived NBRCS/LES is converted to wind speed after a parametric fitting using e.g. a Minimum Variance Estimator (MVE) (Clarizia et al., 2014).

The LES and NBRCS, being extracted from the small box containing the specular point, are in principle a feature extracted from DDMs based on theoretical knowledge and are not representative for the entire DDM. Besides, the NBRCS extraction from DDMs requires Bistatic Radar Equations (BREs) describing the DDMs as a function of surface state and technical properties. The BREs are limited by simplifying but necessary assumptions on the physics of the signal scattering. The BRE describing the signal scattering is based on the Geometric Optics limit of the Kirchhoff Approximation (KA-GO), which limits the validity of the BRE to the regime of strong diffuse scattering (Zavorotny and Voronovich, 2000), i.e. to oceans induced with high enough wind speeds.

In addition, the theoretical knowledge is still subjected to significant improvement due to the novelty of the GNSS-R technique and as the models are being validated for field conditions. A deep learning approach, approximating the unknown predictor in a data-driven way, is potentially able to overcome the limitations imposed by imperfect theoretical models in novel remote sensing domains, and beyond that, enhance the physical knowledge empirically. In addition, network architectures based on convolutional layers offer automatic feature extraction from the entire DDMs instead of the small box, which is the main concept employed in CyGNSSnet.

1.2. Related studies

A few studies investigated machine learning techniques for the derivation of GNSS-R ocean winds. A Feed Forward Neural Network (FFNN) was implemented for the wind speed estimation using measurements of a ground-based receiver (Asgarimehr et al., 2020). In that study, an FFNN with one hidden layer was trained so that it predicted wind speed from the waveforms derived using the reflected signals of BeiDou Geostationary Earth Orbit (GEO) satellite number 4.

Training Neural Networks (NNs) using spaceborne measurements is a more challenging task due to the varying transmitters, reflection geometries, and environmental parameters compared to those using ground-based observations. A similar FFNN architecture was trained using TechDemoSat-1 measurements (Asgarimehr et al., 2020). The model took the BRCS at the specular point, along with additional technical parameters to capture the associated systematic patterns. The NN-derived wind speeds led to significantly better estimates compared to a conventional retrieval approach. This previous study discussed that better estimations are achieved as the NN learns and corrects systematic technical effects such as those associated with the varying Equivalent Isotropically Radiated Power (EIRP) of GPS satellites. Such a fusion of raw input features using a multi-hidden layer NN also resulted in promising results using CyGNSS measurements (Reynolds et al., 2020). The advantage of using NNs for GNSS-R Wind Speed Retrievals is further demonstrated in (Liu et al., 2019) processing the entire DDMs. More

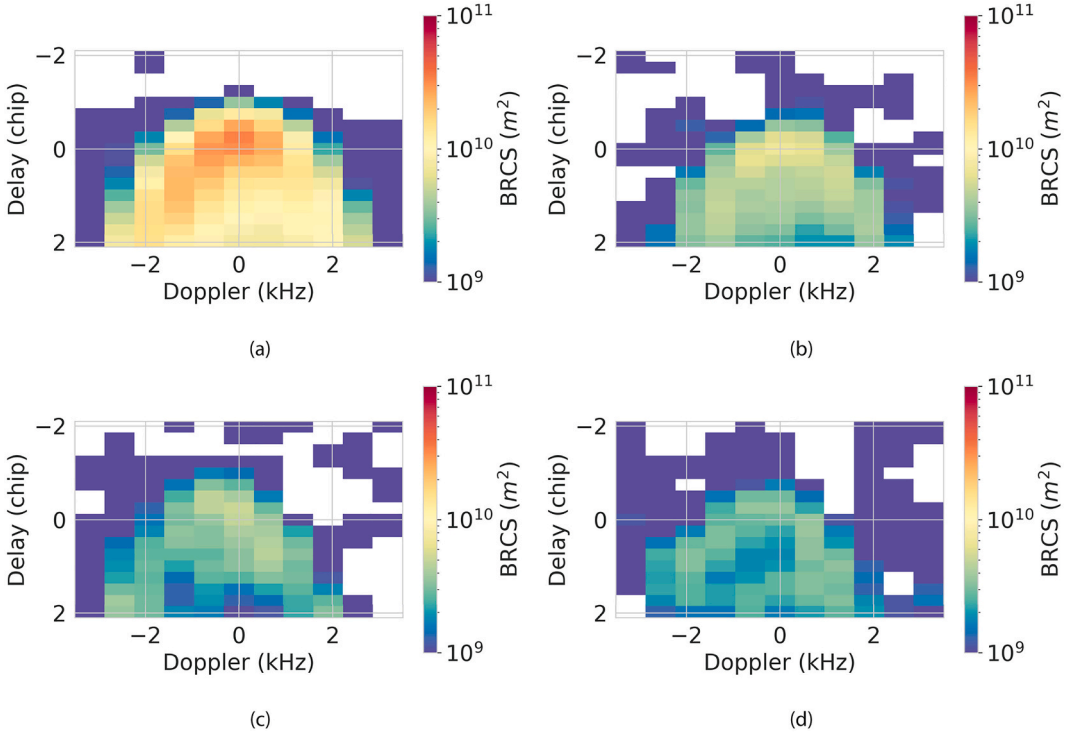


Fig. 1. BRCS DDMs at incidence angles below 13° and different wind speeds: below 3 m/s (a), between 6–9 m/s (b), between 12–15 m/s (c), and above 15 m/s (d).

advanced architectures including convolutional layers to extract features from DDMs was investigated using a one-month TDS-1 dataset for the training and validation (Chu et al., 2020), however, the analysis of the generalization capability of the network to temporally blind test data remained for future studies.

The deep learning retrieval models can calibrate themselves to some extent, empirically based on the seen examples. The main calibrations are most commonly carried out manually by instrument engineers. Machine learning tools can still assist us to control data quality and better understand measurements behavior, such as the usage of Neural Networks for automated outlier detection of CyGNSS retrieved winds (Balasubramaniam and Ruf, 2020b).

In this interdisciplinary study, we propose CyGNSSnet, a deep-learning model processing satellite measurements to retrieve wind speed data products. The network, inspired by NN architectures for the classification of image data, extracts the features from BRCS DDMs and the DDM well-known quantities, i.e. LES and NBRCS (Clarizia and Ruf, 2016). It is trained by supervised learning, using ERA5 reanalysis wind speed data (Hersbach et al., 2020) as labels. Section 2 describes the datasets used. Section 3 gives the details on the model architecture and its implementation and training procedure. The evaluation results are presented in Section 4 and discussed in Section 5. Finally, Section 6 gives the concluding remarks and suggestions for future work.

2. Dataset and preprocessing

This work is based on version 2.1 of the CyGNSS data set (CYGNSS, 2017) covering the full year 2018 and the first 52 days of 2019. The first 215 days are used for training, and the following 75 days are used for validation during training. The last 127 days are set aside as a test set to evaluate the performance of the model on unseen data.

The main measurement used in this study is the BRCS DDM which is a 17 × 11 array of DDM bin effective scattering area (m^2). Physically, these measurements represent the true surface scattering area that contributes power to each DDM bin (the original power DDM). The mapping from scattering area to DDM bin is not unique - some bins include contributions from two different scattering areas. Note that this quantity is different from the power DDM, for which the term DDM is commonly used. In addition to BRCS DDM, each sample contains up to 16 ancillary parameters related to the DDM, the GPS satellite, and the measurement geometry. The NBRCS and the LES were selected based on their theoretical relevance for wind speed estimation. Additional technical parameters were selected from the available metadata, based on the theoretical possible correlation with the predicted wind speed. We train five models separately for each parameter and evaluate their performance on the validation set. CNN-VAR-10 encompasses those parameters that increased performance in at least 3 out of 5 models. CNN-VAR-16 encompasses parameters that increased performance at least in one model. The parameters found not to be helpful, among those initially selected, namely sc_alt, sc_velocity, sc_yaw, and $\log_{10}$(tx_to_sp_range), are henceforth not considered. All input parameters are summarized in Table 1.

For supervised learning, the samples are labeled with nearest neighbor wind speed estimates from ERA5 reanalysis data (Hersbach et al., 2020). Due to the limited spatiotemporal resolution of ERA5, samples recorded within a time interval of 6 s along a CyGNSS observational track will be labeled with the same wind speed. With an integration time of 1000 ms per DDM, this amounts to up to 6 DDMs per ERA5 spatiotemporal bin. We form the training set by randomly selecting one sample per ERA5 spatiotemporal bin. Thus, we achieve marginally higher accuracy on the validation set, compared to including all samples. We should note that the entire 11×17 DDM responds to scattering from areas much further away from the specular point than the ≈36 km motion of the specular point over 6 s. So, this approach implicitly assumes that the wind speed is uniform over the entire spatial domain covered by the 11×17 DDM.

To exclude low-quality measurements, we used the samples which meet the following conditions:

- The uncertainty of the BRCS (ddm_brcs_uncert) is below 1.
- The absolute value of the spacecraft roll is between 1° and 30°, the pitch is between 1° and 10°, or the yaw is between 1° and 5°, indicated by the quality_flags.
- The nano star tracker attitude status is "OK", i.e. nst_att_status = 0.
- The receive antenna gain in the direction of the specular point (sp_rx_gain) is larger than 0 dBi.
- The range corrected gain (RCG) figure of merit (FOM) for the DDM (prn_fig_of_merit) is larger than 0.
- The LES (ddm_les) is larger than 0.
- The Zenith (direct) signal to noise ratio (direct_signal_snr) is larger than 0 dB.

We also remove samples that lie outside the 95% confidence interval of the wind-speed-dependent value range of the NBRCS (ddm_nbrcs), see Fig. 2. The top and bottom border values of the confidence interval were determined empirically on the training set and are $\mathrm{nbrcs}(v) = 27.53\,e^{-0.16v} + 7.99, \mathrm{nbrcs}(v) = 285.0\,e^{-0.40v} + 18.96,$ respectively, where v refers to the wind speed label. These outliers can arise due to

Table 1
CyGNSSnet architectures explored in this study. All architectures include the BRCS DDM (brcs). Architectures CNN-VAR-2, CNN-VAR-10, CNN-VAR-16 include the specified number of ancillary parameters. For reference, FC-VAR-2 and FC-VAR-10 are included. They process the same ancillary variables as CNN-VAR-2 and CNN-VAR-10, but not the BRCS DDM. For the variables descriptions, see CyGNSS L1 V2.1 users's guide and data dictionary, https://podaac-tools.jpl.nasa.gov/drive/files/allData/cygnss/L1/docs/148-0346-6_L1_v2.1_netCDF_Data_Dictionary.xlsx.

Architecture	Maps	Map-related	Receiver-related	Geometry-related
CNN	brcs	—	—	—
CNN-VAR-2	brcs	ddm_nbrcs, ddm_les	—	—
CNN-VAR-10	brcs	ddm_nbrcs, ddm_les, $\log_{10}$(les_scatter_area), $\log_{10}$(nbrcs_scatter_area), ddm_snr	gps_eirp, $\log_{10}$(rx_to_sp_range)	sp_inc_angle, sp_alt, sp_theta_orbit
CNN-VAR-16	brcs	ddm_nbrcs, ddm_les, $\log_{10}$(les_scatter_area), $\log_{10}$(nbrcs_scatter_area), ddm_brcs_uncert, ddm_snr	gps_eirp, $\log_{10}$(rx_to_sp_range),	sp_inc_angle, sp_alt, sp_theta_orbit, sc_pitch sc_roll, sp_az_orbit, zenith_sun_angle_az, zenith_sun_angle_decl
FC-VAR-2	—	ddm_nbrcs, ddm_les	—	—
FC-VAR-10	—	ddm_nbrcs, ddm_les, $\log_{10}$(les_scatter_area), $\log_{10}$(nbrcs_scatter_area), ddm_snr	gps_eirp, $\log_{10}$(rx_to_sp_range)	sp_inc_angle, sp_alt, sp_theta_orbit

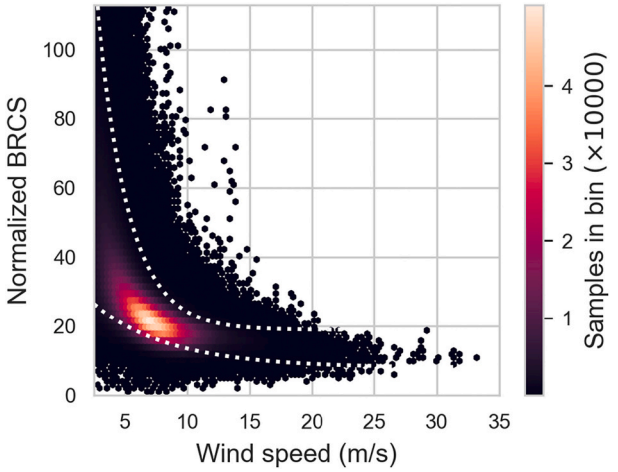


Fig. 2. Sample counts across bins of the Normalized Bistatic Radar Cross Section (NBRCS) and ERA5 wind speed. The dotted lines indicate the cut-off for the 95% confidence interval.

fraudulent behavior of GNSS-R instruments or a flaw in ERA5 reanalysis. A CyGNSS measurement or ERA5 estimate may also have been contaminated by unexpected sources which normally do not affect the major population.

The DDM observables, in particular the NBRCS in the regime of weak diffuse scattering, are insensitive to winds lower than 2.5 m/s. There is no Bragg resonant scattering with the quasi-specular forward scattering, however, the scattering mechanism changes from quasi-specular scattering to that more similar to a high-order Bragg scattering at low wind speeds. This causes an ambiguity confirmed in simulations (Asgarimehr et al., 2018b) and empirically shown using TDS-1 data (Asgarimehr et al., 2020). So, wind speeds below the cut-off at 2.5 m/s cannot be predicted by a deep learning algorithm and are removed in this study.

After filtering and data reduction, 7.2×10^6 samples remain in the training set, 4.7×10^6 samples in the validation set, and 8.8×10^6 samples in the test set, see Fig. 3. Note that the data is clustered in time due to the quality control.

Spaceborne GNSS-R missions sweep the ocean in observational tracks and there are strong correlations and similarities in adjacent DDMs in these tracks. These correlations may skew performance analysis on test data, if the test data would be randomly collected during the same time span as the training data. Typically, processing algorithms for operational purposes are based on an existing data set, and operate on the upcoming measurements. We follow that approach by collecting training, validation, and test data sets in different time spans. This enables us to judge the ability of CyGNSSnet to generalize to unseen data.

We compare the performance of the derived model with that of a conventional model as the baseline. To this end, we use the wind speeds derived based on a Minimum Variance Estimator (MVE), determined for fully developed seas (Clarizia et al., 2014; Ruf and Balasubramaniam, 2018). The MVE determines wind speed in the spatial and temporal bins using a minimum variance combination of wind estimates which are in turn derived from two DMM observables, the NBRCS of the ocean surface and the LES of the radar observables. The wind speeds are available in the Level 2 CyGNSS data product (CYGNSS, 2018), a detailed assessment of which is given in (Ruf et al., 2018b).

3. Model determination

In this section, we describe the general architecture of CyGNSSnet, as well as the scheme to optimize the network hyperparameters.

3.1. CyGNSSnet architecture

CyGNSSnet (Fig. 4) has two input lines, one for DDMs, and one for ancillary parameters. In the first input line, features are extracted from the BRCS DDMs by a block of N_l convolutional layers with fixed kernel size 3×3, based on the VGGNet architecture (Simonyan and Zisserman, 2015). The first layer has N_f filters, and every N_d layers, their number is doubled. Batch normalization is applied after each convolutional layer to reduce the internal covariate shift and allow for larger learning rates (Ioffe and Szegedy, 2015). Maximum 2×2 pooling is applied every N_p layers. A second input line processes ancillary parameters through a fully-connected layer. Both input lines are concatenated and processed in two additional dense layers. Dropout layers are added following each dense layer to avoid overfitting (Srivastava et al., 2014).

We employ the rectified linear unit (ReLU) activation function, $f(x) = \max(0, x)$, throughout. ReLU avoids the problem of vanishing gradients encountered in tanh and sigmoid activation functions (Wang and Xi, 1997). Training is conducted with the Adam optimizer on randomly shuffled minibatches. The loss function is the mean squared error of predicted $(\hat{v}_i)$ and true wind speeds (v_i) across N samples, obtained as

$$l(\hat{v}, v) = \frac{1}{N} \sum_{i=1}^{N} (\hat{v}_i - v_i)^2.$$

We train different CyGNSSnet architectures that vary with respect to the included ancillary parameters. For a summary of architectures and ancillary parameters, see Table 1. CNN-VAR-2, CNN-VAR-10, and CNN-VAR-16 use the CyGNSSnet architecture as described above. For

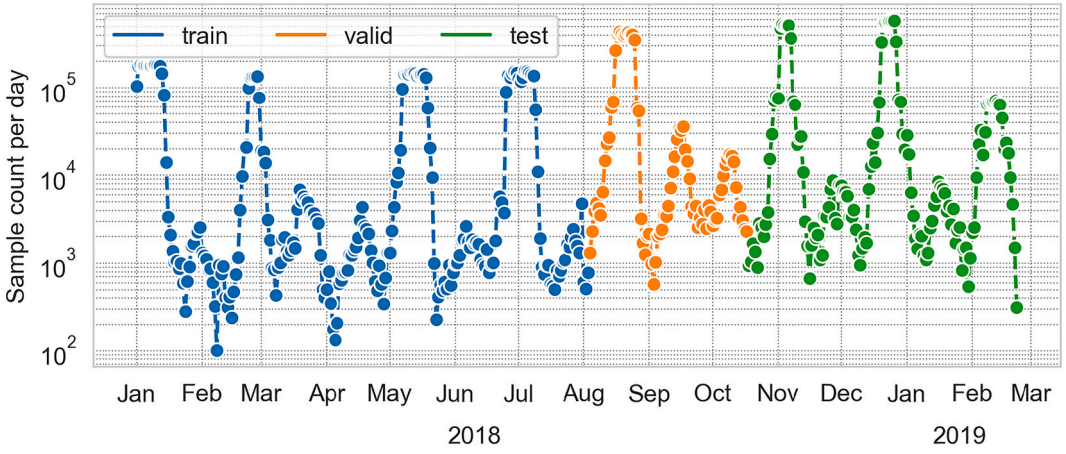


Fig. 3. Sample count per day in the preprocessed training, validation, and test set after applying quality control and data reduction.

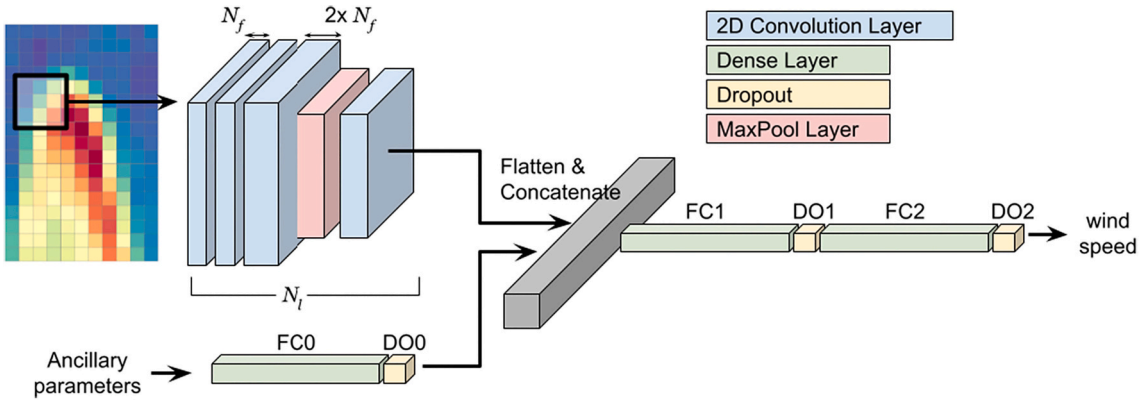


Fig. 4. Architecture of CyGNSSnet. The network combines two input lines: BRCS DDMs are processed through a convolutional neural network (CNN) with a total of N_l 2D convolutional layers with fixed kernel size 3×3. The first layer has N_f filters, and every N_d layers, their number is doubled. Every N_p layers, 2×2 maximum pooling is applied. For illustration purposes, $N_d = 2$ and $N_p = 3$ are chosen. Ancillary parameters (cf. Table 1) are processed through a dense layer FC0. Both input lines are concatenated and processed through two dense layers FC1, FC2. Finally, wind speed is predicted. Dropout layers are added to avoid overfitting (Srivastava et al., 2014). ReLU is used as the activation function in all layers. For the full network hyperparameter search space, see Table 2.

comparison, we provide results for the CNN architecture that does not have the second input line, and for the FC-VAR-2 and FC-VAR-10 architectures that process only ancillary parameters and have no convolutional layers.

On the validation set, the best performance was reached with the CNN-VAR-10 architecture.

3.2. Hyperparameter tuning

The network and training parameters are optimized using the Neural Network Intelligence (NNI) package (Microsoft Corporation, 2020). In Table 2, we summarize the parameters and their respective search spaces. In total, there are up to 12 hyperparameters that need to be optimized. We use the NNI implementation of the Tree-Parzen Estimator (TPE) (Bergstra et al., 2011), which is well suited for high-dimensional parameter spaces and saves on computational resources.

3.3. Training and validation

CyGNSSnet is implemented in PyTorch (Paszke et al., 2019). The networks can be trained on a single NVIDIA Tesla K80 GPU within less than 12 hours. The preprocessed data is normalized feature-wise to zero mean and unit variance. The validation set is employed to avoid overfitting; we employ the early-stopping condition with six epochs patience.

The hyperparameters are optimized separately for each architecture. The best three models are combined to form an ensemble. Predictions on the validation set and the test set are averaged across the ensemble, which typically leads to a slight performance increase of a few percent. The hyperparameters for all ensemble members of all architectures are reported in Table S1 in the supplement.

Table 2
Hyperparameter search space explored using NNI (Microsoft Corporation, 2020). For the three dense layers and dropout layers, the hyperparameters are optimized separately. Thus, the total number of tunable hyperparameters is 12.

Parameter		Search space
Learning rate		$5 \times 10^{-5} \ldots 1 \times 10^{-3}$
Batch size		32 … 2048
Number of convolutional layers	N_l	1 … 8
Filters in 1st convolutional layer	N_f	8 … 64
Filters doubled after layer	N_d	2 … 8
Pooling after layer	N_p	1 … 8
Units in dense layers		4 … 256
Dropout after dense layers		0.0 … 0.3

4. Evaluation

The evaluation is conducted on the hold-out test set covering the last 75 days in 2018 and the first 52 days in 2019. This test set was not seen during training time and is temporally separated from the training and validation set. It is therefore well-suited for evaluating the ability of CyGNSSnet to generalize well on a blind dataset. The performance of the architectures is also compared to the conventional and currently, operational retrieval algorithm using the MVE applied to retrievals over Fully Developed Seas estimating wind speed from the NBRCS and LES (Clarizia and Ruf, 2016; Ruf and Balasubramaniam, 2018).

4.1. Performance of CyGNSSnet

Table 3 shows the root mean square error (RMSE),

$$\mathrm{RMSE}(v, \hat{v}) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\hat{v}_i - v_i)^2}, \tag{1}$$

for all six CyGNSSnet architectures as well as the MVE (conventional model). Overall, the CNN-VAR-10 architecture yields the lowest RMSE (1.36 m/s), closely followed by the CNN-VAR-2 and the FC-VAR-10 architecture. In the regime of strong winds ($v > 12\,\mathrm{m/s}$), the RMSE degrades for all architectures and the MVE. Here, adding ancillary parameters as done in the FC-VAR-10, CNN-VAR-10, and CNN-VAR-16 architecture decreases RMSE. The CNN that processes only DDMs performs poorly compared to the other architectures. This indicates that

Table 3
RMSE obtained on the test set for different architectures and ERA5 wind speeds v. See Table 1 for a description of the architectures. The RMSEs reported are based on the same population of samples and the same selective filter, shown in Fig. 2, is applied for all the methods.

Architecture	All samples RMSE (m/s)	$v \leq 12\,\mathrm{m/s}$ RMSE (m/s)	$12\,\mathrm{m/s} < v \leq 16\,\mathrm{m/s}$ RMSE (m/s)	$v > 16\,\mathrm{m/s}$ RMSE (m/s)
CNN	1.71	1.59	3.83	7.24
CNN-VAR-2	1.37	1.31	2.47	4.97
CNN-VAR-10	1.36	1.31	2.38	4.99
CNN-VAR-16	1.47	1.43	2.26	4.57
FC-VAR-2	1.45	1.37	3.00	5.83
FC-VAR-10	1.37	1.31	2.44	5.13
MVE	1.90	1.88	2.29	3.39

adding the NBRCS and the LES is required to achieve satisfying results, even at average wind speeds. For comparison, we include the FC-VAR-2 architecture. Taken across all samples, the RMSE is comparable to the CyGNSSnet architectures incorporating additional parameters. Only at high wind speeds, this fully-connected architecture performs considerably worse than the architectures that include DDMs and convolutional layers for feature extraction. To some extent, this can be remedied by including more parameters in FC-VAR-10. In terms of general RMSE, all architectures outperform the MVE wind speed, however, the MVE still performs better, in terms of RMSE for winds higher than 16 m/s.

Fig. 5 shows the log-scale density plot of the true wind speed labels and the predicted wind speeds for the CNN-VAR-10 architecture along with those of the MVE, as the baseline for performance comparison. For the other architectures, the plot is found in Fig. S1 of the supplement. The CNN-VAR-10 winds are considerably less spread around the 1:1 line than those from the MVE. A slight overestimation in both figures, being centered marginally above the 1:1 line, is observed. Besides, underestimations at high winds are shown by both models.

Fig. 6(a) shows the RMSE for different values of the ERA5 wind speed. For the k-th bin with borders $v_{k,\min}, v_{k,\max}$, the RMSE is calculated as

$$\mathrm{RMSE}(v,\hat{v};k)=\sqrt{\frac{1}{N_k}\sum_{i=1}^{N_k}(\hat{v}_i-v_i)^2},\quad v_{k,\min}\le v_i<v_{k,\max}, \tag{2}$$

where N_k is the number of samples in each bin. Fig. 6(b) shows the bias calculated for the same ERA5 wind speed bins,

$$\mathrm{BIAS}(v,\hat{v};k)=\frac{1}{N_k}\sum_{i=1}^{N_k}(\hat{v}_i-v_i),\quad v_{k,\min}\le v_i<v_{k,\max}. \tag{3}$$

All architectures slightly overestimate low wind speeds. At the especially challenging high wind speeds, it can be seen that the CyGNSSnet architectures CNN-VAR-2, CNN-VAR-10, FC-VAR-10, and CNN-VAR-16 perform better than the other architectures. Fig. 7 shows the number of samples that are available in the training set for each wind speed bin. Only a small fraction of samples are labeled with wind speeds exceeding 16 m/s.

4.2. Generalization in time

For a potential operational application, it is important to gauge whether the deep learning algorithm generalizes well to time periods that are separated from the training data. Since the test data is clustered in time (cf. Fig. 3) and the RMSE is highly dependent on wind speed (cf. Fig. 6), we evaluate the RMSE on the test set for three different time intervals spanning 42 days each. The intervals contain 3.5×10^6, 4.6×10^6, and 0.8×10^6 samples, respectively, and show comparable wind speed distribution. The results, obtained with the CNN-VAR-10 architecture, are summarized in Fig. 8. CyGNSSnet predictions are robust across these periods. The slightly higher RMSE in the first period is attributed to very high wind speeds beyond 30 m/s, that are associated with high RMSE and only present in this phase.

4.3. Global performance

We evaluate the global performance of the CyGNSSnet architecture CNN-VAR-10 in Fig. 9. The CyGNSSnet generally performs better in terms of bias and RMSE in comparison to the MVE. There are similar spatial patterns in RMSE and bias projections by both methods discussed in Section 5.

5. Discussion

5.1. Model architecture

Previous studies have implemented fully connected layers for wind predictions (Asgarimehr et al., 2020; Reynolds et al., 2020; Liu et al., 2019). Inspired by image processing techniques, we investigated image-like processing of GNSS-R which offers the feature extraction from the BRCS DDMs. The comparisons conducted here show that the GNSS-R wind models can benefit from convolutional layers. This is shown by the better performance of the architectures combining convolutional layers than those with only fully connected layers because they take advantage of the patterns in the DDMs. The features extracted through convolutional layers might not be represented by the NBRCS and LES fed to the fully connected layers. Comparing the performance of FC-VAR-2 and –10 to that of CNN-VAR-2, –10, and –16 at high winds shows that including the BRCS DDMs and the convolutional layers lead to better predictions in this regime.

Although the CNNs are supposed to directly extract features from the inputs, it is shown that aiding them with traditional features can lead to higher prediction accuracies. For instance, the combination of CNNs with hand-crafted features enhances aggregative image classification (Tianyu et al., 2018). Similarly, theoretical remote sensing knowledge has been increasingly combined with deep learning to further advance its performance (Cao et al., 2018; Zhu et al., 2017; Ma et al., 2019). The analysis here also demonstrates that a stand-alone CNN, without fully connected layers to incorporate the ancillary parameters along with the DDM, results in an RMSE of 1.71 m/s. A fully-connected architecture, such as FC-VAR-2 taking NBRCS and LES as the input features, still performs better than the CNN architecture with an RMSE of 1.45 m/s. Hence, the combination of direct feature extraction from BRCS DDMs using convolutional layers and the incorporation of ten ancillary

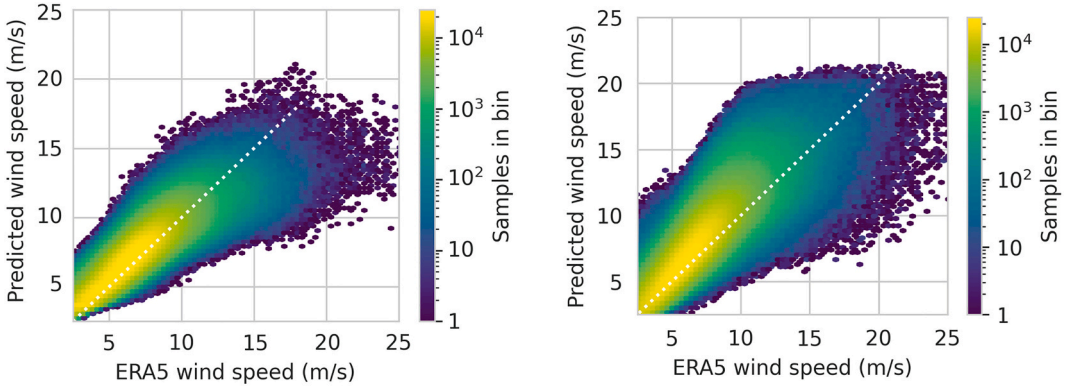


Fig. 5. Log-scale density plot of predicted wind speeds, CNN-VAR-10 architecture (a) and Minimum Variance Estimator (b), versus ERA5 wind speed.

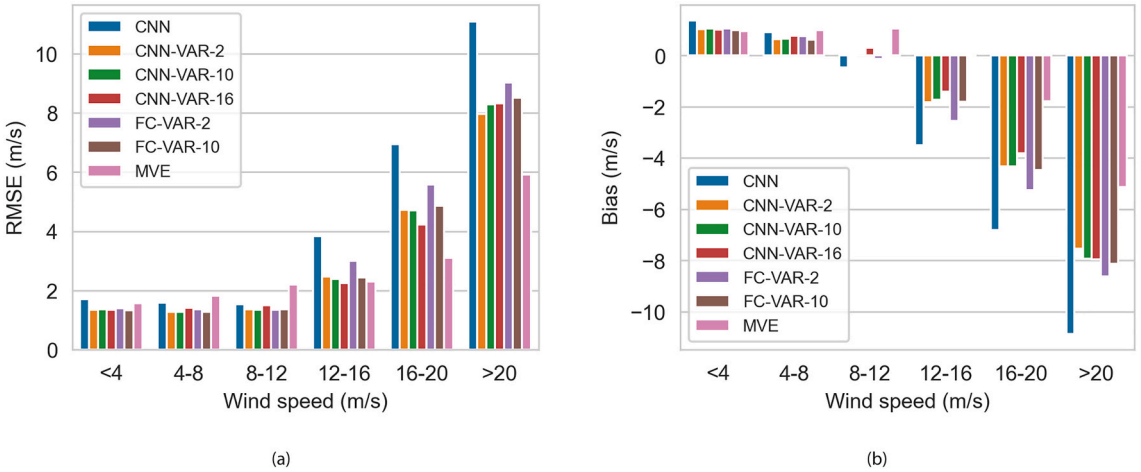


Fig. 6. RMSE for different ranges of the ERA5 wind speed (a), bias for different ranges of the ERA5 wind speed (b).

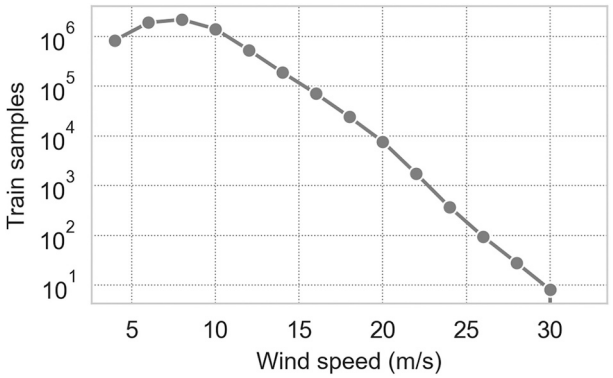


Fig. 7. Number of samples per bin in the training set.

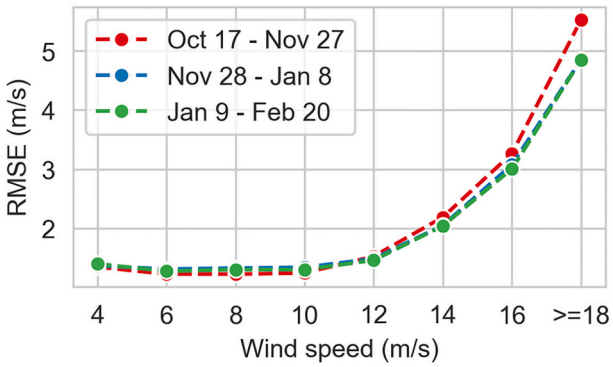


Fig. 8. RMSE in three different time intervals of the test set.

features including the NBRCS and LES, offers optimal performance, RMSE of 1.36 m/s. Comparing the performance of CNN-VAR-10 and -16, we can conclude that adding a higher number of ancillary parameters could not reduce the RMSE over the test data, although these ancillary parameters have been able to improve the performance on the validation set and, based on this fact, have been selected as additional input parameters into the fully connected layers. The performance of the architectures over the validation set is given in Table S2 in the supplement.

5.2. Comparison with conventional models

The MVE is currently operational and is the baseline algorithm for wind speed estimation. In terms of the statistics in Table 3, CyGNSSnet with CNN-VAR-10 architecture outperforms the MVE at winds below 16 m/s and result in a lower value of the general RMSE by 28%. Fig. 9 also confirms that CyGNSSnet generally performs better than the MVE with improved RMSE and bias globally. The reported RMSE of 1.38 m/s is also promising in comparison to the assessment in Ruf et al. (2018b). In that evaluation study, an RMSE of 1.4 m/s is reported for winds lower than 20 m/s, however, one should consider the differences in the data temporal length, version, quality control, filtering, and validation methodology. The CyGNSSnet also shows an encouraging performance compared to recent models based on fully-connected layers processing CyGNSS data (Reynolds et al., 2020; Liu et al., 2019). Note that in this study we evaluate on a temporally separated test set.

Both the MVE and CyGNSSnet underestimate high wind speeds which can be partially due to the sensitivity saturation of DDM observables at very high winds (Ruf et al., 2018b). This is a common issue with radar scatterometers, performance of which is degraded at high wind speeds (Zeng and Brown, 1998). The MVE still performs better at winds higher than 16 m/s, as shown in Fig. 6. Less than 0.5% of training samples have wind speeds beyond 16 m/s (Fig. 7). The challenging wind speed determination in this regime, combined with the limited amount of data and the network tendency to regress to average values, limits the performance of the deep learning algorithm in this range, and MVE outperforms CyGNSSnet.

Fig. 9 shows spatial patterns of overestimation especially in Asia-Pacific regions at longitudes between 50° W and 0°. This area is strongly affected by the Quasi-Zenith Satellite System (QZSS) L-band signals which can be potentially a source of radio-frequency interference (RFI). The RFI caused by other L-band signals, especially those from Satellite Based Augmentation Systems (SBAS), have been already known as a source of SNR degradation and overestimation of GNSS-R winds (Querol et al., 2016). Similar overestimations in equatorial regions in TDS-1 measurements are also reported (Asgarimehr et al., 2018a). The similarity of the patterns in both MVE and CyGNSSnet predictions confirm that they are measurement-derived and not caused by the retrieval methods.

5.3. Existing challenges

Empirical approaches, both conventional algorithms such as the MVE, and deep learning models suffer from mislabeling at extreme ranges. The new CyGNSS dataset (version 3.0), includes the measurements from the block IIF GPS observations which are filtered out in the current datasets (version 2.1) due to unknown transmitter antenna gain patterns (Balasubramaniam and Ruf, 2021). Besides, the improved estimation of EIRP in the new datasets could lead to more accurate

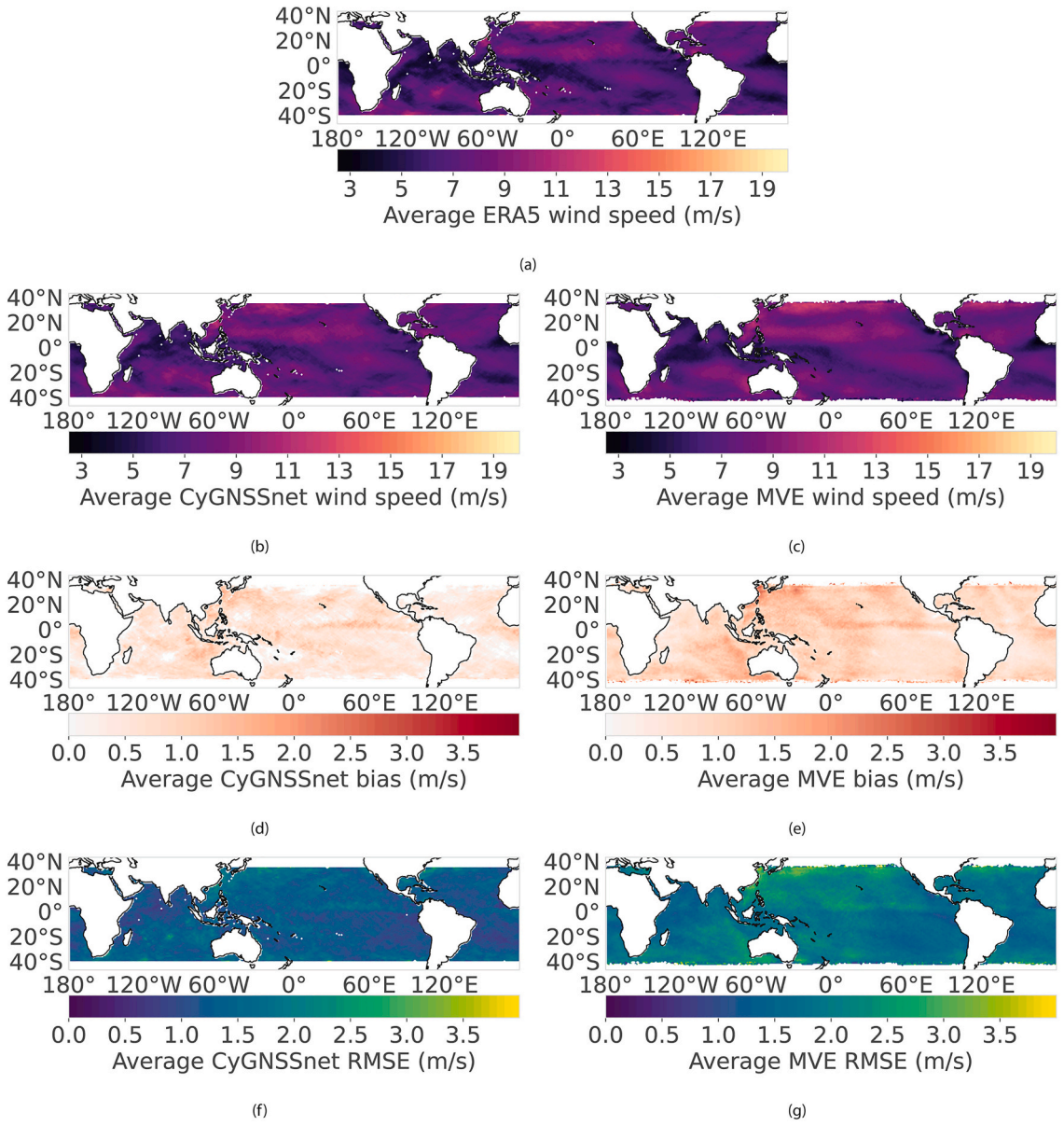


Fig. 9. True wind speed labels from ERA5 (a). Average predicted wind speed (b,c), computed bias (d,e) and RMSE (f,g) of CyGNSSnet and Minimum Variance Estimator gridded with 1 degree resolution, using the test data (October 17, 2018 - February 21, 2019).

predictions at high wind speeds.

Also, in the regime of very low and extremely high wind speeds, wind speed is highly variable and changes rapidly. For training deep learning models as well as determining conventional Geophysical Model Functions (GMFs), DDM observables are related to observations to near-coincident measurements of the wind speed estimated by other instruments or reanalysis models. The probability of mismatching (mislabeling) of the measurement due to the high variability of wind speed increases at extreme ranges as perfect temporal and spatial concurrency is impossible in practice.

The "tendency to the mean" can be the main issue of training AI models with unbalanced datasets. The main challenge faced in wind speed regressions is that the deep learning architectures deal with unevenly distributed samples over wind speed. During the training, due to the substantially larger amount of data at moderate winds, the architecture tends towards conservative predictions at extreme ranges. In other words, due to the intrinsic averaging in the determination of loss values such as Mean Squared Error (MSE) and Mean Absolute Error (MAE), the loss value is marginally affected by mispredictions at high winds and it will be close to its optimal value although the performance at extreme regimes is not optimized. This effect is also well discussed for the case of precipitation nowcasting, see Franch et al. (2020) and references therein. This can be also a problematic issue for high-order conventional parametric regressions and is a significant challenge to be addressed for future GNSS-R deep learning models. One possibility would be to train specialized models for high wind speeds. The increased performance of the CNN-VAR-16 architecture, vs. the CNN-VAR-10 architecture in this regime, is promising in that regard.

Besides, the relationship between DDMs and wind speed is dependent on sea age. The model determinations normally assume that the sea is fully developed, in which the local wind has conferred its maximum energy to the waves. At high winds, the sea condition can be similar to partially developed or young seas resulting in misleading higher power in DDMs and consequently in BRCS and LES. This is also a challenge for developing the conventional algorithm, where a Young Seas/Limited Fetch GMF is determined for higher winds (Ruf and Balasubramaniam,

2018).

6. Concluding remarks

CyGNSSnet is a deep learning model for wind speed predictions from CyGNSS measurements. Outperforming the existing models, the CyGNSSnet shows a significant improvement in the general RMSE (28%) compared to the baseline winds derived from the algorithm based on minimum variance estimations. However, reaching MVE accuracy is still a challenge at higher winds. Addressing the unbalanced data with novel architectures, and the improvement of the predictions in this regime remains a future area of study. We believe that CyGNSSnet is still one of the first GNSS-R deep learning models encouraging further investigations.

CyGNSSnet shows that GNSS-R ocean wind speed monitoring can benefit from deep learning. Beyond this application, the technique is particularly beneficial for other GNSS-R application domains where the existing models do not describe the underlying physical processes well, such as in GNSS-R soil moisture monitoring where the effects of surface roughness and vegetation is still a challenge (Camps et al., 2020). Nevertheless, we believe that the combination of physics-based and deep learning methods is an up-and-coming direction. It was shown that the model still needs the LES and NBRCS for better prediction accuracy, the extraction of which from DDMs relies on the theoretical models.

Deep learning offers the capability of correcting the effects dictated by the data. As the next step, CyGNSSnet will be further trained by ancillary data on bias-causing parameters, such as precipitation (Asgarimehr et al., 2018b, 2021; Balasubramaniam and Ruf, 2020a), to correct the associated effects.

Finally, we encourage interdisciplinary efforts within the GNSS-R community to exchange knowledge on GNSS-R and data science domains and so be able to address further challenges and extend the quality and variety of the data products. Deep learning has not reached its level of maturity in the GNSS-R domain in comparison to other remote sensing domains and this study tried to take the step forward. “Big data” form the common point where both data scientific and GNSS-R fields of research meet each other, one offers the technique for the cost-effective Earth data collection and the other the efficient processing approach.

CRediT authorship contribution statement

Milad Asgarimehr: Conceptualization, Methodology, Validation, Data curation, Writing – original draft, Writing – review & editing, Visualization, Formal analysis, Project administration. **Caroline Arnold:** Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing, Visualization, Investigation. **Tobias Weigel:** Writing – review & editing, Methodology, Project administration. **Chris Ruf:** Writing – review & editing. **Jens Wickert:** Writing – review & editing, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This study is conducted with financial supports from the German Research Centre for Geosciences at GFZ, Potsdam, Germany. We are also grateful for the valuable assistance in software development given by Felix Stiehler at the German Climate Computing Centre DKRZ, Hamburg, Germany. CA and TW were funded by Helmholtz Association’s Initiative and Networking Fund through Helmholtz AI [grant number: ZT-I-PF-5-01]. This work used resources of the Deutsches Klimarechenzentrum (DKRZ) granted by its Scientific Steering Committee (WLA) under project ID AIM. The datasets used in this study are available free of charge and we thank scientific teams associated with the CyGNSS mission at NASA and the University of Michigan and ERA5 reanalysis estimates at the European Centre for Medium-Range Weather Forecasts (ECMWF).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.rse.2021.112801.

References

Asgarimehr, M., Wickert, J., Reich, S., 2018a. TDS-1 GNSS reflectometry: development and validation of forward scattering winds. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 11 (11), 4534–4541.

Asgarimehr, M., Zavorotny, V., Wickert, J., Reich, S., 2018b. Can GNSS reflectometry detect precipitation over oceans? Geophys. Res. Lett. 45 (22), 12–585.

Asgarimehr, M., Zhelavskaya, I., Foti, G., Reich, S., Wickert, J., 2020. A GNSS-R geophysical model function: machine learning for wind speed retrievals. IEEE Geosci. Rem. Sens. Lett. 17 (8), 1333–1337. https://doi.org/10.1109/LGRS.2019.2948566.

Asgarimehr, M., Hoseini, M., Semmling, M., Ramatschi, M., Camps, A., Nahavandchi, H., Haas, R., Wickert, J., 2021. Remote sensing of precipitation using reflected GNSS signals: response analysis of polarimetric observations. IEEE Trans. Geosci. Rem. Sens. https://doi.org/10.1109/TGRS.2021.3062492.

Balasubramaniam, R., Ruf, C., 2020a. Characterization of rain impact on L-band GNSS-R ocean surface measurements. Remote Sens. Environ. 239, 111607.

Balasubramaniam, R., Ruf, C., 2020b. Neural network based quality control of CYGNSS wind retrieval. Rem. Sens. 12 (17), 2859.

Balasubramaniam, R., Ruf, C.S., 2020. Performance assessment of cygnss high wind retrieval for the improved eirp calibration. In: IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 5805–5807.

Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B., 2011. Algorithms for hyper-parameter optimization. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K. Q. (Eds.), Advances in Neural Information Processing Systems, Vol. 24. Curran Associates, Inc, NY 12571 United States.

Camps, A., Park, H., Castellví, J., Corbera, J., Ascaso, E., 2020. Single-pass soil moisture retrievals using GNSS-R: lessons learned. Rem. Sens. 12 (12), 2064.

Cao, Q., Zhong, Y., Ma, A., Zhang, L., 2018. Urban land use/land cover classification based on feature fusion fusing hyperspectral image and lidar data. In: IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, pp. 8869–8872.

Chu, X., He, J., Song, H., Qi, Y., Sun, Y., Bai, W., Li, W., Wu, Q., 2020. Multimodal deep learning for heterogeneous gnss-r data fusion and ocean wind speed retrieval. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 13, 5971–5981. https://doi.org/10.1109/JSTARS.2020.3010879.

Clarizia, M.P., Ruf, C.S., Jales, P., Gommenginger, C., 2014. Spaceborne GNSS-R minimum variance wind speed estimator. IEEE Trans. Geosci. Rem. Sens. 52 (11), 6829–6843. https://doi.org/10.1109/TGRS.2014.2303831.

Clarizia, M.P., Ruf, C.S., 2016. Wind speed retrieval algorithm for the Cyclone global navigation satellite system (CYGNSS) mission. IEEE Trans. Geosci. Rem. Sens. 54 (8), 4419–4432.

CYGNSS, 2017. CYGNSS Level 1 Science Data Record Version 2.1. https://doi.org/10.5067/CYGNS-L1X21.

CYGNSS, 2018. CYGNSS Level 2 Science Data Record Version 2.1. https://doi.org/10.5067/cygns-l2x21.

Foti, G., Gommenginger, C., Jales, P., Unwin, M., Shaw, A., Robertson, C., Rosello, J., 2015. Spaceborne GNSS reflectometry for ocean winds: first results from the UK TechDemoSat-1 mission. Geophys. Res. Lett. 42 (13), 5435–5441.

Franch, G., Nerini, D., Pendesini, M., Coviello, L., Jurman, G., Furlanello, C., 2020. Precipitation nowcasting with orographic enhanced stacked generalization: improving deep learning predictions on extreme events. Atmosphere 11 (3), 267.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horányi, A., Muñoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D., et al., 2020. The ERA5 global reanalysis. Quart. J. Royal Meteorol. Soc. 146 (730), 1999–2049.

Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv:1502.03167.

Liu, Y., Collett, I., Morton, Y.J., 2019. Application of neural network to GNSS-R wind speed retrieval. IEEE Trans. Geosci. Rem. Sens. 57 (12), 9756–9766. https://doi.org/10.1109/TGRS.2019.2929002.

Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., Johnson, B.A., 2019. Deep learning in remote sensing applications: a meta-analysis and review. ISPRS J. Photogram. Rem. Sens. 152, 166–177.

Microsoft Corporation, 2020. NNI (Neural Network Intelligence) 1.8. https://github.com/microsoft/nni.

Munoz-Martin, J.F., Capon, L.F., Ruiz-de Azua, J.A., Camps, A., 2020. The flexible microwave Payload-2: a SDR-Based GNSS-reflectometer and L-band radiometer for CubeSats. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 13, 1298–1311.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch:

an imperative style, high-performance deep learning library. In: Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R. (Eds.), Advances in Neural Information Processing Systems 32. Curran Associates, Inc, NY 12571 United States, pp. 8024–8035.

Querol, J., Alonso-Arroyo, A., Onrubia, R., Pascual, D., Park, H., Camps, A., 2016. SNR degradation in GNSS-R measurements under the effects of radio-frequency interference. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 9 (10), 4865–4878.

Reynolds, J., Clarizia, M.P., Santi, E., 2020. Wind speed estimation from CYGNSS using artificial neural networks. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 13, 708–716. https://doi.org/10.1109/JSTARS.2020.2968156.

Ruf, C.S., Balasubramaniam, R., 2018. Development of the CYGNSS geophysical model function for wind speed. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 12 (1), 66–77.

Ruf, C.S., Chew, C., Lang, T., Morris, M.G., Nave, K., Ridley, A., Balasubramaniam, R., 2018a. A new paradigm in earth environmental monitoring with the CYGNSS small satellite constellation. Scientific Rep. 8 (1), 1–13.

Ruf, C.S., Gleason, S., McKague, D.S., 2018b. Assessment of CYGNSS wind speed retrieval uncertainty. IEEE J. Select. Top. Appl. Earth Observ. Rem. Sens. 12 (1), 87–97.

Simonyan, K., Zisserman, A., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv:1409.1556.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958.

Tianyu, Z., Zhenjiang, M., Jianhu, Z., 2018. Combining cnn with hand-crafted features for image classification. In: 2018 14th IEEE International Conference on Signal Processing (ICSP). IEEE, pp. 554–557.

Wang, C., Xi, Y., 1997. Convolutional Neural Network for Image Classification. Johns Hopkins University, Baltimore, MD 21218.

Zavorotny, V.U., Voronovich, A.G., 2000. Scattering of GPS signals from the ocean with wind remote sensing application. IEEE Trans. Geosci. Rem. Sens. 38 (2), 951–964. https://doi.org/10.1109/36.841977.

Zavorotny, V.U., Gleason, S., Cardellach, E., Camps, A., 2014. Tutorial on remote sensing using GNSS bistatic radar of opportunity. IEEE Geosci. Rem. Sens. Mag. 2 (4), 8–45. https://doi.org/10.1109/MGRS.2014.2374220.

Zeng, L., Brown, R.A., 1998. Scatterometer observations at high wind speeds. J. Clim. Appl. Meteorol. 37 (11), 1412–1420.

Zhu, X.X., Tuia, D., Mou, L., Xia, G.-S., Zhang, L., Xu, F., Fraundorfer, F., 2017. Deep learning in remote sensing: a comprehensive review and list of resources. IEEE Geosci. Rem. Sens. Mag. 5 (4), 8–36.