

Downscaling wind fields using deep learning



Gisli Björn Helgason

DTU Wind Energy-M-0528

June 2022

Author: Gisli Björn Helgason

DTU Wind Energy-M-0528
June 2022

Title:
Downscaling wind fields using deep learning

Project period:
January - June 2022

ECTS: 30

Education: Master of Science

Supervisors:
Ásta Hannesdóttir
Jens Visbeck Madsen
DTU Wind Energy

Gerald Steinfeld
Uni. Oldenburg

Remarks:
This report is submitted as partial fulfillment of the requirements for graduation in the above education at the Technical University of Denmark.

DTU Wind Energy is a department of the Technical University of Denmark with a unique integration of research, education, innovation and public/private sector consulting in the field of wind energy. Our activities develop new opportunities and technology for the global and Danish exploitation of wind energy. Research focuses on key technical-scientific fields, which are central for the development, innovation and use of wind energy and provides the basis for advanced education at the education.

Technical University of Denmark
Department of Wind Energy
Frederiksborgvej 399
4000 Roskilde
Denmark
www.vindenergi.dtu.dk

Approval

This thesis has been prepared over six months at the division for Wind Energy Systems, Department of Wind Energy, at the Technical University of Denmark, DTU, in partial fulfilment for the degree Master of Science in Engineering, MSc Eng.

Gísli Björn Helgason - s203357

Gísli Björn Helgason
Signature

24.6.2022
Date

Abstract

Deep learning methods such as Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs) can potentially greatly accelerate the production of near-surface wind fields while limiting the amount of computational and time intensive dynamical downscaling. Such methods have recently achieved great success in the field of image super-resolution (SR), a problem closely related to downscaling. This thesis aims to quantify the applicability of such methods for wind field downscaling in the purpose of preliminary wind resource assessment, typically associated with extensive regional weather model simulations. Advanced SRGAN and SRCNN models are trained to estimate plausible high-resolution (HR) wind velocity values conditioned on given low-resolution (LR) wind fields and additional HR topographic data. Mesoscale wind fields from the New European Wind Atlas (NEWA) project are used as ground truth HR targets and LR input data is obtained by averaging the true HR wind fields to mimic the resolution of the forcing ERA5 reanalysis data. The trained super-resolution is thus 10 x, operating on scales from roughly \sim 30 km to 3 km. This idealized approach aims to quantify model errors associated with unresolved atmospheric scales and terrain effects, circumventing issues related to the projection mismatch between global domain and regional numerical weather simulations. An extensive analyses of the reconstruction errors shows that systematic reconstruction errors are associated with prominent topographic features. When provided, the models are able to effectively use HR topographic information to reduce these systematic errors leading to a near zero wind speed magnitude bias while angular deviations persist in complex terrain.

Acknowledgements

Gísli Björn Helgason, MSc Wind Energy, DTU
Author of this Master's thesis work

First of all, I would like to thank my supervisors, Dr. Ásta, Dr. Gerald and Jens, for their support, encouragement and patience, as well as their valuable time and insightful comments during our weekly meetings. They provided me with freedom to explore different scopes of research and helpful advice these last few months, for which I am grateful. A special thanks also go to Wilke Trei for the HPC user support at Oldenburg and Landsvirkjun, the National Power Company of Iceland, for supporting this project financially. My family has of course played a vital role in me completing this thesis. I thank my parents for the unconditional love and support, my siblings for being my role models and motivation and my nephews and nieces for providing me with moments of joy during stressful academic moments. To conclude, I cannot forget to thank my friends, new and old, for all their invaluable support.

Contents

Abstract	v
Acknowledgements	vi
1 Introduction	1
1.1 Motivation	1
1.2 Literature review	2
1.3 Aim	5
1.4 Outline	6
2 Theoretical background	7
2.1 Downscaling approaches	7
2.2 Key concepts of deep learning	8
2.3 Generative Adversarial Networks	13
2.4 Image super-resolution	14
3 Data and methodology	17
3.1 Data	17
3.2 Model architecture	20
3.3 Training	23
3.4 Generator ablation study	24
3.5 Model evaluation	26
4 Results	29
4.1 Final model training	29
4.2 Quantitative analysis	29
4.3 Spatial distribution of prediction errors	33
4.4 Wind climatology	36
4.5 Spectral analysis	38
5 Discussion	41
6 Conclusions	45
6.1 Further work	45
Bibliography	47
A Supporting material	53
A.1 Synoptic setting	53
A.2 Wind rose	53
B Ensemble Analysis	54
C Training	59
C.1 WGAN training algorithm	59
C.2 Loss curves	60

List of Figures

1.1	Illustration of the dynamical and SR downscaling process	3
1.2	CNN architectures considered in Höhlein et al. (2020).	3
1.3	SRGAN framework for wind field downscaling (Stengel et al., 2020).	4
2.1	Illustration of the 2D convolutional operation	10
2.2	(a) Application of a volumetric 2D convolutional filter to a full-color RGB image and (b) a 3D visualization of a convolutional layer, where each convolutional filter corresponds to a slice in the resulting output volume (matched in grey-scale). Both images taken from Buduma and Locascio (2017).	11
2.3	Illustration of the ReLU activation function and its derivative.	11
2.4	Residual learning building block and UNet schematic	12
3.1	Workflow of training data generation and model training framework.	17
3.2	Annual global domain statistics based on CE domain NEWA data (100 m above ground) for the period 2007-2018. Mean statistics over the full NEWA period and the 12 year study period are shown for reference with dash dotted lines.	18
3.3	(a) The Central Europe (CE) and (b) the Great Britain (GB) NEWA model domains.	19
3.4	Average annual (a) mean wind speed magnitude and (b) standard deviation at 100 m above ground from the CE NEWA training set (2011-2016).	19
3.5	Average annual (a) mean wind speed magnitude and (b) variance at 100 m above ground from the GB NEWA domain data covering the training period (2011-2016).	20
3.6	Architecture of the original Generator and Discriminator network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer. Taken from Ledig et al. (2017).	20
3.7	Network modifications related to the ingestion of HR static data	22
3.8	Statistics of normalized time per epoch during training and the evolution of MSE loss on the standardized training set. The maximum training time for CNN2 (used for the normalization) is beyond the scale of the plot.	25
3.9	Topographic map showing the considered site locations in CE domain	28
4.1	Comparison of various SR methods on NEWA CE domain data fields.	30
4.2	Comparison of various SR methods on NEWA GB domain data fields.	31
4.3	Statistics of spatially averaged MSE on the test set, for the CE NEWA domain (left) and the GB domain (right)	32
4.4	Mean magnitude difference (top row) and mean cosine deviations (bottom row) between target high-resolution NEWA CE domain data and prediction of a) CNN1, WGAN and bicubic baseline models without HR static data and b) CNN1, CNN2 and WGAN with HR static data. The average is taken over all four years in the test set.	34
4.5	Mean magnitude difference (top row) and mean cosine deviations (bottom row) between target high-resolution NEWA GB domain data and prediction of a) CNN1, WGAN and bicubic baseline models without HR static data and b) CNN1, CNN2 and WGAN with HR static data. The average is taken over all four years in the test set.	35

4.6	Climatological mean and standard deviation of wind magnitude from the CE test set. Results shown for target data and topographically aware CNN1 and WGAN model output.	36
4.7	Climatological mean and standard deviation of wind magnitude from the GB test set. Results shown for target data and topographically aware CNN1 and WGAN model output.	37
4.8	Normalized histogram showing the distribution of wind speed magnitude at three different locations in the CE domain covering the entire test set period. Weibull fits to the target data (blue) and model output (yellow and green) are shown with solid lines. Bin-wise difference in frequency between target and modelled data is also illustrated.	37
4.9	Radially averaged 2D spectrum. Dashed line depicts the expected -5/3 spectral decay in turbulent mesoscale flow.	39
A.1	Synoptic setting over Europe and North Atlantic Ocean	53
A.2	Wind rose	53
B.1	Comparison of WGAN model output and test target from the CE domain.	55
B.2	Comparison of WGAN model output and test target from the GB domain.	56
B.3	Ensemble standard deviation of 100 images generated by the stochastic WGAN models with and without HR static data. Ground truth target from the CE domain, not following the same color grading, is shown for reference.	56
B.4	Ensemble standard deviation of 100 images generated by the stochastic WGAN models with and without HR static data. Ground truth target from the GB domain, not following the same color grading, is shown for reference.	57
B.5	Radially averaged 2D spectrum for WGAN output and target data. Dashed line depicts the expected -5/3 spectral decay in turbulent mesoscale flow.	57
C.1	Training and validation loss curves for CNN1	60
C.2	Training and validation loss curves for CNN1 - HR static.	60
C.3	Training and validation loss curves for CNN2 with HR static input.	61
C.4	Training and validation loss curves for WGAN.	62
C.5	Training and validation loss curves for WGAN - HR static.	63

List of Tables

3.1	Increase in the number of trainable model parameters of the generator network CNN variants, given as percentage relative to the Stengel et al. (2020) design (CNN1 without HR static data).	22
3.2	Batch size and number of epochs during training	24
3.3	MSE score on unseen validation data along with relative improvement compared to the original design.	26
3.4	Latitude and longitude coordinates of the considered sites.	28
4.1	Descriptive statistics of domain averaged MSE for the considered models on the CE domain.	32
4.2	Descriptive statistics of domain averaged MSE for the considered models on the CE domain.	33
4.3	Descriptive statistics of domain averaged MSE for the considered models on the CE domain.	33
4.4	First moment and fitted Weibull parameters derived from the target data and topographic aware CNN1 and WGAN model output at three sites from the CE test set. Relative difference between target and model output shown for comparison.	38
B.1	Domain averaged MSE [m^2/s^2] for the WGAN model output on one example from the test set.	58
B.2	Relative improvement in domain averaged MSE compared to deterministic approach.	58
C.1	Pseudo code for WGAN with gradient penalty (Gulrajani et al., 2017)	59

1 Introduction

1.1 Motivation

Wind energy is an important and growing source of renewable energy in the world, currently contributing to around 6.7% of the world electricity generation (IEA, 2021). By the end of 2020 the cumulative global wind power installations reached 743 GW and before 2025 the wind power capacity worldwide is projected to exceed 1TW (GWEC, 2021b). The growth of wind power capacity over the last decade has been lead by China, North America and Europe with primarily onshore deployment, while the share of offshore wind power has increased along with its rapid development. Offshore wind is deemed to have the biggest growth potential of any renewable energy source (GWEC, 2021a).

Accurate wind resource assessment is an integral part of the planning and development of wind farms due to the variable nature of wind. This involves estimating the mean wind climate at the sites of interest in the form of wind direction and sector-wise wind speed probability distribution functions, providing important information for siting, sizing and detailed design of the wind farm (Mortensen, 2020). Preliminary wind resource assessment based on modelled long-term and large-scale weather data can provide reliable information for physical planning on national, regional or continental scales. Conventional wind resource assessment and siting calculations are however based on measured wind data at or nearby the site of interest using the wind atlas methodology introduced by Troen and Petersen (1989).

Low-level (boundary layer) wind speed and direction are mainly determined by localized or regional interactions between the large-scale airflow and the surrounding topography, with added complexity of thermal forcing. Local orographic features give rise to speed up effects and surface roughness changes lead to internal boundary layers that alter the vertical wind profile which is further affected by the stability of the atmosphere. These effects are of high importance to wind energy as onshore wind farms may preferably be deployed on top of hills or ridges, utilizing speed-up effects for more energy production, and most offshore installation have been situated in shallow waters within the influence of the land-sea discontinuity. Severe computational expenses limit the availability of high-resolution numerical weather data, especially for large spatial domains. The current resolution of computationally affordable global weather models is not sufficient to accurately reproduce wind climatology over regions with complex orography and heterogeneous surfaces. Refinement of global atmospheric data is thus needed in order to obtain more reliable low-level winds, enabling more accurate localized assessments of the potential long-term economic viability of wind power projects. Mesoscale weather models, like the weather research forecasting (WRF) model (W. Skamarock et al., 2008), are commonly used to refine global atmospheric data for wind energy purposes, referred as dynamical downscaling. These models are still computationally demanding, incorporating complex physical laws governing atmospheric motion across scales and various approximations for near-surface processes affecting the atmospheric boundary layer flow, but deep neural networks (DNN) potentially offer a much cheaper solution after learning the refinement process from dynamically downscaled data.

1.2 Literature review

In the field of computer vision there exists a classical image processing problem known as super-resolution (SR). Single-image SR (SISR) refers to the process of enhancing a low-resolution (LR) image, producing an approximated image of the true high-resolution (HR) version of it (C. Y. Yang, Ma, and M. H. Yang, 2014). This can formally be viewed as a very similar task to downscaling of atmospheric fields. Powerful data-driven deep learning methods, in particular convolutional neural networks (CNNs), have recently achieved state-of-the-art performance when applied to SISR (W. Yang et al., 2019). CNNs are well suited to learn tasks involving spatially distributed data and have generalization abilities in contrast to many other high level statistical methods. They have thus emerged as a promising technique in downscaling atmospheric fields. Vandal et al. (2018) were among the first to apply SRCNNs for climate fields, adopting the original SRCNN architecture by Dong, Loy, He, et al. (2014). More recent studies on deep learning based downscaling of atmospheric fields have further exploited more advanced CNN architectures and improved training schemes but such research is still in its early stages. Among the most promising deep learning frameworks in this regard are Generative Adversarial Networks (GANs) (Goodfellow et al., 2014). They have recently been developed for SR tasks (Ledig et al., 2017)(X. Wang et al., 2018) and the enhancement of atmospheric fields (Leinonen, Nerini, and Berne, 2020)(Stengel et al., 2020)(J. Wang et al., 2021)(Harris et al., 2022)(Price and Rasp, 2022).

Directly applying state-of-the-art SR algorithms from computer vision literature to meteorological data involves some challenges. Most SR architectures have been developed for natural images with coherent objects that have well defined shapes and contrasts, which is largely not the case for meteorological data. The large scale gap between global and regional weather models also requires more extreme SR than most networks are designed for. While a single plausible solution is often acceptable in the case of natural images, it tends to be insufficient in climate and weather applications where the quantification and understanding of uncertainty is highly valuable (Leinonen, Nerini, and Berne, 2020). Appropriate data pairs are furthermore needed for CNNs to approximate the complex mapping between LR and HR atmospheric fields. Regular gridded data is the preferred choice for CNNs, in order to utilize parallel computations on graphics processing unit (GPU) based hardware to efficiently solve optimization tasks. Projection mismatch and different grid structures between available HR and LR numerical simulations thus poses some difficulties for CNNs. Interpolation can be used for regridding, but this generally leads to smoothing that can damage relevant structures and features in the flow field. An approach to circumvent this problem is to coarsen the HR fields and use them as surrogate LR model inputs for training the CNNs (Vandal et al., 2018)(Stengel et al., 2020) while deploying the models on global climate model data. Training CNNs in such an idealized setting is believed to conceive more generalized networks across different numerical inputs. This framework however does not address the correction of specific global forecast model errors and the resulting networks are expected to be negatively affected by unseen biases that exist in real LR simulations. To address this J. Wang et al. (2021) propose a learned multi-resolution dynamical downscaling approach that combines the strengths of the dynamical and DNN-based downscaling. Their training data is constructed from low- and high-resolution simulations conducted separately using a mesoscale model (WRF) with the same dynamical forcing. This approach outperformed a more generally trained SR network when applied to real LR data (the coarser WRF simulation) to downscale precipitation. An illustration of these two frameworks with reference to dynamical downscaling is shown in Figure 1.1. ERA5 reanalysis (Hersbach et al., 2020) is commonly used as global dynamic forcing data (~ 30 km grid spacing) for retrospective WRF simulations

in the wind energy community.

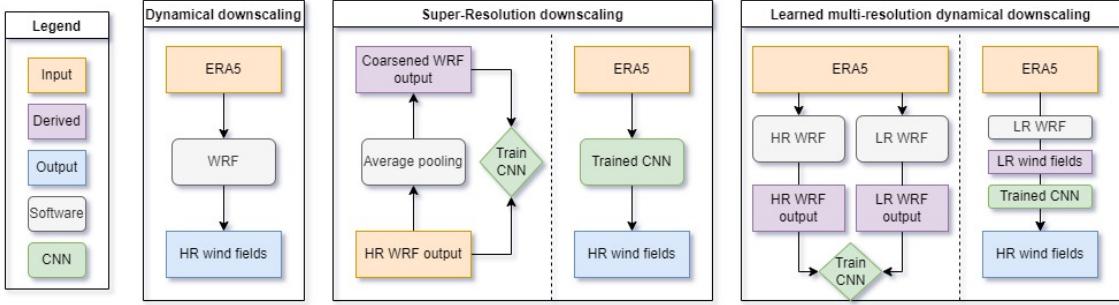


Figure 1.1: Illustration of the dynamical, multiresolution dynamical and SR downscaling process to produce high-res (HR) wind fields, based on (J. Wang et al., 2021).

Höhlein et al. (2020) performed a comparative study of several CNN architectures, with varying degree of model complexity as illustrated in Figure 1.2 and described further in subsection 2.4.1, for downscaling wind fields. More specifically the horizontal wind components 100 m above ground. As LR input to their models, they use short-range forecasts from the ERA5 reanalysis data archive. The higher-resolution target dataset contains short-range forecasts from the operational HRES ECMWF Integrated Forecast System (ECMWF, 2019) with ~ 9 km spatial resolution. Land surface elevation and a binary land-sea mask from both datasets are used as static predictors, along with supplementary dynamical predictors from the LR data. These additional variables are the boundary layer height, forecast surface length and geopotential height at 500 hPa. The data is defined on irregular (octahedral and triangular reduced Gaussian) grids where the number of grid points per latitude circle decreases with increasing latitude, imposing some difficulties for CNNs to properly learn spatial correlations. To preserve the spatial adjacency of most grid points, the data is copied onto rectangular 2D grids with appropriate padding. The training domain is limited to Central Southern Europe, covering various terrain such as the Alps, parts of the Mediterranean and Adriatic Sea and flat agricultural areas in France. All models were trained to minimize mean square error (MSE) loss, with several network variants regressing on different set of LR dynamic predictor fields and HR topographic data.

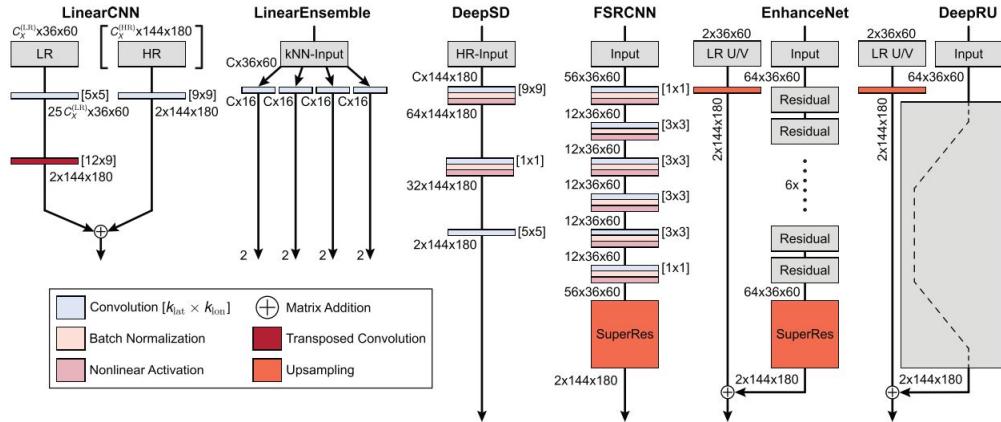


Figure 1.2: CNN architectures considered in Höhlein et al. (2020).

The prediction accuracy of a linear ensemble baseline model was higher than achieved

by shallow and nonlinear CNN architectures. Deeper nonlinear CNNs were however able to compete with the prediction quality of the baseline and achieved superior results when more predictors and high-resolution topographic information were fed into the models. The best performing model was found to be a UNet architecture (Ronneberger, Fischer, and Brox, 2015) with residual blocks (He et al., 2016a). An analysis of feature importance showed that the trained networks base their predictions mainly on the LR wind speed predictors, with stronger error propagation with more complex models, along with LR and HR topographic information.

Training CNNs solely to minimize pixelwise loss leads to overly smooth results that lack the expected high fidelity of the true HR data (W. Yang et al., 2019). GANs provide a framework that produces more visually appealing output (Ledig et al., 2017) but are generally harder to train. The GAN framework is based on a game between two competing networks, a generator and a discriminator which strives to discriminate between real and generated data. Stengel et al. (2020) trained a slightly modified variant of the SRGAN by Ledig et al. (2017) to operate on scales from 100 km input data (such as global climate simulations) to 2 km mesoscale wind fields. An illustration of the conditional GAN training framework is shown in Figure 1.3. For training they used WRF simulations covering the entire continental United States as HR targets and the same averaged wind fields as LR inputs. The networks were trained in a two step process going from LR to MR with 10 x SR and then from MR to HR with 5 x SR. A deeper single-network approach (performing one-step 50 x SR) exhibited very slow learning. The generator of the model was initially trained without the discriminator in order to roughly learn the enhancement process and further stabilize the GAN training. The additional adversarial loss term involved with the GAN training scheme was found to effectively enforce the network to produce more physically consistent mesoscale fields than the same CNN only trained to minimize pixelwise loss, i.e. without the discriminator, in terms of intermittent turbulent behaviour measured by an increment probability density function and 5/3 decaying power spectrum.

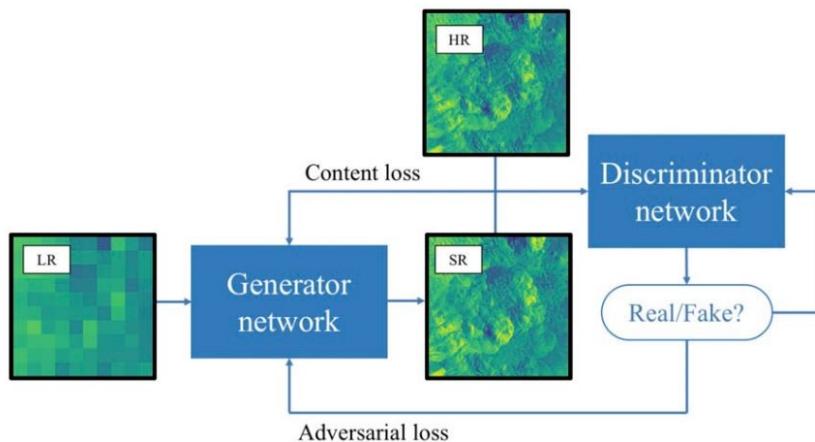


Figure 1.3: SRGAN framework for wind field downscaling (Stengel et al., 2020).

(Tran et al., 2020) similarly used the Enhanced SRGAN (ESRGAN) (X. Wang et al., 2018) to map mesoscale wind fields at 2.5 km to a much finer resolution at 200 m, suitable for wind farm siting calculations. The HR data was generated by a coupled meso-micro scale physics based solver (HARMONIE-SIMRA) over mountainous terrain in Norway. As coarse scale inputs they used downsampled HR fields, representing the mesoscale solver resolution of 2.5 km, via nearest neighbour algorithm.

Annau (2022) has recently implemented the SRGAN in a more stable Wasserstein GAN framework (Arjovsky, Chintala, and Bottou, 2017) with gradient penalty (Gulrajani et al., 2017) to perform downscaling of surface wind fields (10 m wind components). The networks are trained in a non-idealized setting, with input data taken from the Era-Interim reanalysis (Dee et al., 2011) in roughly 80 km resolution and mapped to coarsened WRF fields at 10 km resolution over the continental US. His main findings are that WGANs perform nicely for non-idealized coarse and fine resolution fields and that WGANs introduce realistic fine scale spatial variability not present in the coarse fields. WGAN with gradient penalty is very stable, even in the case of many predictor fields. The surface pressure, surface roughness length and convective available potential energy were considered as additional dynamic input fields along with LR static elevation data and land-sea-mask. Frequency separation of losses, that can help deal with noisy images (Fritsche, Gu, and Timofte, 2019), did not improve the training but reveals that rigid pixelwise metrics do not necessarily capture quality in extreme SR and that physics informed analyses can further help to quantify the performance of the models.

Using the stochastic nature of GANs, Leinonen, Nerini, and Berne (2020) trained a network to produce ensembles of reconstructed high-resolution atmospheric fields given coarsened input data. They further model the time evolution of the generated atmospheric fields with a recurrent network design. Harris et al. (2022) demonstrated that the approach proposed by Leinonen, Nerini, and Berne (2020) can be extended to the more challenging problem of increasing both the accuracy and resolution of a low-resolution weather forecasting model, using HR radar measurements as a "ground truth". Price and Rasp (2022) similarly explore the use of a GAN to simultaneously correct and downscale global ensemble forecasts of precipitation, leveraging HR radar measurements as ground truth. Their trained networks approach the performance of operational NWP and invite the open question if deep neural networks can even replace regional numerical models for HR predictions of some atmospheric fields.

1.3 Aim

The central question of this study can be stated as:

"Can deep learning be considered a reliable technique for mesoscale wind field downscaling in the purpose of wind resource assessment?"

To investigate this, an advanced conditional SRGAN architecture is trained to enhance low-level wind fields. The trained resolution enhancing process estimates plausible HR wind fields conditioned on LR wind fields and HR topographic data. Mesoscale windfields from the NEWA database (NEWA, n.d.) are used to train the networks with coarsened HR wind fields used as surrogate global model inputs. See Dörenkämper, Olsen, et al. (2020) and Hahmann et al. (2020) for a detailed description of the data source. The aim is to establish a model that is able to correct errors in LR simulated data due to unresolved scales and terrain-related physical processes, but the training framework does not tackle specific weather model errors associated with imperfect physics and initial/boundary conditions. To obtain a more clear perspective on the former source of model errors, the trained models are evaluated solely on their ability to reconstruct target data from their LR rendering. They are thus not tested on real global model weather data as done in other research adopting the idealized SR training framework. Previous work has shown that CNNs are able to compete with and outperform other statistical downscaling approaches for large spatial domains especially when associated with additional input fields such as topographical static data. CNNs trained in a GAN framework have further been found to produce more physically consistent wind fields in terms of small scale flow features that

are reflected in a more realistic spectral behaviour. Their performance with respect to domain characteristics have however not been evaluated in great detail. Key objectives in this study are

- to further quantify how and if HR topographical data can be used to improve the reconstruction ability and spatial generalisability of CNNs and,
- to investigate the nature of high-frequency features in GAN model output.

1.4 Outline

The following pages first provide a brief review on the theoretical foundation of this work, covering the different downscaling approaches, key concepts of deep learning and relevant aspects related to deep learning based image super resolution. Following is then a data and methodology chapter describing the data processing and the model architecture, training and evaluation methods. Last pages finally contain results, discussions and conclusions and further work.

2 Theoretical background

2.1 Downscaling approaches

Atmospheric flow is determined by dynamics across a broad-variety of spatio-temporal scales of motion, largest being of planetary scale such as Rossby waves that cover several thousands of kilometers in space and months in time. Smaller are synoptic scale pressure systems and fronts, covering hundreds of kilometers and spanning days or weeks, and smaller still are mesoscale processes such as thunderstorms, convective systems and land-sea breeze, spanning tens of kilometers and several hours up to a day. Smaller than that are microscale turbulent motions (Holton and Hakim, 2013).

Accuracy of weather models is largely based on their ability to resolve atmospheric motions across these scales. Their effective resolution is determined as the scale at which the model kinetic energy spectrum decays relative to the expected or observed spectrum (W. C. Skamarock, 2004), and has been shown to be around 8 times the grid spacing for current operational weather models (Saleh Abdalla and Wedi, 2013). Flow features smaller than that are strictly not to be taken as physical. Observations suggest that mesoscale flow (highly stratified and 2D in nature) has $k^{-5/3}$ dependency (Lindborg, 1999)(Nastrom and Gage, 1985). This behaviour is not fully understood on theoretical grounds but is likely the result of a cascade processes in turbulent kinetic energy (W. C. Skamarock, 2004)(Lindborg, 1999).

Unresolved atmospheric processes and other land-surface interactions are more influential in complex terrain. Refinement of globally simulated wind data is needed in such areas for more accurate assessment of the wind climatology. This scale gap can be bridged to a certain degree by the act of downscaling, i.e. inferring information about physical variables at local scale from coarsely simulated data using appropriate refinement processes, avoiding costly high-resolution simulations over large spatial scales. Such methods can be broadly classified into dynamical and statistical approaches.

2.1.1 Dynamical downscaling

Mesoscale models are often used to resolve atmospheric flow down to the range of 1-10 km for limited regions of the globe. Global analysis or reanalysis weather and sea surface data supply the data for lateral and initial boundary conditions, needed to numerically solve a set of so called primitive equations that govern atmospheric flow with various simplifications and assumptions. The open-source Weather Research and Forecasting (WRF) (W. Skamarock et al., 2008) model is one of the most renowned, especially in the wind energy community. The setup of WRF typically involves nested domains that pass information from the low resolution forcing data to the high resolution grid in a physically consistent manner. Such mesoscale models do not completely resolve important boundary-layer processes occurring at even smaller scales of motion such as radiation, cloud microphysics and turbulence. Although these processes are interacting with each other, the one with most direct impact in the context of wind energy is turbulent mixing. The mean flow characteristics for turbulent flow conditions is modeled by planetary boundary layer (PBL) schemes that parameterize turbulent mixing of momentum, heat and moisture (W. C. Skamarock, 2004).

The New European Wind Atlas (NEWA) project followed a well-defined scientific methodology based on best-practise to produce an almost bias-free prediction of the long-term wind resource (Rodrigo et al., 2020). Hahmann et al. (2020) and Dörenkämper, Olsen,

et al. (2020) describe the model configuration, production and evaluation of the results against observations. The mesoscale part of NEWA was created from WRF model simulations (version 3.8.1) for all of Europe at a grid spacing of 3 km x 3 km, covering 30 years from 1989 to 2018. The WRF model simulations were done in 10 partly overlapping domains that each where the inner-most domains in a 3-domain nested setup, sharing a common parent domain with one-way nesting. The simulation length was 8 days at a time (24 h spin up time), with dynamic forcing from ERA5 reanalysis (Hersbach et al., 2020) and OSTIA sea ice and sea surface temperatures (Donlon et al., 2012). Topographical model domain information was based on 100 m CORINE land use data (Copernicus Land Monitoring Service, n.d.) and global multi-resolution terrain elevation data 2010 (GMTED2010) at 30 arcsec (Danielson and Gesch, 2010). The simulations were kept consistent with the synoptic setting at the driving coarse scales using spectral nudging. Each year–domain configuration spent 80.000 to 140.000 core hours and the full wind atlas thus took about 35 million core hours. This shows that dynamical downscaling remains computationally demanding and time-consuming despite the restricted size of the model domain. It however leads to a drastic reduction of computational costs compared to global domain simulations.

It should be noted that the bridging of mesoscale modelled data to microscale wind resource mapping is obtained via the so-called generalization method, described by Badger et al. (2014). The generalized wind climate approach allows the resulting mesoscale wind climate (mean statistics) to be downscaled even further via microscale models following the wind atlas framework (Troen and Petersen, 1989). Linearized flow models, such as the WAsP model, are simple and computationally efficient and can thus be applied to large areas with limitations in very complex terrain. The goal is capture and account for local flow features resulting from relatively simple sub-grid variations in orography and surface roughness. These effects can be vital for accurately determining the local wind climate at a site and to further assess its suitability with respect to wind farm design.

2.1.2 Statistical downscaling

Statistical downscaling methods use low-resolution simulation (predictor) to infer predictions at a finer scale (predictand), thus avoiding high-resolution simulations. Relationships between the two scales are learned by training statistical models on known predictor-predictand data pairs (Höhlein et al., 2020)(Vandal et al., 2018). Many such empirical approaches have been developed, based on regression methods, weather typing classification and probabilistic weather generators (Wilby and Wigley, 1997). Some of these approaches generally perform well, but realistic reconstruction of spatial variability in high-resolution fields remains a challenge for most models (Maraun, Widmann, and Gutiérrez, 2019). The more advanced methods are usually fine tuned with local observational inputs based on expert knowledge and thus cannot be generalized well across different regions. Recent machine learning approaches, in particular the field of deep learning, can potentially provide more flexible and generalised statistical models for downscaling atmospheric fields.

2.2 Key concepts of deep learning

Deep learning methods are among the leading examples of state-of-the-art machine learning techniques, which refers to statistical methods that automatically improve while solving a certain task by learning from examples. This can be done in a supervised manner, where each example contains a known input and result and the goal is to learn a mapping that minimizes a measure of difference between the model output and targets. During unsupervised learning the models work on their own to discover relations in the data,

guided by a cost function that depends on the data and the network's output.

Most deep learning models are based on artificial neural networks (ANN), which are built from a collection of nodes (so called artificial neurons) that are connected by edges enabling them to receive and pass information. Each incoming signal is processed independently by an adjustable weight and a shared bias term, and the collective sum is passed to an activation function which is usually non-linear. The neurons are typically aggregated into layers. ANNs base their prediction on features that are extracted from the raw input data, and multiple layers enable higher-level feature representations. The classical example of a feed-forward ANN is the fully connected (dense) network where each neuron in an individual layer has connections to the neurons of the subsequent layer, see e.g. Buduma and Locascio (2017). Another type of feed-forward networks are convolutional neural networks (CNNs) that have successfully been applied in various complex image processing and understanding tasks. CNNs are mainly used in this work and are further described in subsection 2.2.2. The passing of information is unidirectional in feed-forward networks, i.e. it does not propagate backwards. Recurrent neural networks, such as the long short term memory (LSTM) (Hochreiter and Schmidhuber, 1997), on the other hand have cyclic connections enabling them to process sequential data such as time series. Graph neural networks (GNNs) (Zhou et al., 2018) are another more recent type of ANN models that capture relationships in graphs via message passing between the nodes of graphs and are suitable with non-structured data.

2.2.1 Optimization

ANN model parameters (weights and biases) are optimized to minimize a given loss function L . The loss in a supervised framework is a measure of error between the predicted outputs and the true outputs. The model training then involves multiple iterations through a training set containing input \mathbf{x} and ground truth target data \mathbf{t} , to approximate a mapping of the form $\mathbf{y} = f(\mathbf{x})$ that minimizes $L(\mathbf{t}, \mathbf{y})$. Model parameters are regularly updated during training after processing a set of data, to yield a progressively better fit to the targets based on gradients of the loss function. This is commonly done using back-propagation, originally proposed by (Rumelhart, Hinton, and Williams, 1986). The weights and biases are firstly initialized with random small values, e.g. sampled from a standard Gaussian distribution. The algorithm then calculates the gradient of the resulting loss function with respect to the model parameters of each weighted layer with chain rule calculus, starting at the last layers and proceeding backwards through the network. Partial computations of the gradient are reused from layer to layer allowing for more efficient computation of the gradient at each layer. The model parameters are then updated using gradient descent. In mathematical terms the updated weights and biases (collectively denoted by a parameter vector θ) are then given by

$$\theta^{n+1} = \theta^n - \alpha \partial_\theta L(\mathbf{t}, \mathbf{y}, \theta^n) \quad (2.1)$$

where α is the learning rate, controlling how much the network parameters are adjusted with respect to the loss gradient. Stochastic estimates of the gradient are obtained by using small batches of training data, effectively avoiding local minima and saddle points of the gradient (Buduma and Locascio, 2017). The process is repeated until a certain number of epochs¹ have been reached, ideally at a point when the loss has converged. A popular extension to the backpropagation algorithm is to add so called momentum to the gradient decent, by considering a fraction $|\gamma| < 1$ of the previous weight update to the

¹One epoch corresponds to a complete pass through the entire data.

current one

$$\mathbf{m}_{n+1} = \gamma \mathbf{m}_n - \alpha \partial_{\theta} L(\mathbf{t}, \mathbf{y}, \theta^n), \quad (2.2)$$

$$\theta_{n+1} = \theta_n + \mathbf{m}_{n+1}. \quad (2.3)$$

This exponentially weighted decay of past gradients effectively leads to bigger but less volatile updates, generally speeding up convergence (Buduma and Locascio, 2017). High learning rates can lead to overshooting, especially when used with momentum, and too low values impedes convergence. Scheduling a time-based, step-based or exponential decrease in the learning rate can help the convergence but requires some tuning. Adaptive learning rate schemes are often used in practise. An prominent example is the Adaptive Moment Estimation (Adam) algorithm (Kingma and Lei Ba, 2015), which has been found to work well for deeper networks and typically requires little tuning. The method computes a specific adaptive learning rate for every parameter in the network from first and second order moment estimates of the gradients using a default learning rate as an upper limit. (Ede and Beanland, 2020) suggest an adaptive learning rate clipping algorithm for stochastic gradient decent (without momentum).

The choice of batch size is somewhat a trade-off between computational time and optimization benefits. Smaller batch size yields more frequent and stochastic parameter updates that can benefit the optimization process but is more computationally demanding (more parameter updates per epoch). Bad batches with difficult examples can further lead to loss spikes causing abrupt model parameter changes that destabilizes the training. Many weight initialization schemes further exist that have been shown to bring substantially faster convergence, such as the Xavier initializer which aims to preserve the variance of input features with each passing layer as well as in the back-propagated signal (Glorot and Y. Bengio, 2010).

2.2.2 Convolutional Neural Networks

CNNs are based on a trainable shared-weight architecture which performs regression tasks with localized filter kernels that slide along input features. This results in a feature map which can in mathematical terms be expressed as

$$m_{ij} = f((\mathbf{w} * \mathbf{x})_{ij} + b) \quad (2.4)$$

where \mathbf{w} is the corresponding weight filter, b is a shared bias term, \mathbf{x} is the input data and $f(\cdot)$ denotes an activation function. The convolutional operation, here denoted by $(*)$, represents the sum of element-wise product between a local subset of the input and the weight kernel, illustrated in Figure 2.1. The output size is effectively determined by the kernel stride in the lateral and vertical directions and input data padding.

$$\begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad x$$

$$\begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \quad w$$

$$\begin{pmatrix} 1 & 4 & 3 & 4 & 1 \\ 1 & 2 & 4 & 3 & 3 \\ 1 & 2 & 3 & 4 & 1 \\ 1 & 3 & 3 & 1 & 1 \\ 3 & 3 & 1 & 1 & 0 \end{pmatrix} \quad x * w$$

Figure 2.1: Illustration of the 2D convolutional operation, based on (Stackoverflow, 2018)

This simple mathematical description however does not fully describe 2D convolutional filters as they are used in practise. More specifically, they operate on entire volume of feature maps to fully process the input. A simplified example is a typical RGB image containing three input features. A typical convolutional filter processes the input with a specific weight matrix for each input channel. In a volumetric sense, these filters are represented by a tensor that has the same depth as the input (see Figure 2.2a). The resulting feature map is then made up by the sum of element-wise multiplication between the 3D tensor and a local 3D subset of the input, with the addition of a shared bias term. Stacked convolution layers (see Figure 2.2b) have multiple feature maps, which all correspond to a specific weight tensor (marked in grey-scale) that processes the input data. For a full description of the convolutional layer, see Buduma and Locascio (2017).

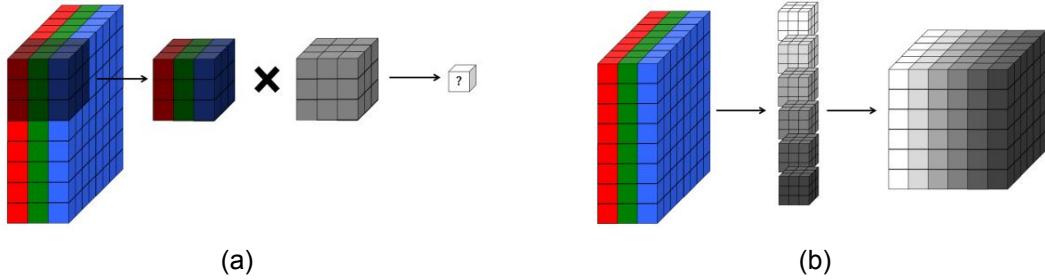


Figure 2.2: (a) Application of a volumetric 2D convolutional filter to a full-color RGB image and (b) a 3D visualization of a convolutional layer, where each convolutional filter corresponds to a slice in the resulting output volume (matched in grey-scale). Both images taken from Buduma and Locascio (2017).

Applying nonlinear activation functions between successive convolution layers enables the model to learn nonlinear mappings. An example of a commonly used activation function is the restricted linear unit (ReLU), having the functional form $f(z) = \max(0, z)$. A leaky ReLU has a small slope a for negative values instead of a flat slope $f(z) = \max(az, z)$, see Figure 2.3.

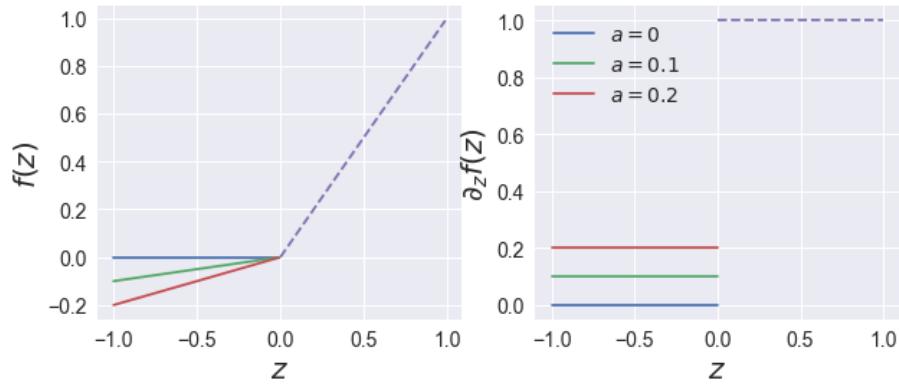


Figure 2.3: Illustration of the ReLU activation function and its derivative.

2.2.3 Design of deep CNNs

Deeper networks are able to model more complex mappings than shallower designs. They are however harder to train, with more iterations generally required for convergence and being more prone to suffer from instabilities related to optimization tasks. In particular due to the problem of vanishing gradients (Glorot and Yoshua Bengio, 2010) that

arises from the build-up of small gradients (influenced by non-linear activation functions) in the layer by layer estimation of model weight updates based on chain-rule calculus. This effectively prevents the weights from significantly changing their values, especially in the early layers of the network, slowing down and possibly even completely stopping the learning progress. Batch normalization (Ioffe and Szegedy, 2015), i.e. normalizing inputs to every layer in the neural network, has been found to increase the stability during training, but vanishing gradients still remain an inherent problem of deep neural network architectures.

An effective approach to address this problem are so called skip-connections, that directly pass the output of earlier layers to a later stage in the network while bypassing one or more layers. This effectively reduces the dependency of intermediate layers in parameter updates and thus circumvents the build-up of small gradients. The deep residual learning framework, proposed by (He et al., 2016a), is a successful example that uses such connections. In stead of making the stacked weighed layers, one after each other, directly fit a underlying mapping $H(\mathbf{x})$ they suggest these layers to fit a residual mapping $F(\mathbf{x}) := H(\mathbf{x}) - \mathbf{x}$, such that the features before and after intermediate processing are added together (see Figure 2.4a). The weighted layers then learn whatever is different between the input and output, i.e. the residuals. Weighted layers are generally well-suited to learn functions close to zero (like residual functions), as they are initialized with small random numbers and due to weight regularisation methods that are commonly applied. This framework thus enables the model to more directly learn mappings that are close to identity, through the entire network or within residual units.

Another type of skip-connections, commonly associated with a so-called UNet architecture (Ronneberger, Fischer, and Brox, 2015) concatenate the output of a previous layer with the result of an intermediate layer. The U-Net architecture (see Figure 2.4b) consists of two symmetric branches, connected by skip connections. Firstly an branch that encodes the data into an reduced latent feature representation, and secondly a branch that decodes these feature representations to match the expected output.

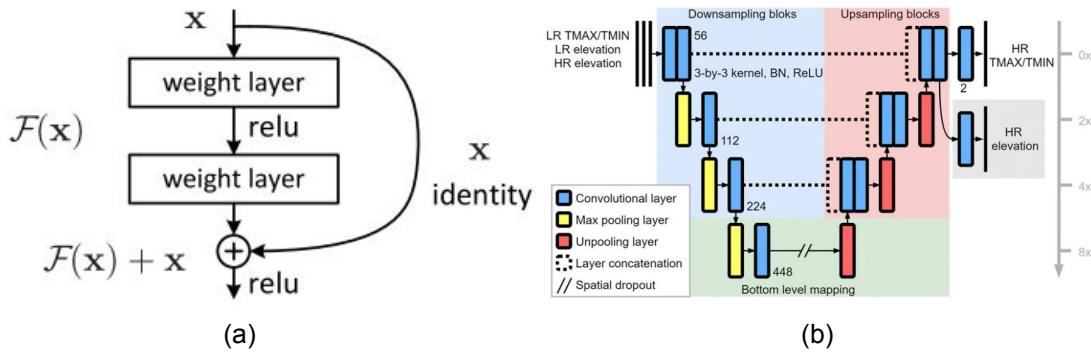


Figure 2.4: (a) The residual learning building block He et al., 2016a and (b) a schematic of the UNet architecture from (Sha et al., 2020).

2.2.4 Hyperparameter tuning

There are quite a few control parameters and choices that are involved in the creation of any machine learning model, in particular for neural networks. Some of these so called hyper-parameters directly affect the learning process of the model, such as the learning rate, batch size, and number of training epochs. Other hyper-parameters affect the actual topology and structure of the model, such as the number of weighted layers, the size of kernels and their strides, and the choice of weight regularization and initialization methods

and activation functions. Hyper-parameters are usually fixed and do not change during training. The configuration and tuning of hyper-parameters is usually done by applying a grid search. A value for each hyper-parameter is then picked from a finite set of options and the model is trained with every possible permutation of hyper-parameter choices. The combination of hyper-parameters with the best performance on unseen data is finally selected (Buduma and Locascio, 2017).

2.2.5 Overfitting and regularization

Overfitting is a problem commonly encountered for more complex machine learning models, especially in the case of data paucity. ANN can in that regard quickly overfit to a small dataset, achieving very good performance on the training data but performing poorly with unseen data. To evaluate if the model is starting to show signs of overfitting during training, one usually tests it simultaneously on an independent set of data, socalled validation set with known targets. Overfitting is then observed if the training loss keeps decreasing while the validation loss increases. Many approaches exist that strive to avoid overfitting and increase the model generalizing capability, ultimately striking a balance between variance and bias related errors (Herlau, Schmidt, and Mørup, 2020), such as dropout and weight decay (Buduma and Locascio, 2017). Another approach is to implement early stopping before the model starts to fit to undesirable noise in the training data. This is done at the risk of not reaching a proper state of learning (i.e. underfitting and thus higher bias error). Data augmentation further increase diversity in training examples that effectively reduces the chance of overfitting.

2.3 Generative Adversarial Networks

The generative adversarial network (GAN) training strategy (Goodfellow et al., 2014) is based on a game between two competing machine learning models (usually CNNs for image processing), a generator G and a discriminator D . These networks are trained simultaneously in an adversarial manner, i.e. against each other, with the following objective

$$\min_G \max_D \mathbb{E}[\log(D(\mathbf{t}))] + \mathbb{E}[\log(1 - D(\mathbf{y}))] \quad (2.5)$$

where the discriminator output represents the probability of the sample being real ($D \in [0, 1]$) and $\mathbf{y} = G(\mathbf{x})$. The discriminator thus tries to distinguish between real and generated samples and the generator at the same time strives to output samples that the discriminator considers real. Minimizing this value function implicitly minimizes a distance between the target data distribution and the generative model output distribution. The training procedure for this original formulation of GANs is however known to suffer from vanishing gradients in early stages of learning as the discriminator saturates. Goodfellow et al. (2014) suggest training the generator to minimize $-\mathbb{E}[\log(D(\mathbf{y}))]$ in practise for better gradients, having the same optimum.

Arjovsky, Chintala, and Bottou (2017) propose an alternative Wasserstein GAN (WGAN) objective function with better theoretical properties

$$\min_G \max_D \mathbb{E}[D(\mathbf{t})] - \mathbb{E}[D(\mathbf{y})]. \quad (2.6)$$

The discriminator output is then unbounded but limited to a subspace of well behaving functions (1-Lipschitz condition), playing a role of a critic since it is not classifying the output. The gradients of the resulting critic function with respect to its input are better behaved making optimization of the generator easier. However to enforce these properties the authors use weight clipping which has problematic aspects. Using a softer version of these constraints Gulrajani et al. (2017) propose an alternative WGAN training scheme

using a gradient penalty term. The resulting objective, adopting the convention in neural network training to minimize a loss function, is constructed as

$$L = \mathbb{E}[D(\mathbf{y})] - \mathbb{E}[D(\mathbf{t})] + \lambda \mathbb{E}[(\|\partial_{\hat{\mathbf{y}}} D(\hat{\mathbf{y}})\| - 1)^2] \quad (2.7)$$

with $\hat{\mathbf{y}}$ sampled uniformly along straight lines between pairs of points from the target data \mathbf{t} and generator output $\mathbf{y} = G(\mathbf{x})$. The Wasserstein loss can be understood intuitively as the discriminator trying to make its output as large as possible for real samples and as small as possible for generated samples.

2.4 Image super-resolution

Being a illposed and underdetermined inverse problem, there are many possible HR solutions from a given LR input. This makes SR an extremely difficult task, especially for high scaling factors. Image SR algorithms can roughly be divided into interpolation-based methods, reconstruction-based methods and learning-based methods (W. Yang et al., 2019). Interpolation-based methods include bilinear or bicubic interpolation of the LR pixels. These methods do not require any training data but tend to significantly smooth out the image, loosing out on small-scale details (C. Y. Yang, Ma, and M. H. Yang, 2014). Reconstruction-based methods use sophisticated prior knowledge on primitive image properties such as edges and segments to reduce the possible solution space and inject sharp details. The performance however generally degrades rapidly with increasing scale factor and these methods are usually time-consuming (W. Yang et al., 2019). More powerful and efficient learning based algorithms utilize machine learning algorithms to analyze statistical relationships and establish complex mappings between the LR and HR information from training data. Powerful data-driven deep learning methods, in particular convolutional neural networks (CNNs), have recently achieved state-of-the-art performance when applied to SISR. A detailed overview of such methods for SR is given in (W. Yang et al., 2019).

2.4.1 A brief overview of deep learning based SR

The super-resolution CNN (SRCNN) by Dong, Loy, He, et al. (2014) was one of the first CNN-based architecture for SISR, used as foundation for the DeepSD model (Höhlein et al., 2020) shown in Figure 1.2. Bicubic interpolation is used to upscale the LR input image, which is then further processed by three convolution layers with rectified-linear (ReLU) activation functions in between. This original SRCNN architecture is problematic in terms of computational speed as it operates on a HR interpolant of the LR input image. This leads to more computational operations and requires larger convolution kernel sizes to address spatial correlations in HR space, leading to a large number of trainable parameters causing even further computational overhead. A Fast SRCNN (FSRCNN) design (Dong, Loy, and Tang, 2016), also shown in Figure 1.2, circumvents these problems by applying convolution layers to the LR inputs directly and upsampling features to the target resolution at the final stage of the model. The network then learns the upscaling filters directly, replacing the bicubic interpolation with a transpose convolution associated with nearest neighbour interpolation or relevant data padding, which increases the performance in terms of accuracy and speed.

The performance for SR tasks generally improves as the network complexity grows (W. Yang et al., 2019). An example of a more complex deep SR CNN architecture with residual connections is the EnhanceNet by Sajjadi, Schölkopf, and Hirsch (2017). UNets (Ronneberger, Fischer, and Brox, 2015) have also recently been applied to SR problems (Hu et al., 2019), forming the basis of the DeepRU network for wind field SR (Höhlein et al., 2020). Both networks are shown in Figure 1.2. The UNet design however requires the

input and output to be of the same size, involving the need of interpolating the input fields to match the target dimension.

Deep CNNs that have been optimized to solely minimize pixelwise losses, such as the mean squared error (MSE), fail to match the fidelity expected in the HR reconstruction, especially for larger scaling factors. The behavior of such optimization-based SR methods is in fact principally governed by the choice of the objective function (Ledig et al., 2017). Minimizing MSE effectively promotes a pixel-wise average reconstruction of plausible solutions that have high peak signal to noise ratios but are typically overly smooth and have poor perceptual quality (W. Yang et al., 2019). Recent work using deep CNNs for SR has thus focused on finding image quality metrics, so called perceptual losses, that can serve as additional terms in the loss function to produce more realistic images. The loss is then shifted from image-space to a higher-level feature space based on intermediate layers of pretrained auxillary networks, see e.g. Johnson, Alahi, and Fei-Fei (2016).

Generative adversarial networks (GANs) have recently been used to train CNNs for SR tasks, producing realistic super-resolved images from LR input images. GANs developed for SR tasks are conditioned on a low-resolution image with the goal of producing a high-resolution version of it. The discriminator then evaluates the quality of the reconstruction, while the generator learns to generate more realistic samples. The adversarial training effectively drives the reconstruction towards a more visually convincing manifold without necessarily relying on user-based and domain specific loss functions for increased perceptual quality. The SRGAN (Ledig et al., 2017) is a seminal work that is capable of generating photo-realistic natural images for 4 \times upscaling factors. To further enhance the visual quality, X. Wang et al. (2018) thoroughly studied the network design and loss function of the SRGAN and imposed several improvements to derive an state-of-the-art Enhanced SRGAN (ESRGAN). In the original formulation of GANs, the fake data is constructed by first sampling a random vector that is transformed to structured data via the generator. This stochastic nature of GANs is generally not considered in super-resolution applications, with advanced SR GAN architectures adopting a completely deterministic approach.

3 Data and methodology

3.1 Data

Post-processed mesoscale WRF simulations from the NEWA database (Dörenkämper and Hahmann, 2019)(NEWA, n.d.), covering the Central Europe (CE) and Great Britain (GB) domains, are considered in this study for model training and evaluation. The selected domains are shown in Figure 3.3. See Dörenkämper, Olsen, et al. (2020) for an overview of the final wind atlas parameters. Relevant post-processing steps involve the conversion of (terrain following) hybrid sigma levels to meters above ground, interpolation to 7 wind energy relevant height levels, destaggering of the raw WRF wind field grid, rotation of the raw WRF wind components relative to the Earth's surface and the calculation of the meteorological wind direction ϕ (in degrees) and wind speed magnitude U from the rotated wind components. The half-hourly WRF simulated NEWA data is sampled every three (full) hours over the 12 year period 2007-2018. Wind fields (at a given time) containing invalid data are completely discarded.

The networks are trained to operate on the zonal and meridional wind components u and v from the 100 m vertical level. They are calculated as follows

$$u = -U \sin\left(\phi \frac{\pi}{180}\right) \quad (3.1)$$

$$v = -U \cos\left(\phi \frac{\pi}{180}\right). \quad (3.2)$$

The post processed NEWA wind components are taken as ground truth training and evaluation targets. LR input to the models is created by applying an averaging filter to the target data. Namely by using a 10x10 kernel with weights $w_{ij} = 1/n^2 = 10^{-2}$ and calculating the product sum while sliding through each input field separately with a stride of (10, 10), i.e. not a conventional 2D convolution but feature wise average pooling. The resulting coarsened fields roughly match the resolution of the ERA5 dynamic forcing data (~ 30 km). A flowchart of the training data generation is shown in Figure 3.1. High resolution static ground elevation data (HGT) and land-sea-mask (LSM) from the WRF geogrid processing system are also utilized.

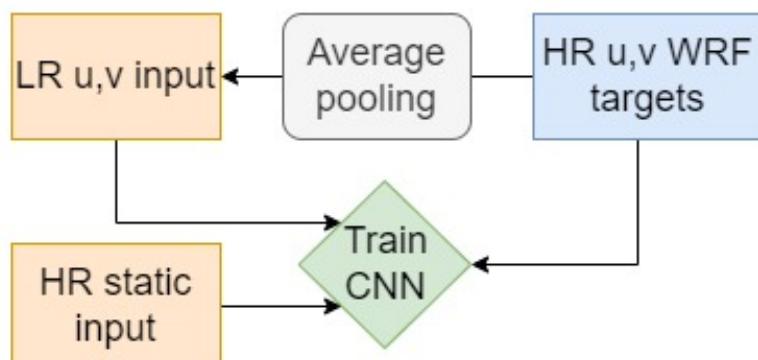


Figure 3.1: Workflow of training data generation and model training framework.

The CE NEWA domain is exclusively used for training the models. The training split is chosen as the period 2011-2016, while the period 2007-2010 is reserved for testing. The

years 2017 and 2018 are used for validation and fine tuning of the model architecture respectively. Annual global domain statistics of the simulated wind speed magnitude from the CE model domain data are illustrated in Figure 3.2. Mean statistics over the full NEWA period and the 12 year study period are shown for reference. A significant trend in these statistics is not observed during the study period. It is however slightly calmer and has less variance on average than the full 30 year dataset. Relatively high and low wind speed years are within the test set while the training set contains four near average wind years and two years of relatively high and low wind speeds. Similar observations can be noted for the annual variance. The train and test splits are believed to give a reasonable representations of inter-annual, seasonal and diurnal variations observed in the NEWA data.

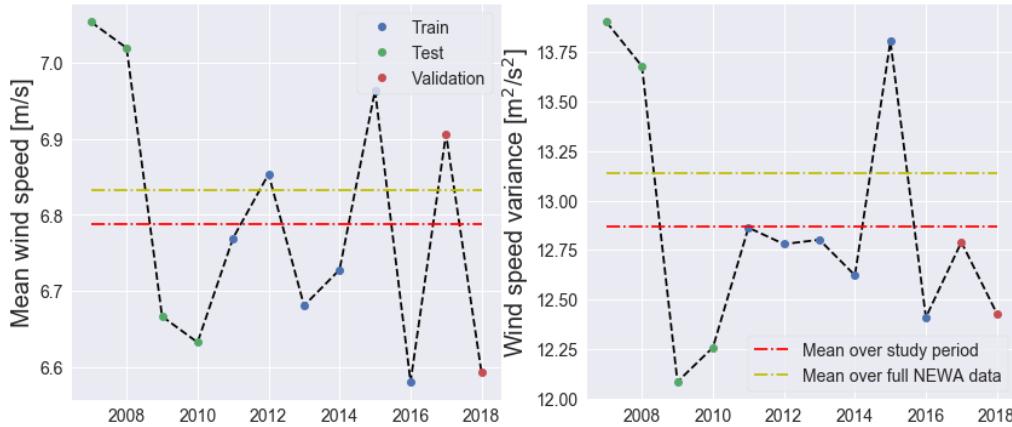


Figure 3.2: Annual global domain statistics based on CE domain NEWA data (100 m above ground) for the period 2007-2018. Mean statistics over the full NEWA period and the 12 year study period are shown for reference with dash dotted lines.

3.1.1 Scaling

Before training the wind velocity data, both inputs and targets, is standardized by subtracting the sample mean and dividing by sample standard deviation

$$\tilde{\mathbf{x}} = \frac{\mathbf{x} - \mu}{\sigma}, \quad \tilde{\mathbf{t}} = \frac{\mathbf{t} - \mu}{\sigma}. \quad (3.3)$$

These sample statistics are estimated from snapshot computations of the HR targets over the entire domain and training period. The standardization is not based on test set targets to avoid leakage of information that can effect the performance evaluation. The static HGT data is also standardized by global domain statistics. Standardizing the LSM is not considered relevant due to its binary nature. Standardization has proven useful in machine learning and other optimization based methods (Ioffe and Szegedy, 2015)(Buduma and Locascio, 2017). It makes the model predictions invariant to different scales of input fields which can dominate initial weight updates through gradient calculations.

3.1.2 Domain description

The CE domain (Figure 3.3a) is centered at 51.04°N and 13.47°E and is composed of 414 grid points in North-South (1242 km) and 540 grid points in the West-East direction (1620 km). To complement the $10 \times$ SR factor, the corresponding data fields are slightly reduced by excluding the most northerly and southerly grid points (four North-South rows of data). The domain contains sub-regions with a broad variety of topographical features, posing a challenge for the considered models. Most notable orographic properties are the

Alps and the Carpathian mountain ranges, and smaller mountainous regions in Central Europe. The domain further covers flat agricultural areas and coastal regions of the Baltic and North Sea. Complex topography related low-level wind systems and phenomena are known to develop in the region such as the Böhmischer Wind, Höllentäler, Wisperwind and the Alpine Fohn winds (Deutscher Wetterdienst, 2021). They are however unlikely to be well represented in NEWA data due to limitations of mesoscale weather models. The near coastal regions in the German Bight and Baltic sea are further home to complex flow systems that develop in connection to the large surface roughness contrast at the coast over which large temperature gradient generally exist (Dörenkämper, Optis, et al., 2015). The training set wind climatology, shown in Figure 3.4, seems to be highly influenced by prominent topographic features in the domain, with higher mean wind speed offshore, a sharp contrast at the coastline and local onshore variations associated with orography.

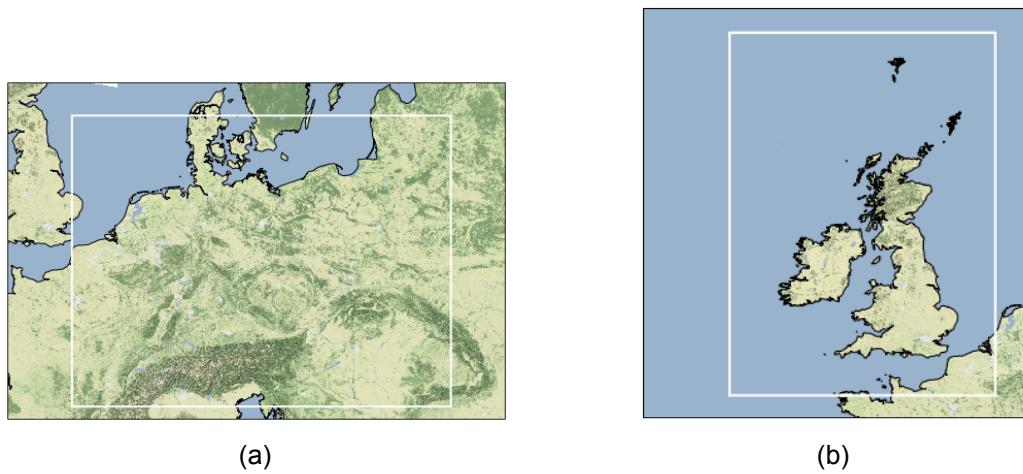


Figure 3.3: (a) The Central Europe (CE) and (b) the Great Britain (GB) NEWA model domains. The background is the stamen terrain background from <http://maps.stamen.com/terrain-background> (last access: 9 May 2022) – © OpenStreetMap contributors 2020.

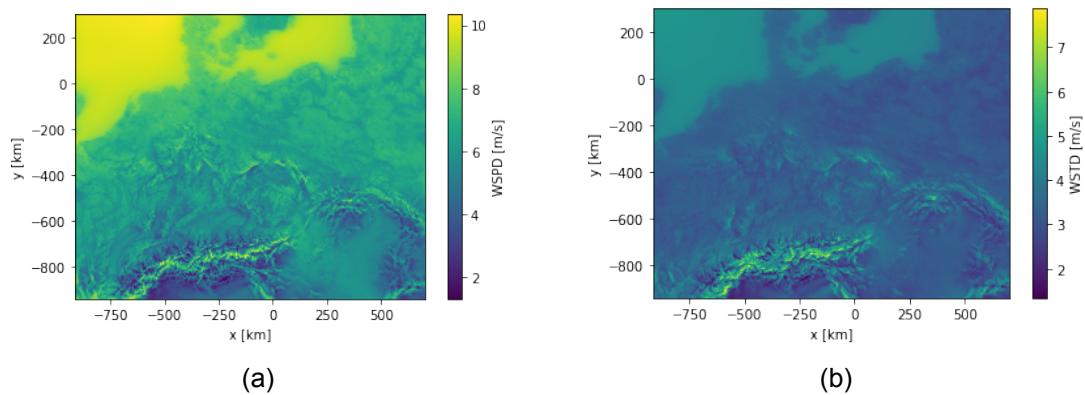


Figure 3.4: Average annual (a) mean wind speed magnitude and (b) standard deviation at 100 m above ground from the CE NEWA training set (2011-2016).

The GB domain (Figure 3.3b) is centered at 55.92°N and 6.36°W and spans 540 grid points in North-South (1242 km) and 396 in the West-East direction (1188 km). Data fields are reduced to 390 West-East columns to match the trained $10 \times \text{SR}$. The domain covers more homogeneous terrain with a higher portion of offshore conditions than the CE

domain. Among notable orographic features within the GB domain are the Scottish highlands and mountain ranges in Wales. Figure 3.4 represents the wind climatology over the training set period for the GB domain. Higher mean wind speed is gradually observed offshore and onshore variations are largely associated with orographic features.

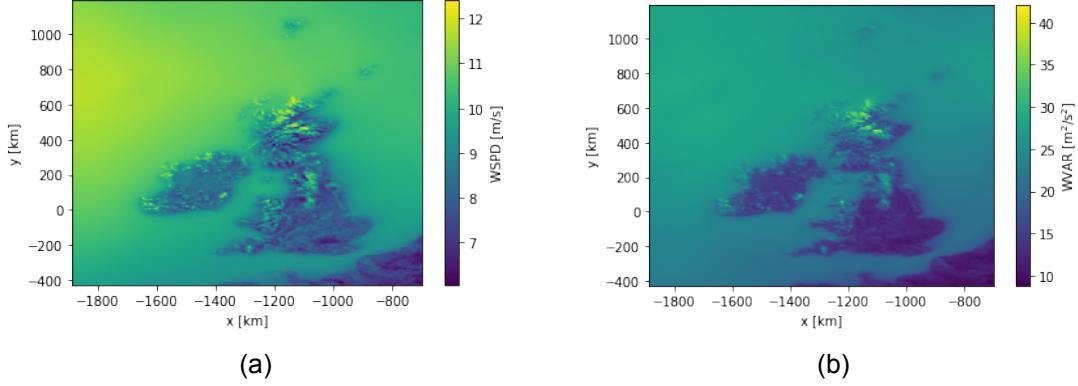


Figure 3.5: Average annual (a) mean wind speed magnitude and (b) variance at 100 m above ground from the GB NEWA domain data covering the training period (2011-2016).

3.2 Model architecture

The model architecture is based on the Super Resolution Generative Adversarial Network (SRGAN) (Ledig et al., 2017), see Figure 3.6, with several modifications. Stengel et al. (2020) adjusted the network input layers to consist of two data channels corresponding to either (u, v) wind velocity components, in contrast to the three RGB channels typically used for image processing. They further uncovered the need to remove batch normalization layers from the architecture. It should however be noted that Höhlein et al. (2020) did not observe performance issues when using batch normalization for their CNN architectures.

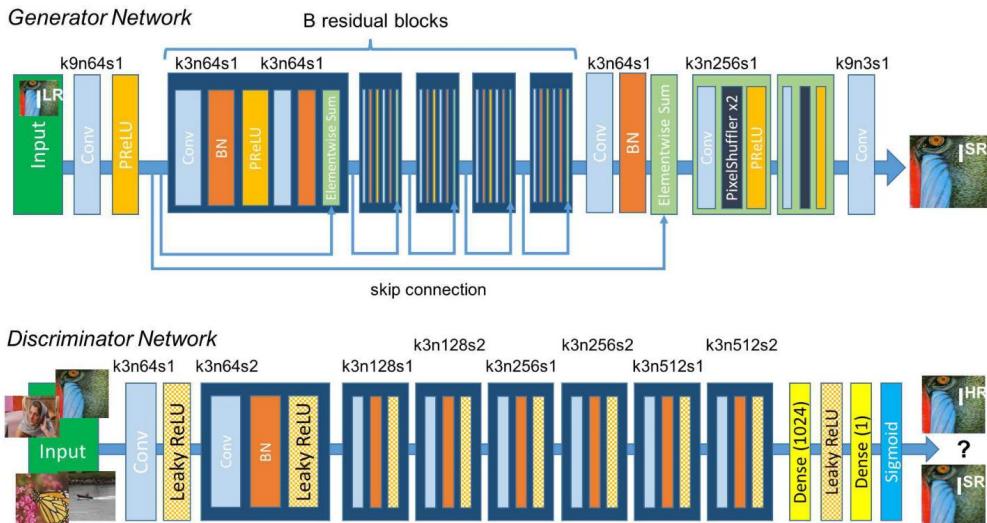


Figure 3.6: Architecture of the original Generator and Discriminator network with corresponding kernel size (k), number of feature maps (n) and stride (s) indicated for each convolutional layer. Taken from Ledig et al. (2017).

3.2.1 Generator

The generator is a deep fully convolutional neural network with 16 residual blocks (He et al., 2016a) and skip connections (He et al., 2016b). The LR input is first processed with a single convolution creating 64 input features that are passed to the residual blocks. After the residual blocks there are two SR layers which upscale the input features. The number of LR features are first increased by a single convolution layer followed by a subpixel convolution¹ (Shi et al., 2016) which shift these feature channels into the spatial domain. The resulting features are then passed to an output convolution. All convolutional kernels are 3×3 and are followed by ReLU activation functions. Stengel et al. (2020) adapted the SR layers to accommodate the larger-resolution jumps being performed, in steps determined by the prime factorization of the SR factor. Their generator network, hereinafter referred to as CNN1, roughly has 2.3 million trainable parameters.

Further modifications to the network design are needed to allow for the ingestion of HR static data. Höhlein et al. (2020) use an input branch equipped with a sequence of strided convolutions that successively decrease the resolution of the HR input features to match the LR dynamic input fields. A generator CNN1 variant is here considered that uses a similar HR input branch with four strided convolutional layers. The first two layers, equipped with 3×3 filter kernels, reduce the dimension by a factor of two in the vertical and lateral dimension respectively. The following two layers, with 5×5 filter kernels, have a stride of 5 in the vertical and lateral directions respectively. These successive convolutions, with leaky ReLU activation function ($a = 0.2$ in between), thus give a $10 \times$ downsampling factor in total. The resulting features are concatenated to the output of the original LR input convolutional layer (limited to 32 features opposed to the original 64). The 64 concatenated input features are then passed to the main part of the model, without further modifications of the network structure. This approach thus allows the model to benefit from HR information in a computationally efficient way without drastic surgery on the main model components. It is however limited to LR feature representations of the HR static data.

Another approach that does not effectively downsample the HR static information is to ingest it after the SR stage in the model. This requires some further processing of the output features at the target resolution which involves more computational operations (compared to the processing of LR features) that decreases the efficiency of the network. A second generator design (CNN2) processes the HR static information separately with three convolutional layers with 16 feature maps that are concatenated to the output features of the SR stage of the model (reduced to 16 features by a single convolution). The 32 concatenated features are passed to a single convolution reducing the feature maps to 16 which are then further processed by four residual blocks before finally being passed to an output convolution. All kernels in these additional layers are 3×3 , keeping the number of trainable parameters comparable to the original design. Extending the network design still adds computational overhead to the optimization process as more gradient calculations are involved in the back-propagation algorithm with the added model layers.

The third option (CNN3) combines the two aforementioned approaches, i.e. using both an HR input branch and adding the HR information again to the model after the SR stage. In the absence of HR static data, CNN3 is simply the same model as CNN2. The described modifications to the original network are illustrated in Figure 3.7 and the resulting increase in the number of trainable parameters is given in Table 3.1. A set of 3×3 weights for each of the feature maps in the first convolutional layer (32 features with a HR input branch but 64 without it) are further added to the number of trainable parameters per additional LR

¹also called depth-to-space

input fields other than the u , v components.

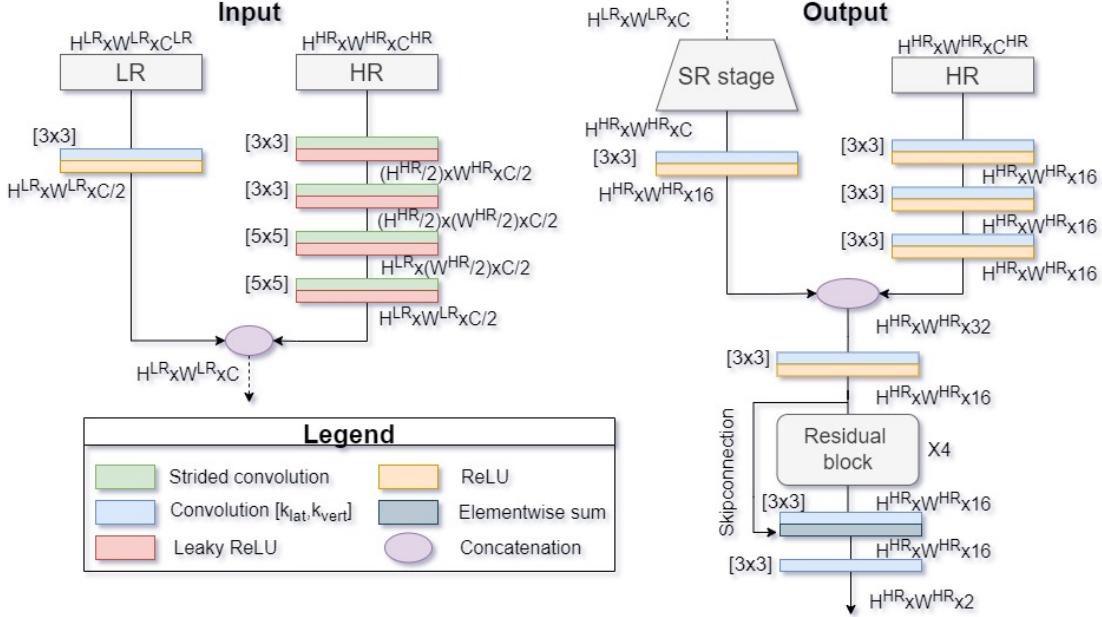


Figure 3.7: Network modifications related to the ingestion of HR static data. The HR input branch is used with CNN1 and CNN3 and the output extension is used for CNN2 and CNN3. Convolutional layers are described by (H, W, C) which represents the number of pixels in vertical and lateral dimensions and the number of feature channels respectively. The final model training is based on $C = 64$ and $C^{LR} = C^{HR} = 2$.

Table 3.1: Increase in the number of trainable model parameters of the generator network CNN variants, given as percentage relative to the Stengel et al. (2020) design (CNN1 without HR static data).

Model	Without HR static data	With HR static data
CNN1	0 %	1.8 %
CNN2	1.3 %	3.4 %
CNN3	1.3 %	5.2 %

3.2.2 Critic

The critic network (discriminator in the original design) contains eight convolutional layers with an increasing number of 3×3 filter kernels, increasing by a factor of 2 from 64 to 256 kernel tensors (feature maps). Strided convolutions are used to reduce the image resolution each time the number of features is doubled. The resulting 256 feature maps are finally passed through two dense layers with 1024 and 1 hidden nodes. Leaky ReLU activation ($a = 0.2$) is used between all layers. A final sigmoid activation function to obtain a probability for sample classification has been removed from the original architecture to accommodate the WGAN training scheme. The number of trainable parameters in the convolutional layers are 1.2 million, and the two dense layers have 78.6 million parameters. The number of parameters in the discriminator network is thus an order of magnitude greater than the generator network size.

3.2.3 Baseline model

Bicubic interpolation of the LR coarsened wind fields onto the HR grid is considered as a traditional statistical baseline model. The bicubic interpolation uses a nearest neighbour prediction for points in the edges of the domain, i.e. no spatial extrapolation. The use of more elaborate radial basis functions (RBF) did not seem to significantly improve the reconstruction while taking more time than bicubic interpolation.

3.3 Training

3.3.1 Patch training

Since the networks are fully convolutional, they are not fixed to a particular size of input data. This enables training on smaller patches of the data, reducing the memory needed to load in the data and allowing for a larger batch-size. Larger fields can further be used in deployment.

Höhlein et al. (2020) found that training their CNN models on sub-patches of the full domain yielded lower training and test losses compared to models trained on full domains. Patch-training increases diversity and variance of the training samples. It is further believed to improve the usage of static predictor information in comparison to full domain training. During full domain training the static fields remain invariant such that they can be ignored by the model or be used by the model to base its predictions purely on local patterns rather than regressing on the provided dynamic information. Intermediate patches were found to give the best results, while too small patches hindered the models to effectively learn spatial correlations in the data.

Patch-training is similarly used in this study. Sub-patches of size (24,32) in North-South and West-East dimension respectively are selected randomly for each daily LR input data file. The corresponding HR targets are synchronized to match the enhanced predictor patch of size (240,320). The induced randomness further decreases the chance of overfitting to the training input. Patch-training likely reduces the tendency of the models to solely pick up on features associated with the highly varying winds over mountainous regions in the domain during training. Thus it is expected to increase the generalizability of the model to unseen domains. Note that patching is only applied during training and that predictions are based on the full domain for evaluation of model performance.

3.3.2 Loss functions

The GAN networks are trained as Wasserstein GANs (WGAN) with gradient penalty λ (Gulrajani et al., 2017). The conditional losses for the critic network (still referred as D for discriminator) is constructed from the WGAN objective in Equation 2.7 giving

$$L_D = D(\mathbf{y}) - D(\mathbf{t}) + \lambda(\|\partial_{\hat{\mathbf{y}}}D(\hat{\mathbf{y}})\| - 1)^2 \quad (3.4)$$

where $\mathbf{y} = G(\mathbf{x})$ and

$$\hat{\mathbf{y}} = \epsilon \mathbf{t} + (1 - \epsilon) \mathbf{y} \quad (3.5)$$

with ϵ sampled randomly from the uniform distribution between 0 and 1. The loss function of the generator is taken as

$$L_G = L_{content} + \beta L_{adversarial} \quad (3.6)$$

with the adversarial loss being defined by the objective in Equation 2.6

$$L_{adversarial} = -D(\mathbf{y}). \quad (3.7)$$

The content loss is taken as spatially averaged pixelwise mean square error (MSE)

$$L_{content} = \left\langle \|\mathbf{t} - \mathbf{y}\|^2 \right\rangle_{Dom} \quad (3.8)$$

where $\|\cdot\|$ is the L_2 vector norm and $\langle \cdot \rangle_{Dom}$ indicates an average over the spatial domain. This is a widely used optimization target for image SR. All losses are averaged over the batch before each model update.

3.3.3 Implementation and optimization

The software implementation is largely based on python scripts (Stengel et al., 2021), developed for Stengel et al. (2020), involving the base model architecture and training configuration. Main modifications to this code repository involve the network changes related to the ingestion of HR static data and the WGAN training framework and losses. The networks are implemented using the end-to-end TensorFlow platform (Abadi et al., 2016) (version 1.12) and their training was conducted on GPU-accelerated nodes on the ForWind HPC cluster Eddy. The cluster is equipped with three NVIDIA Tesla P100 PCI-Edition GPUs with 16GB memory. During training, data for all models are processed in batches of 50 to 128 samples, depending on the model complexity and memory requirements during the optimization. The number of training epochs are given in Table 3.2.

Table 3.2: Batch size and number of epochs during training

Model	CNN1	CNN2	WGAN
Batch size	128	80	50
Number of Epochs	400	250	600

The networks are trained using the Adam optimizer (Kingma and Lei Ba, 2015) with a learning rate of $\alpha = 10^{-4}$ and model weights are initialized with Xavier normal initializer (Glorot and Y. Bengio, 2010). Following the WGAN training algorithm (Gulrajani et al., 2017), shown in section C.1, the generator weights are updated every fifth batch while the discriminator is updated every iteration with a gradient penalty of $\lambda = 10$. The generator network is trained with an adversarial loss component scaling of $\beta = 0.001$ in case of a WGAN and $\beta = 0$ for a regular CNN (i.e. without the critic network).

3.4 Generator ablation study

To investigate the importance and behaviour of the various generator model components related to the ingestion of static topographical data, a simple ablation study is performed using the validation set. The CNN networks are trained for 200 epochs with a batch size of 64. Hourly fields are sampled from the 2018 simulation period and used for training. The performance is evaluated on unseen data sampled every 6h from the year 2017. The importance of features and model components is evaluated based on accuracy (loss) on the test data, training behaviour and computational aspects.

The first generator design is trained with no static input data, both HR and LR static data and only HR static data. The second generator design (CNN2) is trained with and without static data. The evolution of MSE loss on the standardized training set and the normalized training time per epoch are shown in Figure 3.8. The training curves for CNN3, considering both HR and LR data and only LR static data, were almost identical to CNN2 and are thus not shown in the figure. The networks have not fully converged after 200 epoch but meaningful improvements affecting this simple comparison are not expected with further

training. These training metrics show important aspects of the different generator designs. The original CNN architecture (CNN1 without static data) is more computationally efficient than the extended CNN2 design which took 70 % more time to train per epoch on average. This is expected as processing images in the target HR resolution involves more operations which increase the computational costs. Back propagation of gradient calculations are furthermore more complicated with the extended architecture, which likely explains the more noisy training loss curves and slower learning. Further adding HR static information to the model output processing state involves more trainable parameters and more computations in HR space leading to an additional 24 % increase in training time on average (112 % longer compared to the original design). This however results in considerable lower training loss. Adding an HR input branch to the original design slightly increases the number of trainable model parameters which leads to a modest increase in training time (about 8 % on average), seemingly without compromising the stability of the optimization. The inclusion of LR static input fields has negligible effect on the training time as expected.

The performance on the unseen validation data, shown in Table 3.3 indicates that the generator benefits considerably by receiving HR topographic data. It is however interesting to note that extending the network design with further feature processing layers does not seem to increase the predictive performance in the absence of HR static data. CNN2 slightly outperforms the original design with a HR input branch. Adding an HR input branch to the extended design does not improve its performance. This indicates that the topographic related information that is extracted by the HR input branch is fully represented by the later processing of the HR data.

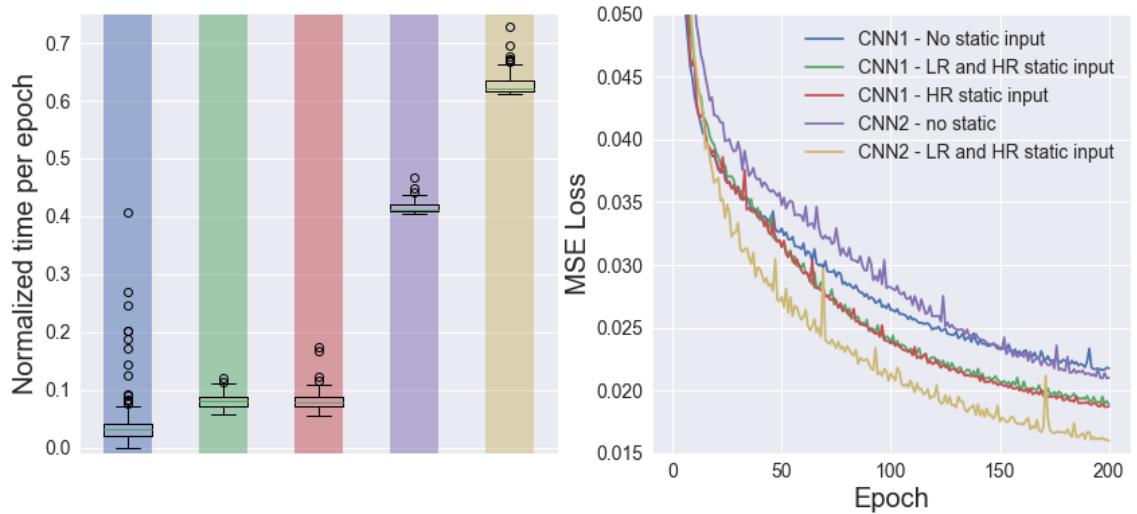


Figure 3.8: Statistics of normalized time per epoch during training and the evolution of MSE loss on the standardized training set. The maximum training time for CNN2 (used for the normalization) is beyond the scale of the plot.

Table 3.3: MSE score on unseen validation data along with relative improvement compared to the original design.

	CNN1			CNN2		CNN3	
	-	HR,LR	HR	-	HR,LR	HR	HR,LR
MSE [m ² /s ²]	1.82	1.57	1.52	1.82	1.20	1.18	1.20
Improvement [%]	0	-13.7	-16.5	-0.2	-34.1	-35.2	-34.2

3.4.1 Summary

The HR input branch is much more efficient than CNN2, but the latter achieves lower reconstruction errors measured by domain averaged MSE. Both models are thus included in the final model training. Combining both approaches in a CNN3 design adds computational overhead with little to none predictive improvement compared to CNN2, and is thus not further considered in this study. The WGAN is only trained with the CNN1 generator variant, as using CNN2 is not considered feasible in a more complex training framework. Further analysis of the predictive performance of these two CNN models is needed to determine what information is lost when using an HR input branch, which relies on LR feature representations of the HR static data, compared to processing it at the target resolution. A trade-off between predictive skill and computational time is expected. The network variants do not seem to base their predictions on coarsened static data in the presence of HR static data. They are thus deemed irrelevant for further model analysis.

3.5 Model evaluation

Both GB and CE NEWA domains are used to evaluate the model performance. The test set is standardized with sample statistics from the train set (CE domain) before processing and the model output is converted back to real scale for evaluation purposes. The spatial generalization ability is evaluated by testing the models on the GB domain which was completely omitted during training.

3.5.1 Quantitative analysis

The model output for two individual wind fields from the test set, one for each domain, are chosen for a quantitative analysis of the model performance based on a visual perspective. The chosen date is the 1st January 2007 at 21 UTC. A synoptic weather chart from the retrospective Deutscher Wetterdienst (DWD) analysis (Deutscher Wetterdienst, n.d.), covering Europe and valid three hours before the simulated wind fields, is shown in appendix A. Statistics of global domain MSE (Equation 3.8) on the entire test set, for each domain separately, are also calculated as a first base indicator of the model performance.

3.5.2 Spatial distribution of errors

The spatial distribution of reconstruction errors in both domains is examined by measures of angular and magnitude deviations, averaged over the test set for each individual grid point. The cosine dissimilarity (CosDis)

$$\text{CosDis} = \frac{1}{2} (1 - \langle \cos(\mathbf{t}_i, \mathbf{y}_i) \rangle) = \frac{1}{2} \left(1 - \left\langle \frac{\mathbf{t}_i \cdot \mathbf{y}_i}{\|\mathbf{t}_i\| \|\mathbf{y}_i\|} \right\rangle \right) \quad (3.9)$$

is considered for angular deviations between target and model predictions. Magnitude difference (MD)

$$\text{MD} = \langle \|\mathbf{t}_i\| - \|\mathbf{y}_i\| \rangle \quad (3.10)$$

is used to measure systematic model bias in terms of wind speed magnitude. In both measures \mathbf{t}_i , \mathbf{y}_i denote the target and predicted wind vectors at grid point i in the model domain, $\|\cdot\|$ is the L_2 vector norm and $\langle \cdot \rangle$ indicates an average over the test set.

3.5.3 Spectral analysis of wind fields

The ability of the models to recover and resolve various physical scales is evaluated by comparing the spectrum of the model output to the spectra of target data and theoretical $-5/3$ considerations. The spectral analysis is based on discrete Fourier transform that is applied to the wind speed magnitude data fields and resolves them into their underlying simple sinusoidal functions covering all possible spatial scales with wavenumbers

$$\begin{aligned} k_1 &= \frac{2\pi}{\Delta x i}, & i &= 2, \dots, H/2 \\ k_2 &= \frac{2\pi}{\Delta x j}, & j &= 2, \dots, W/2 \end{aligned} \quad (3.11)$$

where $\Delta x = 3000$ m is the grid resolution and W, H number of grid points in lateral and vertical dimensions respectively. The power spectral density (PSD) is then calculated as

$$S(k) = \frac{\Delta x}{2\pi WH} |X(k)|^2 \quad (3.12)$$

with $k = \sqrt{k_1^2 + k_2^2}$ and $X(k)$ being radially averaged spectrum in k -space. The resulting spectral density is finally log-bin smoothed. The example wind field target from the CE domain, chosen for the quantitative analysis, is used to evaluate and compare the ability of the various model variants to produce small scale variance in their predictions as measured by the spectral density of their output fields.

3.5.4 Climatology and long-term distributions

The probability distribution of wind speed magnitude at a given location in space plays a key role in wind resource assessment to give an accurate estimation of mean annual energy production. The Weibull distribution, characterized with coefficients A and k , is often used in that regard². Weibull fits to the generated wind data are here obtained using the method of moments. It can be shown that the n -th statistical moment of a Weibull-distributed variable is given by (Kelly, 2020)

$$\mu_n = A^n \Gamma(1 + \frac{n}{k}) \quad (3.13)$$

where the gamma function $\Gamma(x)$ is defined as

$$\Gamma(x) = \int_0^\infty t^{x-1} \exp(-t) dt. \quad (3.14)$$

Taking the ratio of the third moment and the cube of the mean wind speed gives

$$\frac{\mu_3}{\mu^3} = \frac{\Gamma(1 + 3/k)}{\Gamma(1 + 1/k)} \quad (3.15)$$

which can be solved iteratively for k based on the sample moments. By plugging the solution for k into the analytical expression of either the mean or third moment the corresponding solution for A is obtained. Other variations of this fitting method exist, employing different reformulations of the analytical expressions of moments. This approach is

²Weibull generally gives a good fit to the distribution of observed 10 min averaged wind speed (Kelly, 2020)

deemed relevant for wind energy purposes due to the dependency on wind speed cubed which is directly related to power.

The mean and standard deviations for the entire test set model output are calculated to give an idea of how well the wind climatology can be represented in the model reconstruction. Further analysis is based on three grid points situated in different terrain types, shown in Figure 3.9. One offshore, in the German Bight, and two onshore grid points, in flat agricultural area in the Netherlands and hilly terrain of central Germany respectively. The location of the selected sites is given in Table 3.4. Time series for these points are extracted from the model output and the binwise wind speed magnitude frequency distribution estimated and compared to target data, along with all sector Weibull fitting.

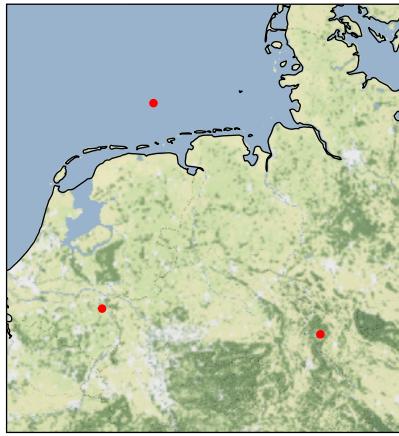


Figure 3.9: Topographic map showing the considered site locations in CE domain

Table 3.4: Latitude and longitude coordinates of the considered sites.

Site	Lat	Lon	SI°E
Offshore	53.97	6.34	
Onshore simple	51.71	5.83	
Onshore complex	51.65	9.68	

4 Results

4.1 Final model training

The final model training, described in section 3.3, took 2 – 3 days for the CNNs and around 5 days for the WGAN. The evolution of MSE loss on the standardized training and validation set are shown in section C.2. Saturation of validation losses was usually achieved by 200 epochs for the CNN1 model, at values around 0.024 and 0.02 with and without a HR input branch respectively, and by 100 epochs for CNN2 at a value of around 0.018. Marginal improvements ($\sim 10^{-5}$) were obtained in the training loss beyond these points. Training the CNN1 model further than 350 epochs lead to high error spikes with significant weight updates that destabilized training, see Figure C.1 and Figure C.2. The weights after 350 epochs of training are thus used for the final model evaluation. Tendencies of overfitting are not observed with further training once the models converged. The WGAN models exhibit higher MSE on the standardized training and validation set than the CNN models. Their validation losses saturate after around 500 epochs for both generator and critic networks. Their final validation MSE loss is around 0.035 and 0.025, with lower loss obtained for the topographically aware model. The critic validation loss converges at 0 while a progressive increase in the training loss is observed after around 100 training epochs.

4.2 Quantitative analysis

4.2.1 Generated wind fields

Figure 4.1 and Figure 4.2 contain results from the 10 x SR applied to an unseen example from the test set. The figures show LR and HR wind velocities along with the bicubic baseline, CNN1 and WGAN model output. The resulting wind fields for the CE domain are shown in Figure 4.1. Strong westerly flow with sharp gradients is observed in the northern part of the CE domain, related to the frontal passage of a rather deep low-pressure system centered over Scandinavia. Local flow features associated with topography, such as speed-up effects and highly varying winds, are observed in the southern and central part of the domain. The LR rendering of the HR target image preserves large-scale flow features and some smaller scale details related to most prominent topographical features. The interpolation based SR is not able to recreate lost high frequency details and yields overly smooth data while the CNN based SR fields are more realistic, with sharper gradients and contrasts and introducing small scale features. The WGAN generates data with very sharp gradients and even more refined small-scale features.

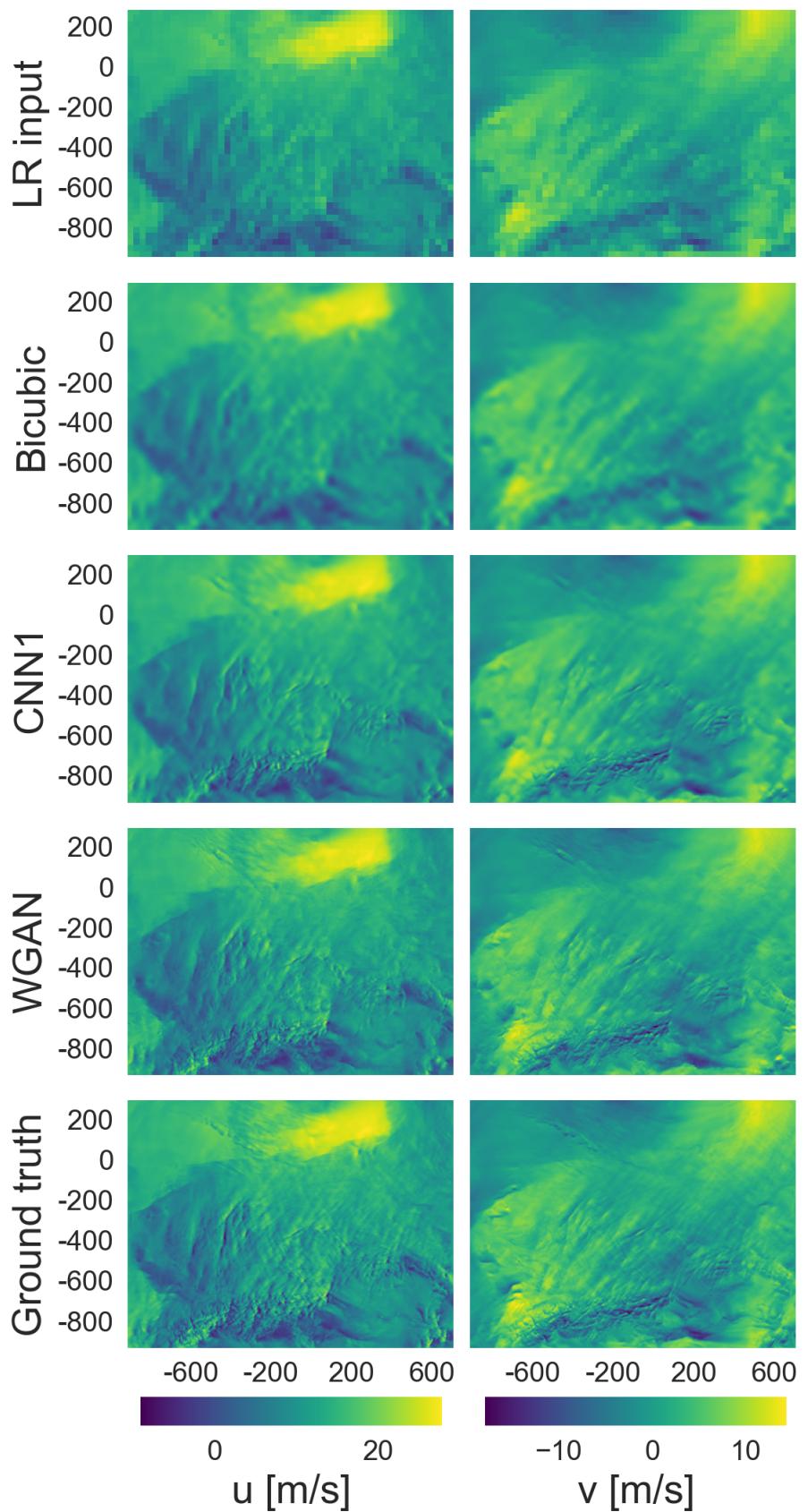


Figure 4.1: Comparison of various SR methods on NEWA CE domain data fields.

The corresponding wind field data for the GB domain is shown in Figure 4.2. The flow field is influenced by a small low-pressure system located in the north-western part of the domain. Similar remarks regarding the various model output quality can be made as for the CE domain example. There is however notable discrepancy between the model output and the target data in the vicinity of the low-pressure system. The WGAN generates the most detailed and convincing flow field with more realistic structures than the CNN. The injected small scale features are however in many cases not accurately placed with respect to the target data.

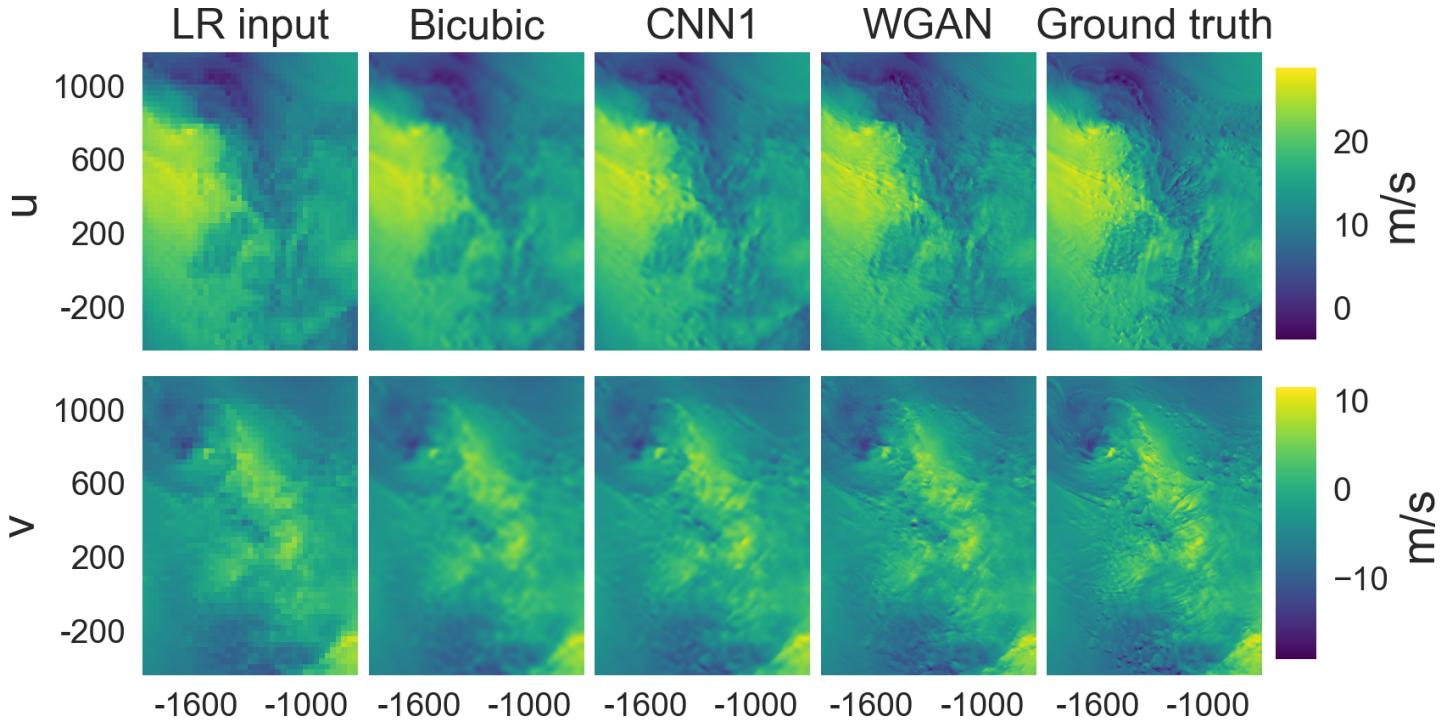


Figure 4.2: Comparison of various SR methods on NEWA GB domain data fields.

4.2.2 Domain error metrics

Statistics of spatially averaged MSE on the test set are shown in Figure 4.3 for the trained CNN and WGAN models along with a bicubic baseline. The errors from the CE domain, shown on the left image, are generally in the range of $0.8\text{--}1.4 \text{ m}^2/\text{s}^2$. Descriptive statistics are shown in Table 4.2. Positive skewness naturally arises from the definition of MSE, which is bounded by zero, with minimum reconstruction errors ranging from $0.2\text{--}0.4 \text{ m}^2/\text{s}^2$ and maximum reconstruction errors reaching $3.7\text{--}6.7 \text{ m}^2/\text{s}^2$. The distributions never the less have a strong central tendency. The bicubic baseline has the greatest spread of error and highest maximum error. The WGAN generator network has slightly less spread in MSE errors, with lower extreme errors and quartiles. Leveraging HR static data reduces the spread of errors to some extent, lowering the maximum error by $1 \text{ m}^2/\text{s}^2$ and reaching performance akin to the original CNN model (without static HR data). The CNN models have the lowest MSE on average. Maximum errors are drastically lowered by the use of HR static data. Using the HR input branch reduces the maximum error by roughly $0.6 \text{ m}^2/\text{s}^2$ and processing it at the target resolution (CNN2) reduces it by $1.1 \text{ m}^2/\text{s}^2$. The spread of error is also reduced by using HR static data and the quartiles shifted towards lower values.

The errors on the GB test set, illustrated on the right in Figure 4.3 and summarized in Table 4.3, are generally in the range of $0.4 - 1.2 \text{ m}^2/\text{s}^2$ with minimum errors around $0.2 - 0.3 \text{ m}^2/\text{s}^2$ and reaching $2.9 - 4.4 \text{ m}^2/\text{s}^2$ in worst cases. The distributions are more positively skewed, i.e. more clustering towards the left tail, than observed for the CE domain. The WGAN model has the highest spread of error and by far the most extreme errors. The distribution of errors is slightly shifted towards lower values by incorporating HR static data in the design. Despite the improvements involved with HR static data, the WGAN model still has roughly $1 \text{ m}^2/\text{s}^2$ higher maximum error and larger spread than the baseline. The CNN models have slightly lower errors than the bicubic baseline on average. The use of HR static data does not drastically reduce their errors, with all CNN variants having similar maximum and minimum errors.

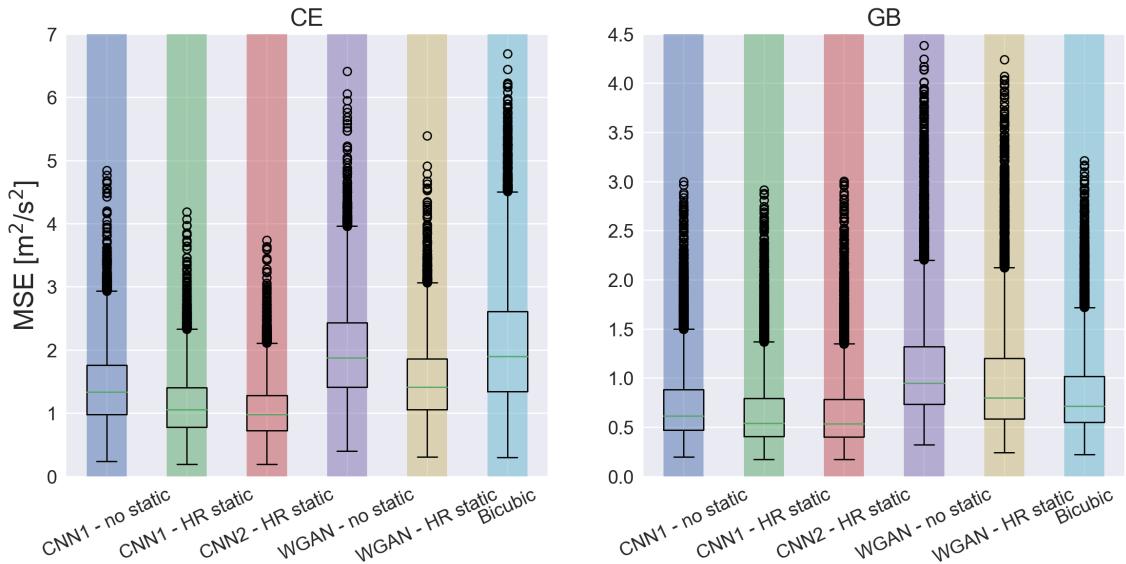


Figure 4.3: Statistics of spatially averaged MSE on the test set, for the CE NEWA domain (left) and the GB domain (right).

Table 4.1: Descriptive statistics of domain averaged MSE for the considered models on the CE domain.

Statistic [m^2/s^2]	Bicubic	CNN1		CNN2		WGAN	
		no static	HR static	HR static	no static	HR static	
mean	2.05	1.41	1.12	1.03	1.96	1.50	
std	0.97	0.60	0.48	0.43	0.75	0.61	
min	0.30	0.24	0.19	0.19	0.40	0.31	
25%	1.34	0.98	0.78	0.73	1.41	1.06	
50%	1.90	1.34	1.05	0.98	1.88	1.41	
75%	2.61	1.76	1.40	1.28	2.43	1.86	
max	6.69	4.84	4.19	3.74	6.41	5.39	

Table 4.2: Descriptive statistics of domain averaged MSE for the considered models on the CE domain.

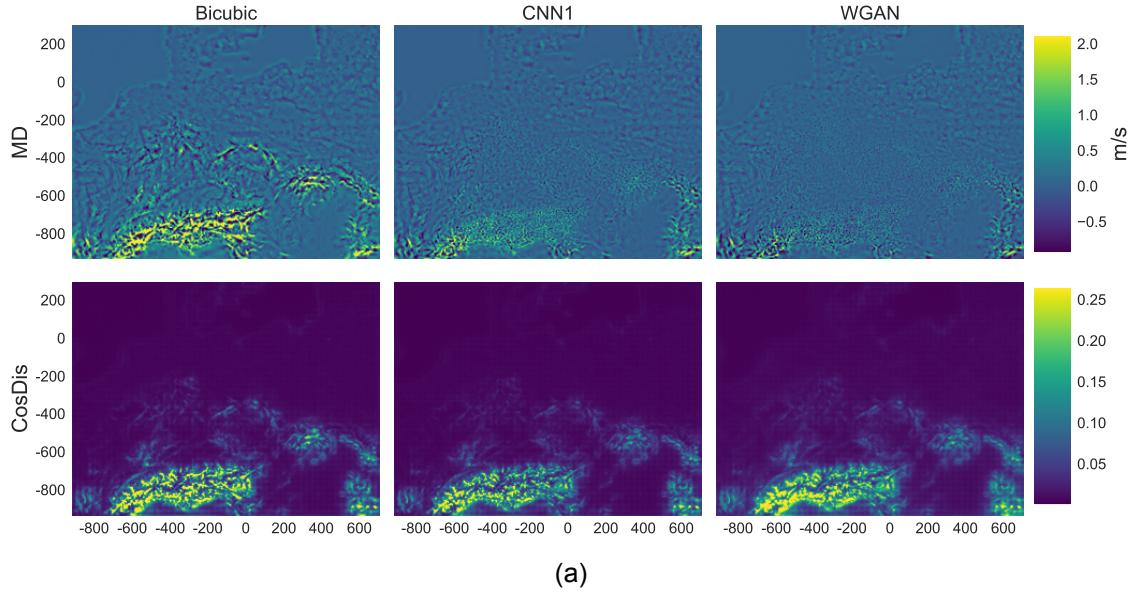
Statistic [m ² /s ²]	Bicubic	CNN1		CNN2		WGAN	
		no static	HR static	HR static	no static	HR static	
mean	0.84	0.73	0.66	0.65	1.11	0.98	
std	0.42	0.38	0.37	0.37	0.55	0.56	
min	0.22	0.20	0.17	0.17	0.32	0.24	
25%	0.55	0.47	0.41	0.40	0.74	0.59	
50%	0.72	0.62	0.54	0.54	0.95	0.80	
75%	1.02	0.88	0.79	0.78	1.32	1.20	
max	3.21	3.00	2.91	3.00	4.38	4.24	

Table 4.3: Descriptive statistics of domain averaged MSE for the considered models on the CE domain.

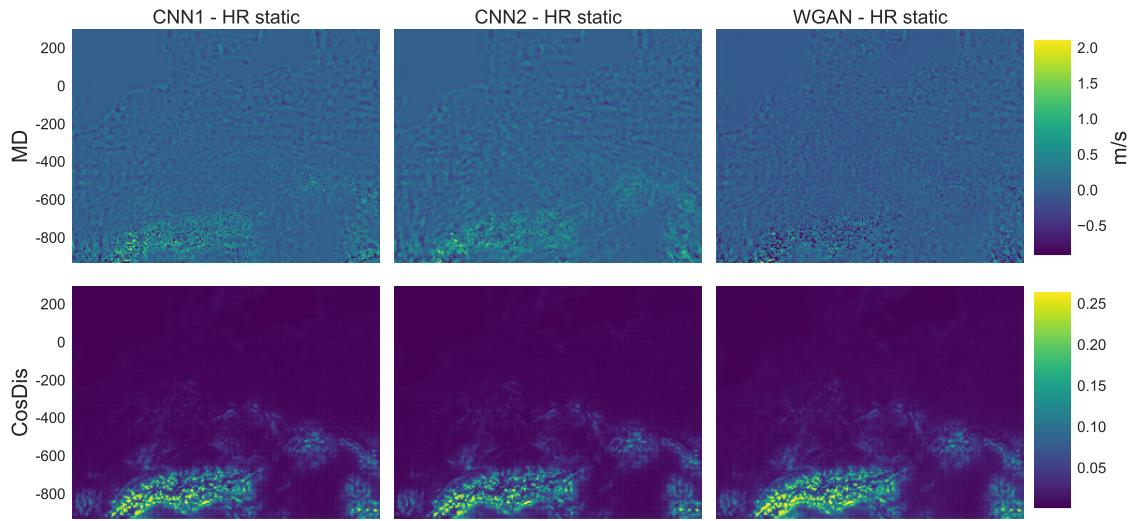
4.3 Spatial distribution of prediction errors

Figure 4.4 depicts average MD and CosDis error metrics, defined in subsection 3.5.2, that show the spatial distribution of systematic deviations in wind magnitude and direction respectively on the CE NEWA domain. Figure 4.4a shows the performance of bicubic baseline, CNN1 and WGAN models. The results show systematic angular and magnitude deviations in connection to localized and regional orographic features. Magnitude errors are largely described by systematic underpredictions on top of hills and mountains and overpredictions in valleys. More prominent errors are observed in the Alpine region with MD up to 2 m/s on average and where the average CosDis error reaches 0.25 , which corresponds to 60°. The land-sea discontinuity is also associated with systematic magnitude errors, with overpredictions and underpredictions going from land to sea, while a near zero bias is observed further offshore with little to none localized variations. The CNN1 and WGAN models have considerably less wind magnitude bias in complex terrain, specifically in the Alpine region, compared to the bicubic interpolator and a lower error contrast at the coast on average. WGAN seemingly has more prominent angular deviations in the Alpine area compared to the other two models. Figure 4.4b similarly shows the performance of CNN1, CNN2 and WGAN when leveraging HR static data in their predictions. Systematic underpredictions in wind magnitude and angular deviations are notably reduced in the Alpine region by adding topographic information. CNN1 has more localized patterns in magnitude error than the other two models, while the WGAN has less underpredictions in wind magnitude but a higher CosDis score compare to the CNNs.

Figure 4.5 similarly depicts average MD and CosDis error metrics on the GB NEWA domain. The bicubic, CNN1 and WGAN model performance is shown in Figure 4.4a. Systematic underpredictions in wind speed magnitude reaching 0.8 m/s are observed over orographic features in the domain, and over- and underpredictions are observed at the land-sea transition for all models. Near zero bias is found offshore. Angular deviations are observed onshore, specifically in more complex terrain where the models yield CosDis values up to 0.02-0.025 on average which corresponds to 50-60°. The WGAN model seems to have higher CosDis values onshore in general. Figure 4.4b illustrates the same metrics for CNN1, CNN2 and WGAN leveraging HR static information. Terrain related underpredictions are drastically reduced and slightly less bias is observed at the coast. Magnitude errors are generally marginally lower for CNN2 compared to CNN1, while the



(a)



(b)

Figure 4.4: Mean magnitude difference (top row) and mean cosine deviations (bottom row) between target high-resolution NEWA CE domain data and prediction of a) CNN1, WGAN and bicubic baseline models without HR static data and b) CNN1, CNN2 and WGAN with HR static data. The average is taken over all four years in the test set.

WGAN has notably lower errors onshore and at the coastline but more variations are observed offshore close to the coast. CosDis errors are also reduced to some extent, but similar maximum deviations are observed on average. WGAN seems to suffer more from prominent directional errors onshore than the CNN models.

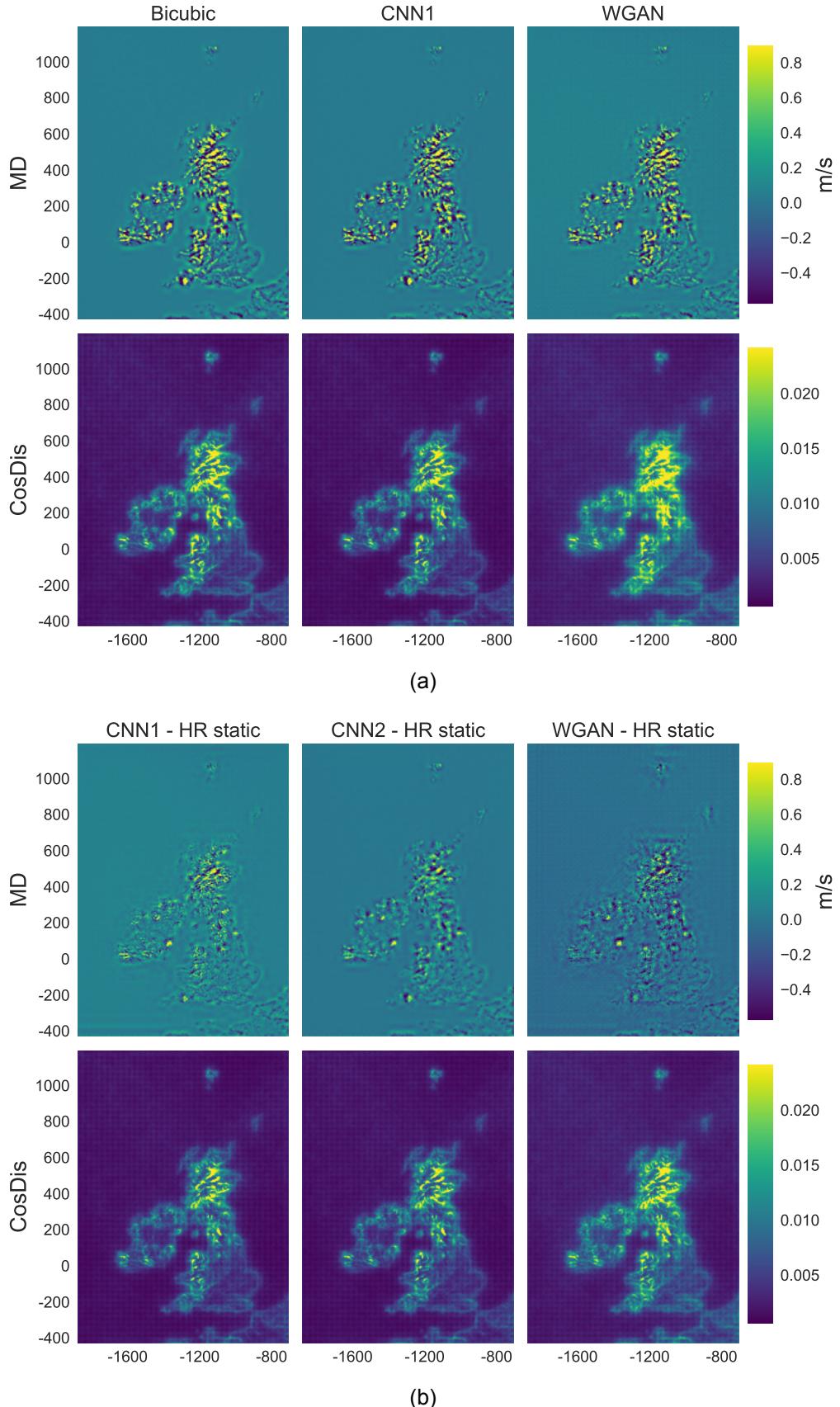


Figure 4.5: Mean magnitude difference (top row) and mean cosine deviations (bottom row) between target high-resolution NEWA GB domain data and prediction of a) CNN1, WGAN and bicubic baseline models without HR static data and b) CNN1, CNN2 and WGAN with HR static data. The average is taken over all four years in the test set.

Downscaling wind fields using deep learning

4.4 Wind climatology

Providing a proxy for wind energy related climatology, the mean and standard deviation of wind magnitude for target data and topography informed CNN1 and WGAN model output on the CE test set are shown in Figure 4.6. Both models provide a decent representation of the target climatology in terms of both magnitude and variations in the resulting fields, reproducing very sharp orographic related patterns. They however lack the high fidelity observed at the coast, in terms of both measures. Similar remarks can be made on the climatological performance on the GB domain, shown in Figure 4.7. The WGAN exhibits sharper patterns in the Scottish highlands and more acute contrast at the coast compared to the CNN.

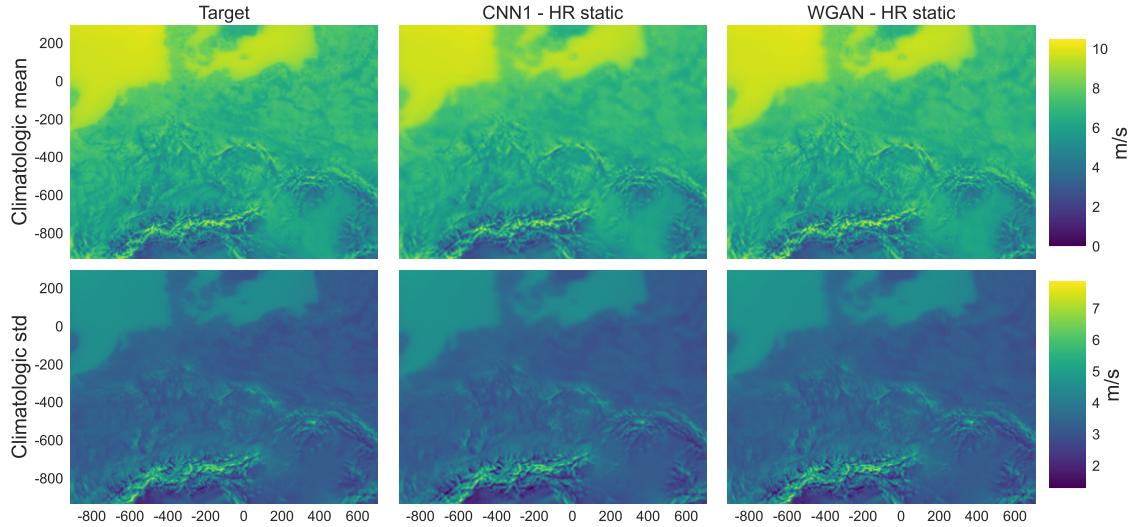


Figure 4.6: Climatological mean and standard deviation of wind magnitude from the CE test set. Results shown for target data and topographically aware CNN1 and WGAN model output.

4.4.1 Wind speed distribution

Figure 4.8 portrays binwise sample probability density of the test set at three given locations in the CE domain that are situated in different terrain. Solid lines depict Weibull fits to the sample distribution and topographically aware CNN1 and WGAN model output. Binwise difference between the normalised frequency of target and model output data are further shown for comparison. Small and seemingly random deviations are observed for the offshore and simple onshore sites with very similar Weibull fits that give a good representation of the data. Numerical values of she sample average and fitted Weibull coefficients along with the relative difference to the target data fit are given in Table 4.4. Less than 2% absolute difference is obtained. The Weibull fits at the onshore site however slightly misplace the mode, with the peak occurring at 1 m/s lower wind speed than indicated by the sample distributions. More deviations are observed in the case of a more topographically complex onshore site, with the models generally underrepresenting low wind speeds and overrepresenting in the range of 5 – 13 m/s compared to the target data distribution with density deviations up to ± 0.02 s/m. The corresponding Weibull fits are shifted towards higher wind speeds compared to the fit derived from target data. Numerical values of the Weibull parameters show around 10 % difference relative to the target data fit, with similar relative difference in the sample mean. To further inspect the nature of these deviations, a wind rose from the target and WGAN model output data is shown in section A.2 on Figure A.2. The wind rose also gives a rough indication of how wind speed

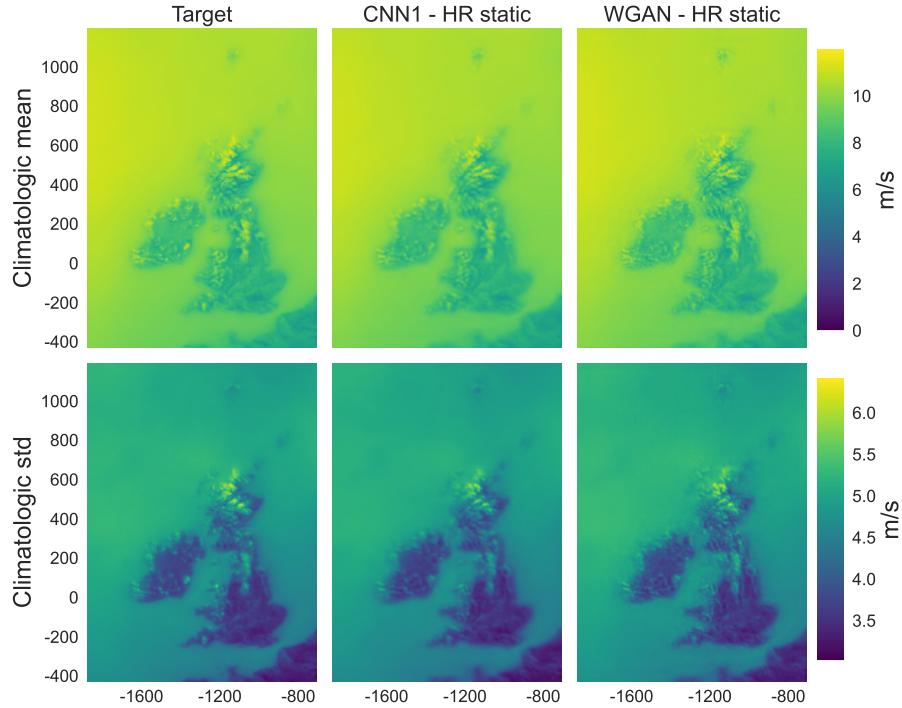


Figure 4.7: Climatological mean and standard deviation of wind magnitude from the GB test set. Results shown for target data and topographically aware CNN1 and WGAN model output.

magnitude is distributed within each 30° sector. Westerly winds are most prominent at the site and there is less representation of low wind speeds in the westerly bin for the model output compared to target data.

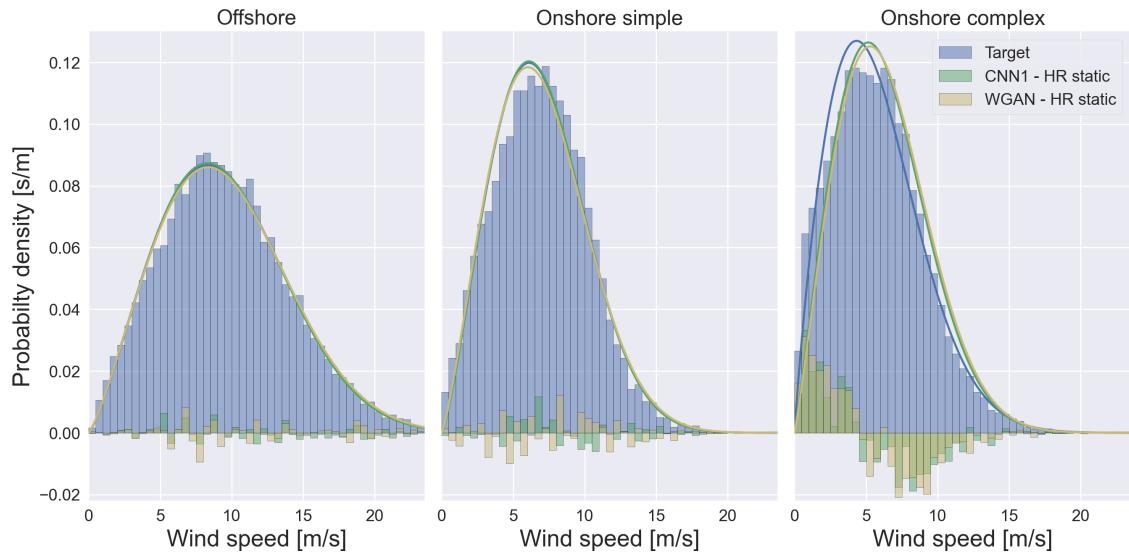


Figure 4.8: Normalized histogram showing the distribution of wind speed magnitude at three different locations in the CE domain covering the entire test set period. Weibull fits to the target data (blue) and model output (yellow and green) are shown with solid lines. Bin-wise difference in frequency between target and modelled data is also illustrated.

Table 4.4: First moment and fitted Weibull parameters derived from the target data and topographic aware CNN1 and WGAN model output at three sites from the CE test set. Relative difference between target and model output shown for comparison.

Site	Parameter	Target	CNN1 HR static	WGAN HR static
Offshore	μ [m/s]	9.6	9.5	9.6
	k	2.3	2.3	2.3
	A [m/s]	10.8	10.7	10.8
	$\Delta\mu$ [%]	-	-0.6	0.3
	Δk [%]	-	0.1	-0.6
	ΔA [%]	-	-0.6	0.3
Onshore Simple	μ	7.0	6.9	7.0
	k	2.3	2.3	2.2
	A [m/s]	7.9	7.8	7.9
	$\Delta\mu$ [%]	-	-0.3	-0.2
	Δk [%]	-	0.2	-1.8
	ΔA [%]	-	-0.3	-0.2
Onshore Complex	μ [m/s]	5.8	6.2	6.3
	k	1.9	2.1	2.1
	A [m/s]	6.5	7.0	7.1
	$\Delta\mu$ [%]	-	7.6	9.6
	Δk [%]	-	11.7	13.0
	ΔA [%]	-	7.9	9.9

4.5 Spectral analysis

Figure 4.9 shows radially averaged spectrum in phase space, described in subsection 3.5.3, for the example wind fields shown in Figure 4.1 along with topographically aware model variant outputs. The WRF model spectrum (target in red) exhibits a strong $-5/3$ decay until $k \approx 2e-4$, corresponding to scales of ~ 30 km which is 10x the grid resolution, where a steeper negative slope is gradually obtained. The bicubic baseline diverges from the target spectrum at scales around $k = 3e-5$ and decays rapidly showing much less spectral density at higher wavenumbers (smaller scales) compared to the HR target data. The CNN model output deviates from ground truth slightly before $k = 1e-4$. The spectral density of CNN1 without static HR data decays relatively faster than the topographically aware CNN variants. The input branch variant (CNN1 - HR static) reaches similar spectral density as CNN1 at smaller scales. The extended CNN2 variant maintains a spectral density comparable to the WGAN towards smaller scales but eventually diverges and reaches the same spectrum as the other two CNNs. The topographically aware WGAN model follows the target spectrum near perfectly with slight deviations at smallest possible scales, while the WGAN without HR static data slightly lacks spectral density below $k = 1e-4$.

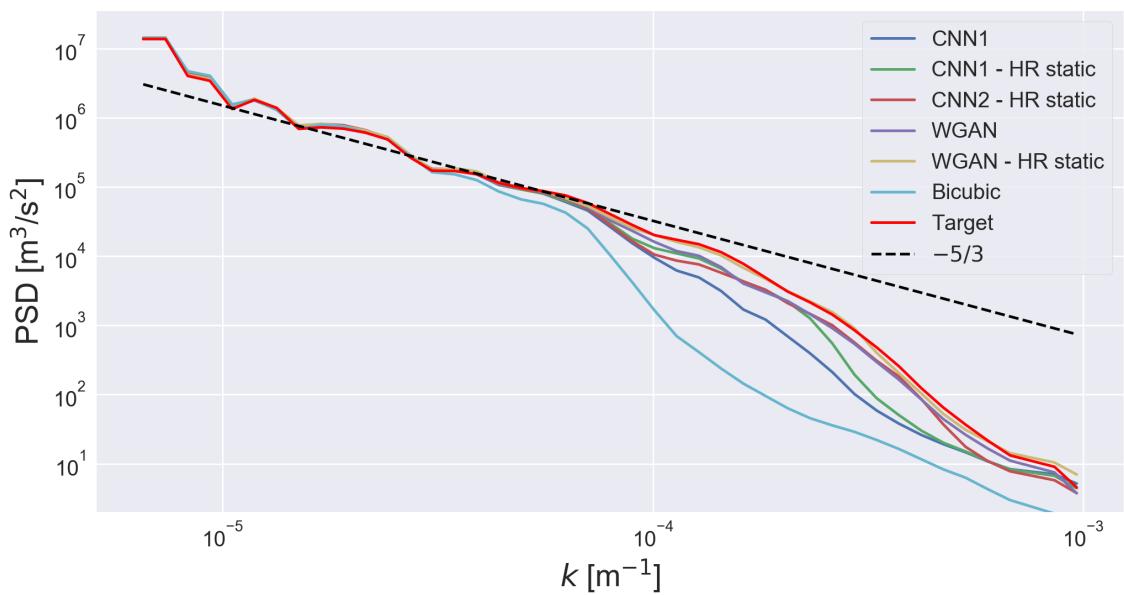


Figure 4.9: Radially averaged 2D spectrum. Dashed line depicts the expected $-5/3$ spectral decay in turbulent mesoscale flow.

5 Discussion

The results are now discussed in connection to the research objectives stated in section 1.3. Higher MSE losses are obtained for the GAN model output as they are not solely optimized to minimize MSE. Misplacement of fine scale details are likely penalized by MSE that otherwise do not lead to systematic errors in the model output. This shows that rigid pixelwise metrics do not necessarily capture the quality of wind field SR. Relying on LR feature representations of the HR static data, results in a more efficient generator design but seems to loose out on topographically related variations at smaller scales. The down-sampling effects of the HR input branch is reflected in phase space as a spectral deficit compared to the CNN2 model output which directly ingest the static HR information after the SR stage of the model at smaller scales. Relying on LR feature representations of HR static data is however a much more efficient design. The WGAN model is further able to effectively use these LR representations to almost completely reproduce the distribution of variance in phase space for all possible scales. A slight spectral deficit compared to ground truth target is observed in the absence of HR static data. This indicates that the nature of the injected fine scale details associated with the GAN training is twofold. Firstly the ability to generate finer scale variations related to topography and secondly sharper gradients associated with large scale flow setting and smaller scale flow features.

The GB domain contains a higher portion of homogeneous offshore conditions and less prominent topographic features than the CE domain. Variations in the wind field data associated with topography are thus a relatively larger error source in the CE domain, which is reflected in the domain averaged MSE metric. The MSE errors further indicate that HR topographic information does not improve performance on the GB domain, while the spatial analysis shows that terrain related systematic errors are reduced in the presence of HR static data. Domain averaged metrics are thus own their own not a completely reliable measure to evaluate the model performance. Patch training likely improves the spatial generalizing ability of the models by avoiding overfitting to specific CE domain characteristics such as the Alps and improves the utilization of HR static data. It would be interesting to compare the performance of non patch-trained models on the unseen GB domain to the results of this thesis to confirm these ideas. The performance on the GB domain can likely be improved by training the final models further on data from the domain itself, i.e. via transfer learning.

The trained models show low wind magnitude bias but seem to suffer more from persistent angular errors in complex terrain. Further spatial error analysis for individual flow fields is needed to break down the nature of these deviations. Some disagreement in the wind climatology is further observed at the more topographically complex onshore site. The site is located in a moderately sloped hill with more open land to the west and a narrow valley to the east. Systematic overpredictions in wind speed magnitude results in an under representation of lower wind speeds ≤ 5 m/s, especially for westerly winds. The higher sample moments further lead to deviations in the Weibull fitting. The reason for the observed deviations are unclear but are likely related to overrepresented orographic speed up effects or underrepresented slow down in the valley. Extracting output for the WGAN and CNN models that do not incorporate HR static data would help to determine the role of terrain height data in this deviation.

A detailed analysis of run-time performance has not been presented due to time constraints. Despite taking a few days to train, the forward filtering of data through the net-

works is very fast. The LR to HR process executes in minutes after the networks are trained. The WGAN training takes much longer than training only the generator as every fifth iteration requires an update step for both discriminator and generator networks. A vast amount of the memory consumption during training being caused by optimization computations. The critic network has significantly more parameters than the generator but fewer convolutional layers. The WGAN further suffers from the lack of interpretability of the unsupervised training framework. The training was further found to be very sensitive to the scaling parameter β associated with the adversarial loss term in the generator loss function. Values of $\beta > 0.001$ lead to unstable behaviour and very high critic model output. It is further unclear why the WGAN critic network training loss progressively increases. The authors of the WGAN gradient penalty framework observe similar behaviour when training on a relatively small dataset and associate it with overfitting. It is however encouraging, and contradictory to overfitting, that the validation loss converges to zero, meaning that the critic evaluates the generated and real data equally. Further inspection of the gradient penalty term might shed light on the nature of this behaviour which is likely related to implementation errors associated with the training loss calculations. The high loss spikes at the end of training for the CNN1 models likely arise from a combined effect of a small batch size, controlled by GPU memory limits in this study, and a too high upper limit of individual parameter learning rates in the Adam optimizer. Early stopping could be implemented to avoid such issues. Other ways to mitigate this problem are switching to SGD at a later point of training and use a learning rate scheduler or adaptive learning rate clipping proposed by Ede and Beanland (2020).

The majority of this work has only considered deterministic wind field SR. GANs are however well suited to deal with stochasticity and can be envisaged for ensemble-based probabilistic downscaling. This has been investigated to some extent by training two WGAN models that incorporate an input noise term by design, with and without a static HR input branch. The results along with a brief description of the procedure and ensemble analysis of predictions made by the models are shown in Appendix B. This limited analysis shows good promise for uncertainty quantification, with more ensemble spread associated with terrain and small scale flow features. MSE for the quantitative flow field examples was reduced by up to 10% for the CE domain and 20% for the GB domain by using ensemble averaged model output, at the cost of slight spectral deficit compared to the deterministic model output.

It should be noted that other CNN architectures might be more suited for wind field SR. Höhlein et al. (2020) found that a UNet architecture was the best performing model of several CNN architectures, including the EnhanceNet design which is largely similar to the SRResNet type generator network considered in this study. UNets are able to efficiently extract multi-scale features by design and can thus be taken as a natural choice for enhancing near surface wind fields which are determined by a complex interplay of atmospheric motion at different scales. UNets can further be implemented with generative abilities and have been applied for stochastic downscaling of precipitation (Harris et al., 2022). However an obvious down cost of UNets is the need of interpolating the LR fields to the target dimension before being passed to the main model processing. It would be interesting to train and compare the performance of a generative UNet to the SRGAN developed in this study.

Höhlein et al. (2020) further found that LR static data improves the predictive performance of their trained CNNs. Experiments with coarsened LSM and HGT data as additional LR inputs however did not seem to yield significant improvements in MSE for the generator CNN variants considered in this study. This information could be more beneficial when

the LR input data is from a different numerical model simulation that has a different representation of the terrain. The use of other dynamic predictors as inputs to their CNNs such as the forecast surface roughness, boundary layer height and geopotential only had a slight impact on the predictive performance, with the surface roughness field being almost negligible. The forecast roughness is provided on the same coarse grid resolution as the predictor winds and the authors suggest that all the information it provides is likely already embedded in the wind components themselves. However it is likely that HR static roughness length or land-use data will be more beneficial, especially in the case of highly forested areas and over very heterogeneous surfaces. It could furthermore benefit the models with a more accurate representation of the land-sea transition.

With numerical model simulations being the closest to ground truth information of gridded HR wind fields, it is not currently possible to completely replace mesoscale weather models with deep learning for wind field downscaling. Deep learning methods learn from examples and will thus at best match the performance of dynamical downscaling. The NEWA database is however arguable one of the best currently available source of HR target wind fields for developing a state-of-art network for wind field SR. It is further clear that the idealized SR training framework adopted in this study is not able to conceive a complete deep learning based surrogate model that can be directly applied to global reanalysis data. Ways of interpolating forcing data onto the WRF grid or using graph neural networks to deal with grid irregularities, should be carefully evaluated and compared to enable a more realistic training framework. Using the WRF parent model grid for learned multi-resolution downscaling is however arguably the most attractive and robust option, that can benefit from the current level of expertise in WRF based downscaling within the wind energy community. The output from the shared parent domain simulations in the NEWA project were unfortunately not stored in the process, making it impossible for this study to adopt and assess this training framework.

6 Conclusions

Turning back to the central question of the study, from section 1.3:

“Can deep learning be considered a reliable technique for mesoscale wind field downscaling in the purpose of wind resource assessment?”

The suitability of deep learning methods for wind field downscaling in the context of wind energy is largely determined by the statistical properties of resulting mesoscale fields. Physical properties of the generated flow fields and generalizing ability of the models to new model domains are also highly relevant. The analysis presented in this thesis has lead to promising results that largely confirm these abilities and motivate further research towards a more complete deep learning model framework. Spatial analysis of CNN model output shows that systematic angular and magnitude deviations are largely limited to complex terrain. The models are however able to leverage high-resolution topographic data to drastically improve the reconstruction by lowering systematic under predictions in wind speed magnitude. Training CNNs in a generative adversarial framework yields more fine scale variability in the model output that is not associated with systematic magnitude errors but seems to enhance angular deviations in complex terrain.

6.1 Further work

A natural step forward, building on the work of this thesis, is to test and train the considered deep learning models on more NEWA domains to further investigate topographically related reconstruction errors and generalizing abilities. The use of more height levels is furthermore highly relevant to evaluate important siting parameters such as wind shear. Further ensemble analysis with stochastic SR is needed to determine if GANs are able to meaningfully address the uncertainty in the reconstruction of wind fields, possibly by the means of rank statistics as done in Leinonen, Nerini, and Berne (2020). The time dimension has been completely neglected in this work, as in other work that has dealt with wind field SR. The use of a recurrent networks architecture, wrapping a convolutional network core, in a generative framework could be implemented for probabilistic modelling of the temporal evolution of wind fields.

Bibliography

- Abadi, Martín et al. (2016). *TensorFlow: A system for large-scale machine learning*. DOI: 10.48550/ARXIV.1605.08695. URL: <https://arxiv.org/abs/1605.08695>.
- Annau, Nic (2022). "Generative Adversarial Networks for Extreme Super-Resolution and Downscaling of Wind Fields at Convection-Permitting Scales". In: *ECMWF - MAELSTROM Machine Learning Workshop*. URL: <https://events.ecmwf.int/event/294/timetable/>.
- Arjovsky, Martin, Soumith Chintala, and Léon Bottou (2017). *Wasserstein GAN*. DOI: 10.48550/ARXIV.1701.07875. URL: <https://arxiv.org/abs/1701.07875>.
- Badger, Jake et al. (2014). "Wind-Climate Estimation Based on Mesoscale and Microscale Modeling: Statistical–Dynamical Downscaling for Wind Energy Applications". In: *Journal of Applied Meteorology and Climatology* 53.8, pp. 1901–1919. DOI: 10.1175/JAMC-D-13-0147.1. URL: <https://journals.ametsoc.org/view/journals/apme/53/8/jamc-d-13-0147.1.xml>.
- Buduma, Nikhil and Nicholas Locascio (2017). *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*. 1st. O'Reilly Media, Inc. ISBN: 1491925612.
- Copernicus Land Monitoring Service (n.d.). *CORINE Land Cover, available at:* URL: <https://land.copernicus.eu/pan-european/corine-land-cover>.
- Danielson, J. J. and D. B. Gesch (2010). *Global multi-resolution terrain elevation data 2010 (GMTED2010)*. URL: <https://pubs.usgs.gov/of/2011/1073/>.
- Dee, D. P. et al. (2011). "The ERA-Interim reanalysis: configuration and performance of the data assimilation system". In: *Quarterly Journal of the Royal Meteorological Society* 137.656, pp. 553–597. DOI: <https://doi.org/10.1002/qj.828>.
- Deutscher Wetterdienst (n.d.). *The World in Weather Charts. Archive DWD analysis charts*. URL: http://www1.wetter3.de/archiv_dwd_dt.html.
- (2021). *Wetter- und Klimalexikon*. [Online; accessed 10.May 2022]. URL: <https://www.dwd.de/DE/service/lexikon/Functions/glossar.html?lv2=102936&lv3=103198>.
- Dong, Chao, Chen Change Loy, Kaiming He, et al. (2014). "Learning a Deep Convolutional Network for Image Super-Resolution". In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Cham: Springer International Publishing, pp. 184–199. ISBN: 978-3-319-10593-2.
- Dong, Chao, Chen Change Loy, and Xiaou Tang (2016). "Accelerating the Super-Resolution Convolutional Neural Network". In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, pp. 391–407. ISBN: 978-3-319-46475-6.
- Donlon, Craig J. et al. (2012). "The Operational Sea Surface Temperature and Sea Ice Analysis (OSTIA) system". In: *Remote Sensing of Environment* 116. Advanced Along Track Scanning Radiometer(AATSR) Special Issue, pp. 140–158. ISSN: 0034-4257. DOI: <https://doi.org/10.1016/j.rse.2010.10.017>. URL: <https://www.sciencedirect.com/science/article/pii/S0034425711002197>.
- Dörenkämper, Martin and Andrea N. Hahmann (2019). *Code sharing for NEWA mesoscale wind atlas*. URL: <https://github.com/newa-wind/Mesoscale>.
- Dörenkämper, Martin, Bjarke T. Olsen, et al. (Oct. 2020). "The Making of the New European Wind Atlas - Part 2: Production and evaluation". In: *Geoscientific Model Development* 13.10, pp. 5079–5102. ISSN: 19919603. DOI: 10.5194/gmd-13-5079-2020.

- Dörenkämper, Martin, Michael Optis, et al. (June 2015). "On the Offshore Advection of Boundary-Layer Structures and the Influence on Offshore Wind Conditions". In: *Boundary-Layer Meteorology* 155.3, pp. 459–482. ISSN: 15731472. DOI: 10.1007/s10546-015-0008-x. URL: <https://link.springer.com/article/10.1007/s10546-015-0008-x>.
- ECMWF (2019). "IFS Documentation CY46R1 - Part VII: ECMWF Wave Model". In: *IFS Documentation CY46R1*. IFS Documentation 7. ECMWF. DOI: 10.21957/21g1hoiuo. URL: <https://www.ecmwf.int/node/19311>.
- Ede, Jeffrey M and Richard Beanland (Mar. 2020). "Adaptive learning rate clipping stabilizes learning". In: *Machine Learning: Science and Technology* 1.1, p. 015011. DOI: 10.1088/2632-2153/ab81e2.
- Fritzsche, Manuel, Shuhang Gu, and Radu Timofte (2019). "Frequency Separation for Real-World Super-Resolution". In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pp. 3599–3608. DOI: 10.1109/ICCVW.2019.00445.
- Glorot, Xavier and Y. Bengio (Jan. 2010). "Understanding the difficulty of training deep feedforward neural networks". In: *Journal of Machine Learning Research - Proceedings Track 9*, pp. 249–256.
- Glorot, Xavier and Yoshua Bengio (13–15 May 2010). "Understanding the difficulty of training deep feedforward neural networks". In: *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Yee Whye Teh and Mike Titterington. Vol. 9. Proceedings of Machine Learning Research. Chia Laguna Resort, Sardinia, Italy: PMLR, pp. 249–256. URL: <https://proceedings.mlr.press/v9/glorot10a.html>.
- Goodfellow, Ian et al. (2014). "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems*. Ed. by Z. Ghahramani et al. Vol. 27. Curran Associates, Inc. URL: <https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf>.
- Gulrajani, Ishaan et al. (2017). *Improved Training of Wasserstein GANs*. DOI: 10.48550/ARXIV.1704.00028. URL: <https://arxiv.org/abs/1704.00028>.
- GWEC (2021a). "Global Offshore Wind Report".
- (2021b). "Global Wind Report".
- Hahmann, Andrea N. et al. (Oct. 2020). "The making of the New European Wind Atlas - Part 1: Model sensitivity". In: *Geoscientific Model Development* 13.10, pp. 5053–5078. ISSN: 19919603. DOI: 10.5194/gmd-13-5053-2020.
- Harris, Lucy et al. (2022). *A Generative Deep Learning Approach to Stochastic Downscaling of Precipitation Forecasts*. DOI: 10.48550/ARXIV.2204.02028. URL: <https://arxiv.org/abs/2204.02028>.
- He, Kaiming et al. (Dec. 2016a). "Deep residual learning for image recognition". In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 2016-Decem, pp. 770–778. ISSN: 10636919. DOI: 10.1109/CVPR.2016.90. arXiv: 1512.03385.
- (2016b). "Identity Mappings in Deep Residual Networks". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 9908 LNCS, pp. 630–645. ISSN: 16113349. DOI: 10.1007/978-3-319-46493-0_38. arXiv: 1603.05027.
- Herlau, T., M. N. Schmidt, and M. Mørup (2020). *Introduction to Machine Learning and Data Mining*.
- Hersbach, Hans et al. (July 2020). "The ERA5 global reanalysis". In: *Quarterly Journal of the Royal Meteorological Society* 146.730, pp. 1999–2049. ISSN: 1477-870X. DOI: 10.1002/QJ.3803.

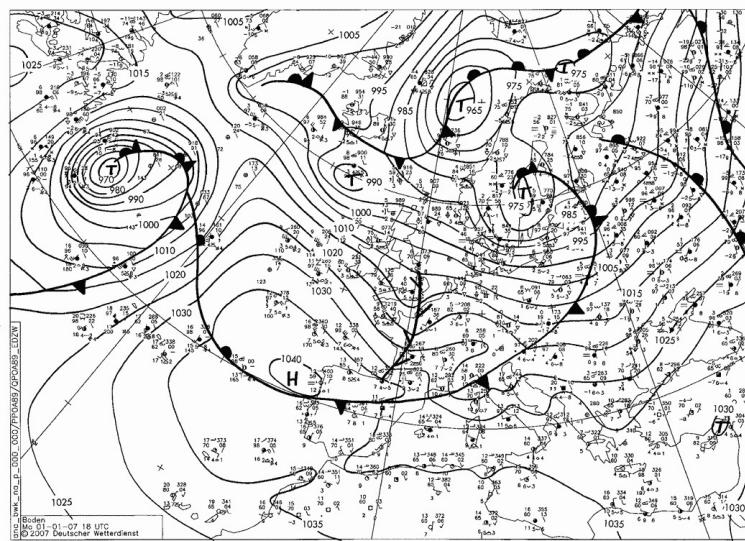
- Hochreiter, Sepp and Jürgen Schmidhuber (Nov. 1997). "Long Short-Term Memory". In: *Neural Comput.* 9.8, pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. URL: <https://doi.org/10.1162/neco.1997.9.8.1735>.
- Höhlein, Kevin et al. (Nov. 2020). "A comparative study of convolutional neural network models for wind field downscaling". In: *Meteorological Applications* 27.6, e1961. ISSN: 1469-8080. DOI: 10.1002/MET.1961.
- Holton, James R. and Gregory J. Hakim (2013). "Chapter 1 - Introduction". In: *An Introduction to Dynamic Meteorology (Fifth Edition)*. Ed. by James R. Holton and Gregory J. Hakim. Fifth Edition. Boston: Academic Press, pp. 1–29. ISBN: 978-0-12-384866-6. DOI: <https://doi.org/10.1016/B978-0-12-384866-6.00001-5>.
- Hu, Xiaodan et al. (June 2019). "RUNet: A Robust UNet Architecture for Image Super-Resolution". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.
- IEA (2021). *Global Energy Review 2021*. Paris. URL: <https://www.iea.org/reports/global-energy-review-2021>.
- Ioffe, Sergey and Christian Szegedy (2015). *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*. DOI: 10.48550/ARXIV.1502.03167. URL: <https://arxiv.org/abs/1502.03167>.
- Johnson, Justin, Alexandre Alahi, and Li Fei-Fei (2016). "Perceptual Losses for Real-Time Style Transfer and Super-Resolution". In: *Computer Vision – ECCV 2016*. Ed. by Bastian Leibe et al. Cham: Springer International Publishing, pp. 694–711. ISBN: 978-3-319-46475-6.
- Kelly, Mark (2020). *Micrometeorology for Wind Energy*. Course notes in DTU 46100.
- Kingma, Diederik P and Jimmy Lei Ba (2015). "Adam: A Method for Stochastic Optimization". In: *undefined*. arXiv: 1412.6980v9.
- Ledig, Christian et al. (2017). *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*.
- Leinonen, Jussi, Daniele Nerini, and Alexis Berne (Nov. 2020). "Stochastic Super-Resolution for Downscaling Time-Evolving Atmospheric Fields With a Generative Adversarial Network". In: *IEEE Transactions on Geoscience and Remote Sensing* 59.9, pp. 7211–7223. ISSN: 0196-2892. DOI: 10.1109/TGRS.2020.3032790. arXiv: 2005.10374.
- Lindborg, Erik (1999). "Can the atmospheric kinetic energy spectrum be explained by two-dimensional turbulence?" In: *Journal of Fluid Mechanics* 388, pp. 259–288. DOI: 10.1017/S0022112099004851.
- Maraun, Douglas, Martin Widmann, and José M. Gutiérrez (2019). "Statistical downscaling skill under present climate conditions: A synthesis of the VALUE perfect predictor experiment". In: *International Journal of Climatology* 39.9, pp. 3692–3703. DOI: <https://doi.org/10.1002/joc.5877>.
- Mortensen, Niels Gylling (2020). *Wind resource assessment using WAsP software*. English. DTU Wind Energy E 0211. Denmark: DTU Wind Energy.
- Nastrom, G. D. and K. S. Gage (1985). "A Climatology of Atmospheric Wavenumber Spectra of Wind and Temperature Observed by Commercial Aircraft". In: *Journal of Atmospheric Sciences* 42.9, pp. 950–960. DOI: 10.1175/1520-0469(1985)042<0950:ACOAWS>2.0.CO;2. URL: https://journals.ametsoc.org/view/journals/atsc/42/9/1520-0469_1985_042_0950_acoaws_2_0_co_2.xml.
- NEWA (n.d.). *The New European Wind Atlas*. URL: <https://map.neweuropeanwindatlas.eu/>.
- Price, Ilan and Stephan Rasp (Mar. 2022). "Increasing the accuracy and resolution of precipitation forecasts using deep generative models". In: p. 2022. DOI: 10.48550/arxiv.2203.12297. arXiv: 2203.12297.

- Rodrigo, Javier Sanz et al. (Jan. 2020). "The New European Wind Atlas Model Chain". In: *Journal of Physics: Conference Series* 1452, p. 012087. DOI: 10.1088/1742-6596/1452/1/012087. URL: <https://doi.org/10.1088/1742-6596/1452/1/012087>.
- Ronneberger, Olaf, Philipp Fischer, and Thomas Brox (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by Nassir Navab et al. Cham: Springer International Publishing, pp. 234–241. ISBN: 978-3-319-24574-4.
- Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams (1986). "Learning representations by back-propagating errors". In: *Nature* 323, pp. 533–536.
- Sajjadi, Mehdi S. M., Bernhard Schölkopf, and Michael Hirsch (2017). "EnhanceNet: Single Image Super-Resolution Through Automated Texture Synthesis". In: *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 4501–4510. DOI: 10.1109/ICCV.2017.481.
- Saleh Abdalla Lars Isaksen, Peter A.E.M. Janssen and Nils Wedi (2013). "Effective spectral resolution of ECMWF atmospheric forecast models". In: *ECMWF Newsletter No. 137*, pp. 19–22.
- Sha, Yingkai et al. (Dec. 2020). "Deep-Learning-Based Gridded Downscaling of Surface Meteorological Variables in Complex Terrain. Part I: Daily Maximum and Minimum 2-m Temperature". In: *Journal of Applied Meteorology and Climatology* 59.12, pp. 2057–2073. ISSN: 1558-8424. DOI: 10.1175/JAMC-D-20-0057.1. URL: <https://journals.ametsoc.org/view/journals/apme/59/12/jamc-d-20-0057.1.xml>.
- Shi, Wenzhe et al. (2016). "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network". In: *CoRR* abs/1609.05158. arXiv: 1609.05158. URL: <http://arxiv.org/abs/1609.05158>.
- Skamarock, William et al. (June 2008). *A Description of the Advanced Research WRF Version 3*. DOI: 10.13140/RG.2.1.2310.6645.
- Skamarock, William C. (2004). "Evaluating Mesoscale NWP Models Using Kinetic Energy Spectra". In: *Monthly Weather Review* 132.12, pp. 3019–3032. DOI: 10.1175/MWR2830.1. URL: <https://journals.ametsoc.org/view/journals/mwre/132/12/mwr2830.1.xml>.
- Stackoverflow (2018). *Drawing a convolution with Tikz*. URL: <https://tex.stackexchange.com/questions/437007/drawing-a-convolution-with-tikz>.
- Stengel, Karen et al. (July 2020). "Adversarial super-resolution of climatological wind and solar data". In: *Proceedings of the National Academy of Sciences* 117.29, pp. 16805–16815. ISSN: 0027-8424. DOI: 10.1073/PNAS.1918964117.
- (2021). *Physics-Informed Resolution-Enhancing GANs (PhIRE GANs)*. URL: <https://github.com/NREL/PhIRE>.
- Tran, Duy Tan et al. (Oct. 2020). "GANs enabled super-resolution reconstruction of wind field". In: *Journal of Physics: Conference Series* 1669.1, p. 012029. ISSN: 1742-6596. DOI: 10.1088/1742-6596/1669/1/012029.
- Troen, Ib and Erik Lundtang Petersen (1989). *European Wind Atlas*. English. Risø National Laboratory. ISBN: 87-550-1482-8.
- Vandal, Thomas et al. (July 2018). "Generating High Resolution Climate Change Projections through Single Image Super-Resolution: An Abridged Version". In: pp. 5389–5393. DOI: 10.24963/ijcai.2018/759.
- Wang, Jiali et al. (Oct. 2021). "Fast and accurate learned multiresolution dynamical downscaling for precipitation". In: *Geoscientific Model Development* 14.10, pp. 6355–6372. ISSN: 19919603. DOI: 10.5194/GMD-14-6355-2021. arXiv: 2101.06813.
- Wang, Xintao et al. (Sept. 2018). "ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 11042 LNCS, pp. 103–114. DOI: 10.1007/978-3-030-00422-0_9.

- Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 11133 LNCS*, pp. 63–79. ISSN: 16113349. DOI: 10.1007/978-3-030-11021-5_5. arXiv: 1809.00219.
- Wilby, Robert L. and Tom M. L. Wigley (1997). “Downscaling general circulation model output: a review of methods and limitations”. In: *Progress in Physical Geography* 21, pp. 530–548.
- Yang, Chih Yuan, Chao Ma, and Ming Hsuan Yang (2014). “Single-image super-resolution: A benchmark”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 8692 LNCS.PART 4, pp. 372–386. ISSN: 16113349. DOI: 10.1007/978-3-319-10593-2_25.
- Yang, Wenming et al. (2019). “Deep Learning for Single Image Super-Resolution: A Brief Review”. In: *IEEE Transactions on Multimedia* 21.12, pp. 3106–3121. DOI: 10.1109/TMM.2019.2919431.
- Zhou, Jie et al. (2018). *Graph Neural Networks: A Review of Methods and Applications*. DOI: 10.48550/ARXIV.1812.08434. URL: <https://arxiv.org/abs/1812.08434>.

A Supporting material

A.1 Synoptic setting



(a) 2007-01-01 18:00 UTC

Figure A.1: Synoptic setting over Europe and North Atlantic three hours before the selected CE and GB NEWA domain data field examples shown in Figure 4.1 and Figure 4.2

A.2 Wind rose

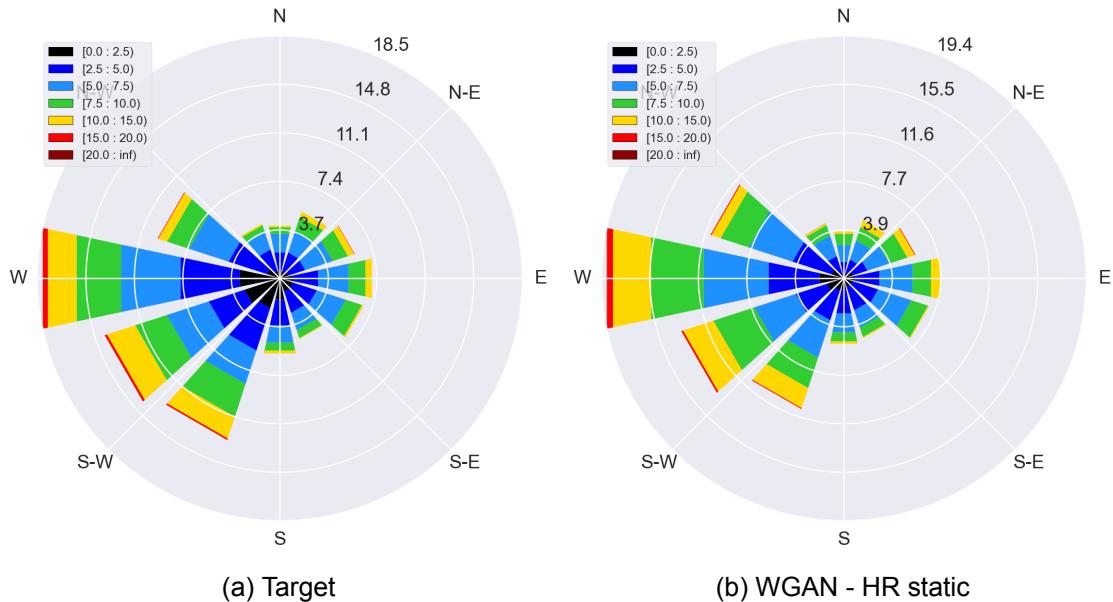


Figure A.2: Wind rose from target and WGAN model output data for the topographically complex site considered in this study.

B Ensemble Analysis

By including a stochastic input source in the generator design the stochastic nature of GANs can be investigated for wind field SR. Two stochastic WGAN (SWGAN) models have been trained using a random input component to a CNN1 type generator, in an otherwise unchanged setting as the deterministic CNN1. The network modifications involve the concatenation of eight LR features sampled randomly from a standard Gaussian distribution to the other input features after the first convolutional stage of the model and the HR input branch. The number of features in the HR branch and in the LR input convolution layer are reduced by four to have the same total number of input features (i.e. 64). In the absence of a HR input branch then 8 features, or as many features as the random component, are removed from the first input convolutional layer.

A LR rendering of the HR target, considering the same test set examples as in the quantitative analysis shown in subsection 4.2.1, has been filtered through the model 100 times to create an ensemble of reconstructions. Output for all trained WGAN models, including the ensemble averages of SWGAN outputs, are shown in Appendix B on Figure B.1 for the CE domain and Figure B.2 for the GB domain. The WGAN model output becomes slightly sharper with the introduction of HR static data and some smoothing is involved with the ensemble averages. Ensemble standard deviation for the reconstructed flow fields are shown in Figure B.3 for the CE domain case. The ensembles have up to 1 m/s spread in individual grid points near sharp gradients and orographic flow features observed in the ground truth data. The ensemble standard deviation on the GB domain case, shown in Figure B.4, similarly measures 1 m/s spread quite extensively near gradients and smaller convective features in the target flow field.

Figure B.5 shows a view of the deterministic WGAN model spectral behaviour at smaller scales, along with the spectral density of an ensemble averaged field from SWGAN output. The ensemble average marginally lacks spectral density compared to the deterministic WGAN.

From this limited analysis it can be concluded that stochastic SR is able to give a measure of uncertainty in the reconstruction by the means of ensembles. This unavoidably leads to some smoothing in the resulting averaged wind field predictions which is reflected in the spectral domain as a slight defect in spectral density. The drastic improvement in terms of MSE is likely due to induced smoothing that mitigates the strong error from misplaced fine scale features.

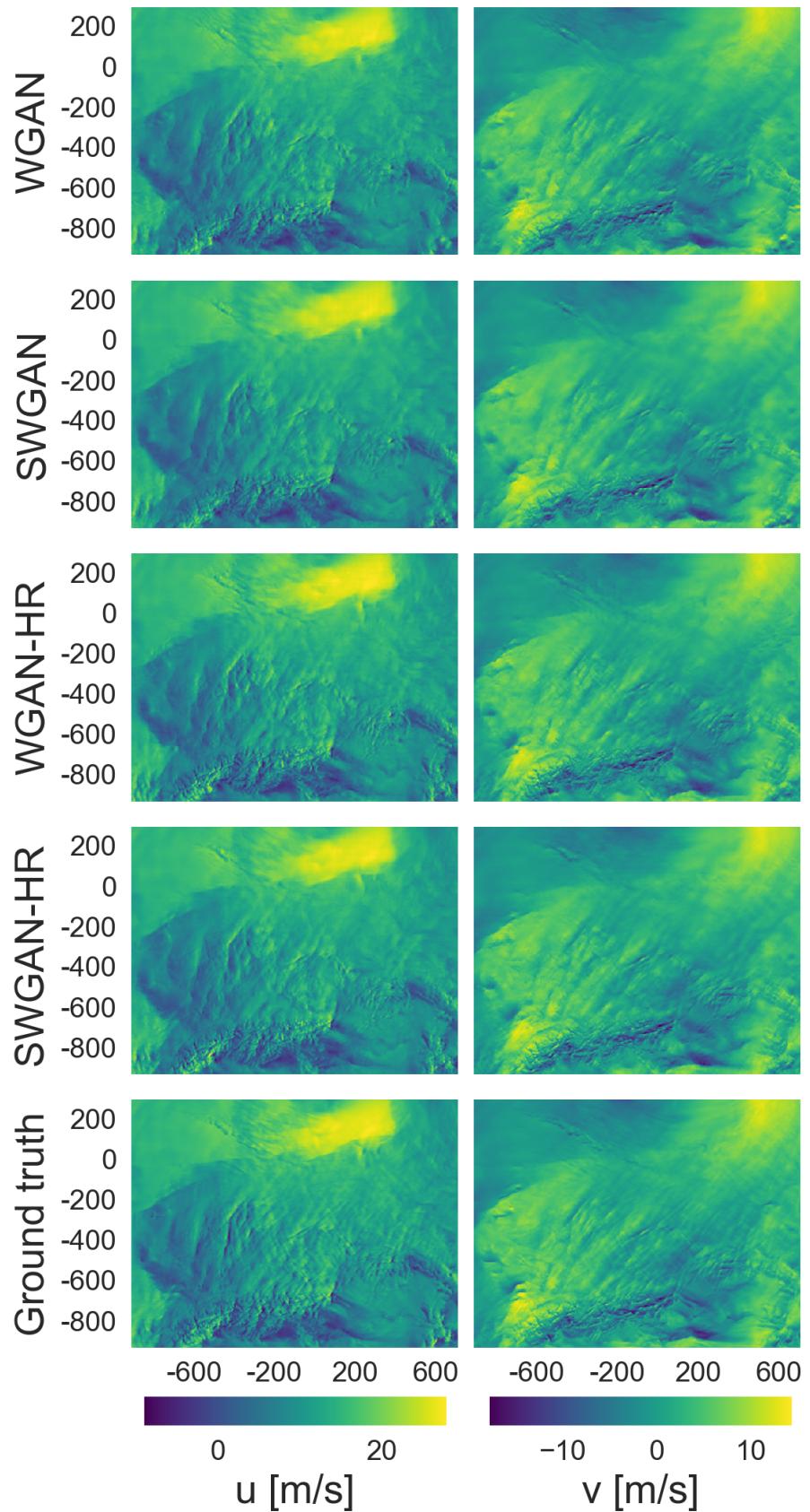


Figure B.1: Comparison of WGAN model output and test target from the CE domain.

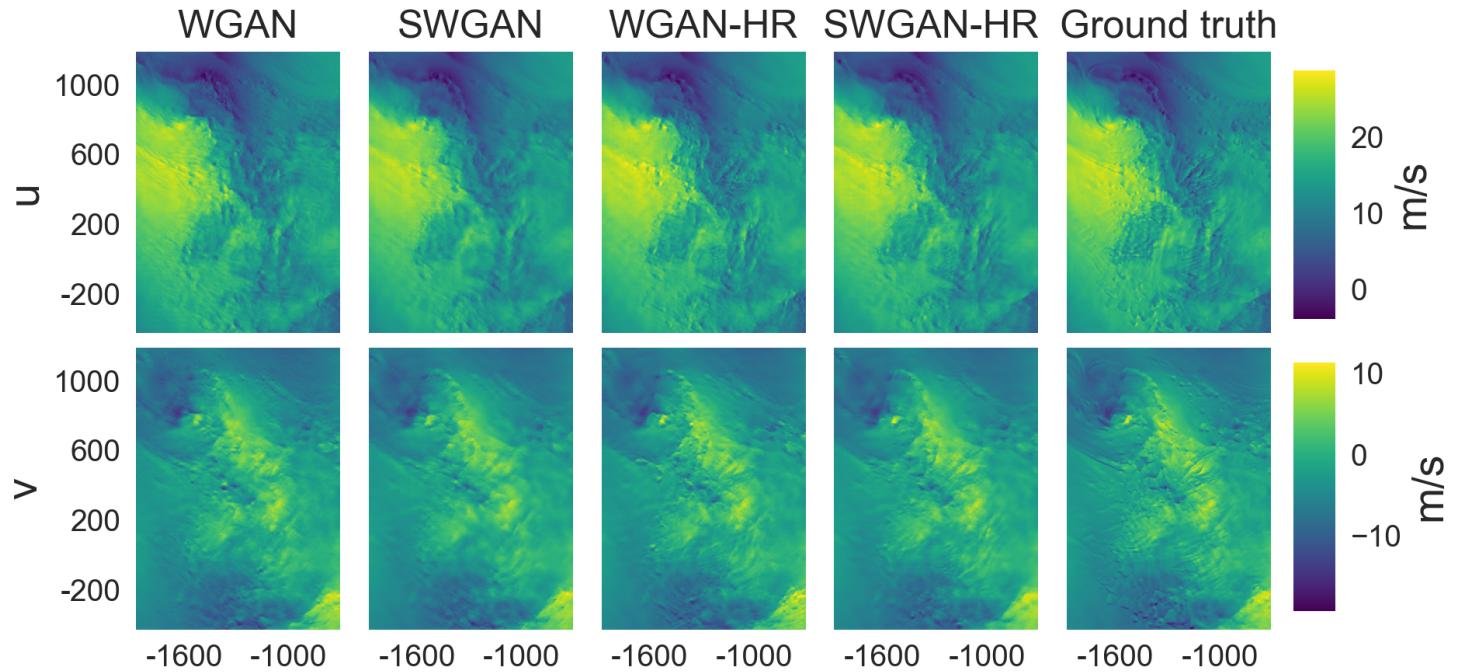


Figure B.2: Comparison of WGAN model output and test target from the GB domain.

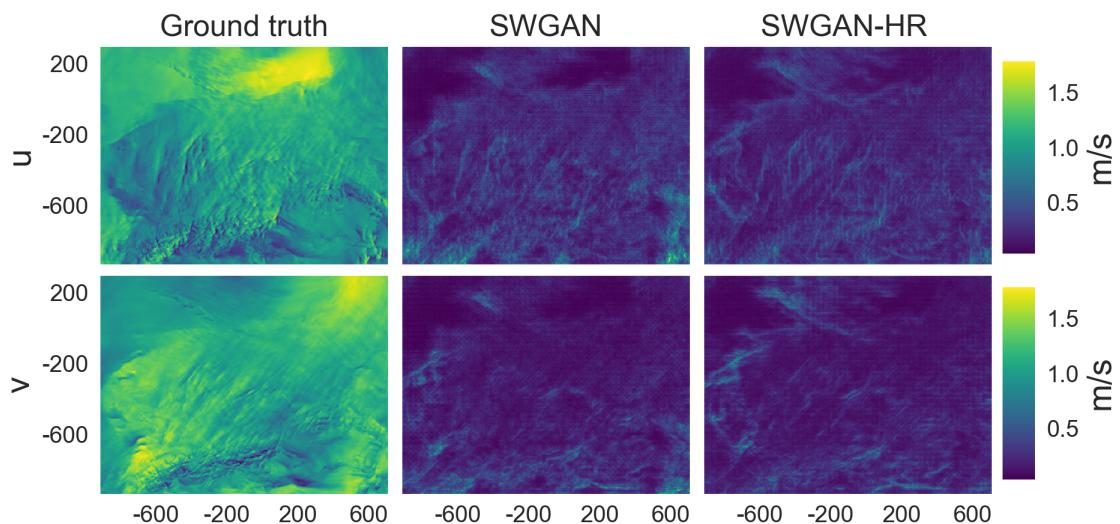


Figure B.3: Ensemble standard deviation of 100 images generated by the stochastic WGAN models with and without HR static data. Ground truth target from the CE domain, not following the same color grading, is shown for reference.

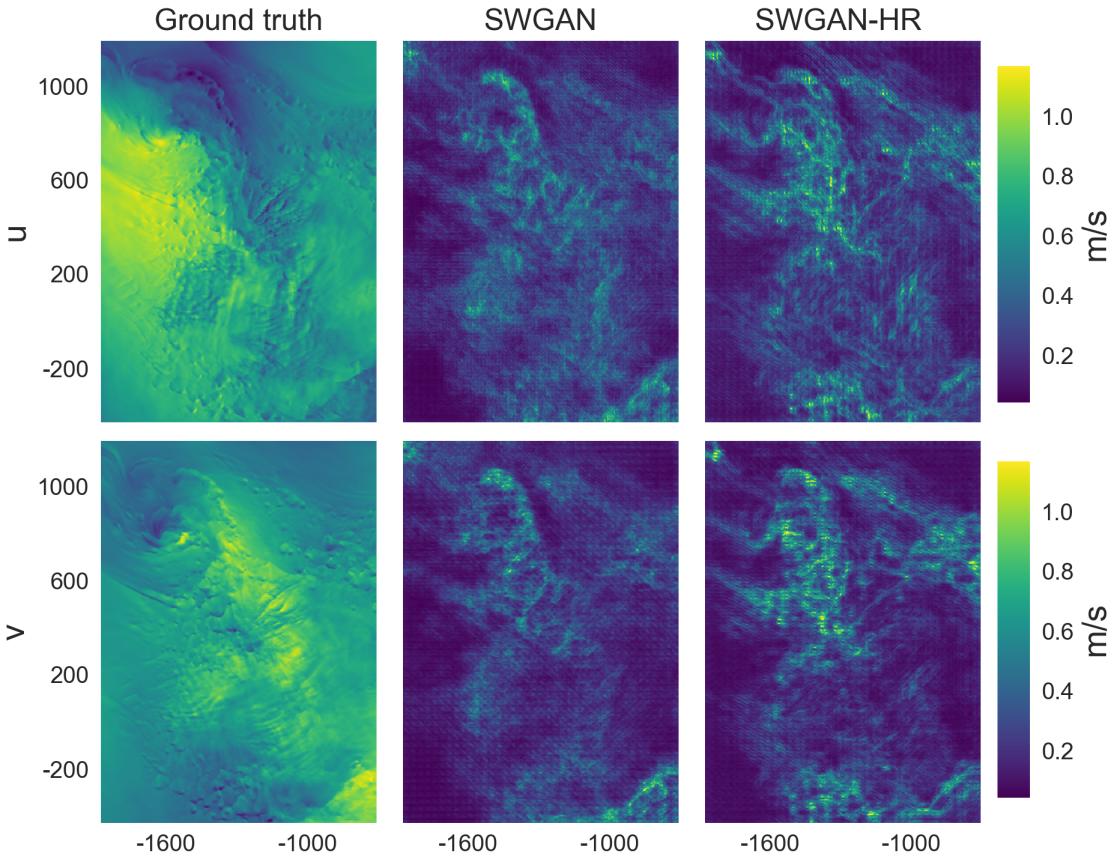


Figure B.4: Ensemble standard deviation of 100 images generated by the stochastic WGAN models with and without HR static data. Ground truth target from the GB domain, not following the same color grading, is shown for reference.

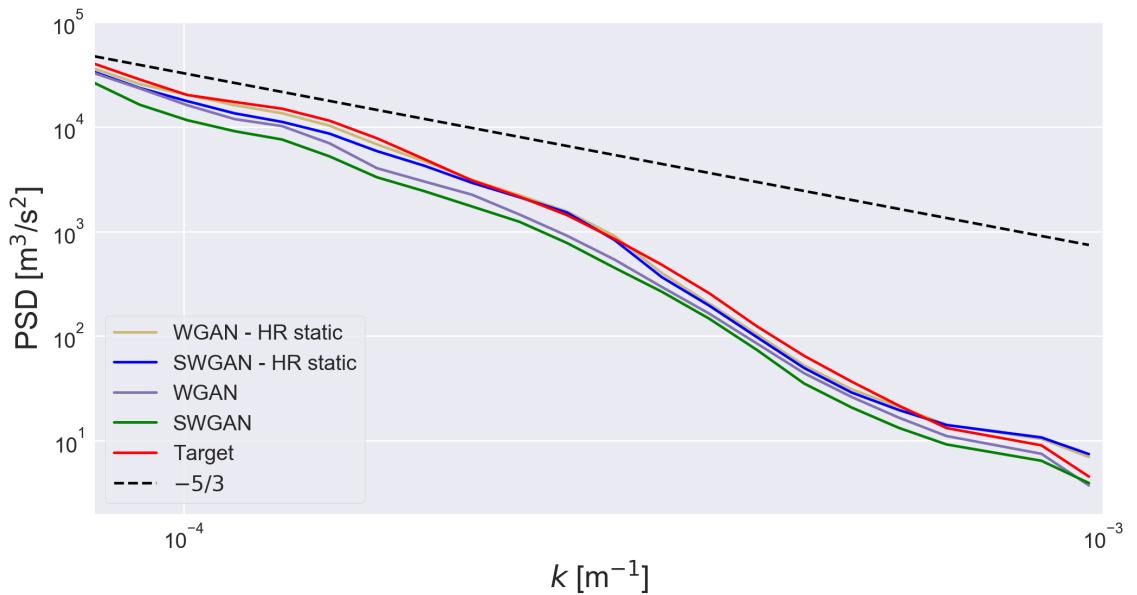


Figure B.5: Radially averaged 2D spectrum for WGAN output and target data. Dashed line depicts the expected $-5/3$ spectral decay in turbulent mesoscale flow.

Table B.1: Domain averaged MSE [m^2/s^2] for the WGAN model output on one example from the test set.

Model	CE	GB
WGAN	4.15	2.84
SWGAN	3.81	2.38
WGAN - HR static	3.19	2.61
SWGAN - HR static	3.00	2.28

Table B.2: Relative improvement in domain averaged MSE compared to deterministic approach.

Model	CE	GB
SWGAN	-9.0%	-19.6%
SWGAN - HR static	-6.3%	-14.7%

C Training

C.1 WGAN training algorithm

Table C.1: Pseudo code for WGAN with gradient penalty (Gulrajani et al., 2017)

Algorithm 1 WGAN with gradient penalty. We use default values of $\lambda = 10$, $n_{\text{critic}} = 5$, $\alpha = 0.0001$, $\beta_1 = 0$, $\beta_2 = 0.9$.

Require: The gradient penalty coefficient λ , the number of critic iterations per generator iteration n_{critic} , the batch size m , Adam hyperparameters α, β_1, β_2 .
Require: initial critic parameters w_0 , initial generator parameters θ_0 .

```
1: while  $\theta$  has not converged do
2:   for  $t = 1, \dots, n_{\text{critic}}$  do
3:     for  $i = 1, \dots, m$  do
4:       Sample real data  $\mathbf{x} \sim \mathbb{P}_r$ , latent variable  $\mathbf{z} \sim p(\mathbf{z})$ , a random number  $\epsilon \sim U[0, 1]$ .
5:        $\tilde{\mathbf{x}} \leftarrow G_\theta(\mathbf{z})$ 
6:        $\hat{\mathbf{x}} \leftarrow \epsilon \mathbf{x} + (1 - \epsilon) \tilde{\mathbf{x}}$ 
7:        $L^{(i)} \leftarrow D_w(\tilde{\mathbf{x}}) - D_w(\mathbf{x}) + \lambda(\|\nabla_{\hat{\mathbf{x}}} D_w(\hat{\mathbf{x}})\|_2 - 1)^2$ 
8:     end for
9:      $w \leftarrow \text{Adam}(\nabla_w \frac{1}{m} \sum_{i=1}^m L^{(i)}, w, \alpha, \beta_1, \beta_2)$ 
10:   end for
11:   Sample a batch of latent variables  $\{\mathbf{z}^{(i)}\}_{i=1}^m \sim p(\mathbf{z})$ .
12:    $\theta \leftarrow \text{Adam}(\nabla_\theta \frac{1}{m} \sum_{i=1}^m -D_w(G_\theta(\mathbf{z})), \theta, \alpha, \beta_1, \beta_2)$ 
13: end while
```

C.2 Loss curves

C.2.1 CNN models

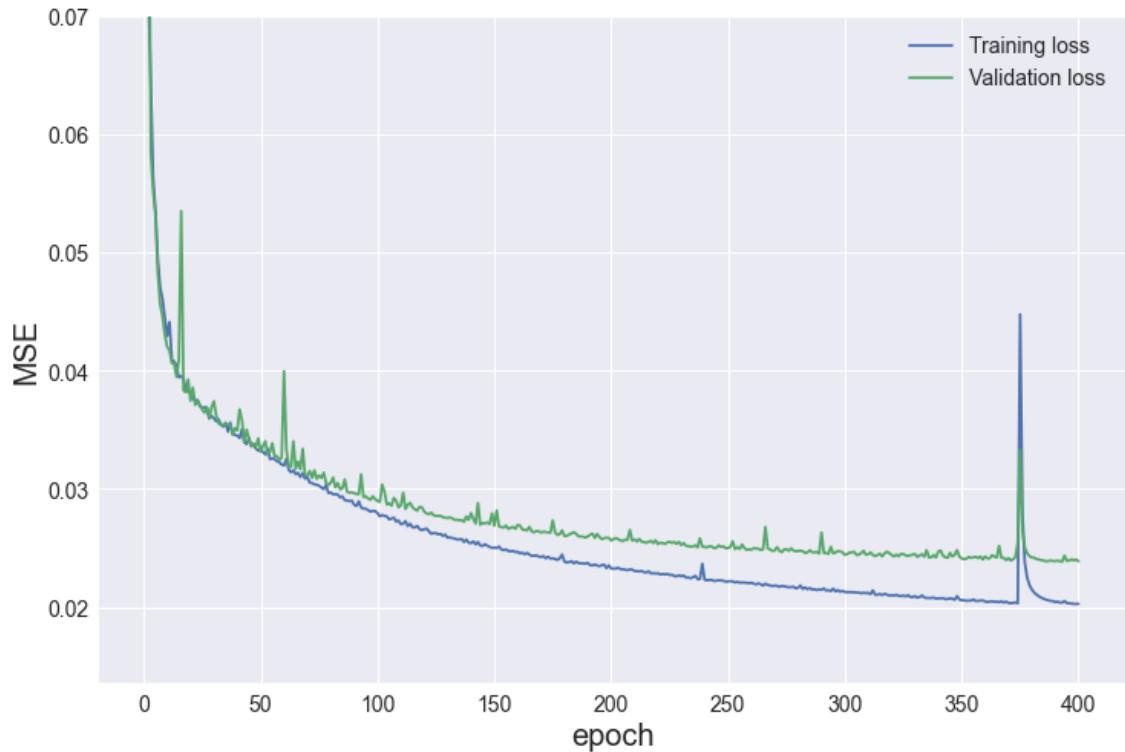


Figure C.1: Training and validation loss curves for CNN1

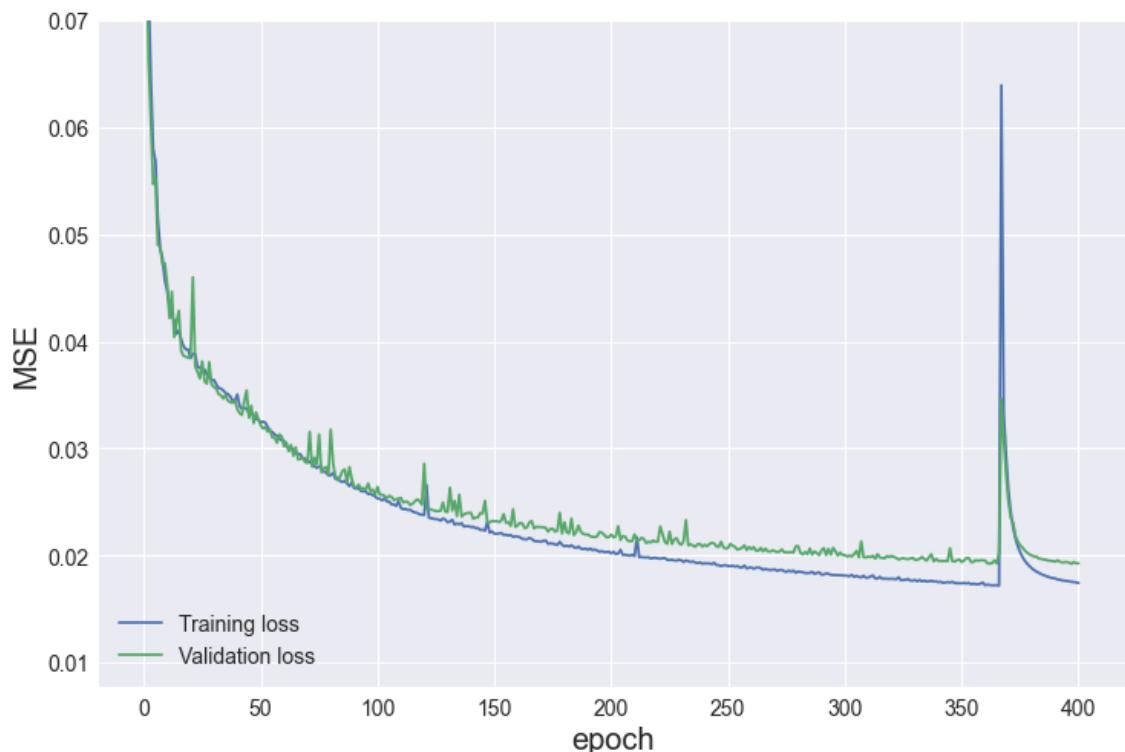


Figure C.2: Training and validation loss curves for CNN1 - HR static.

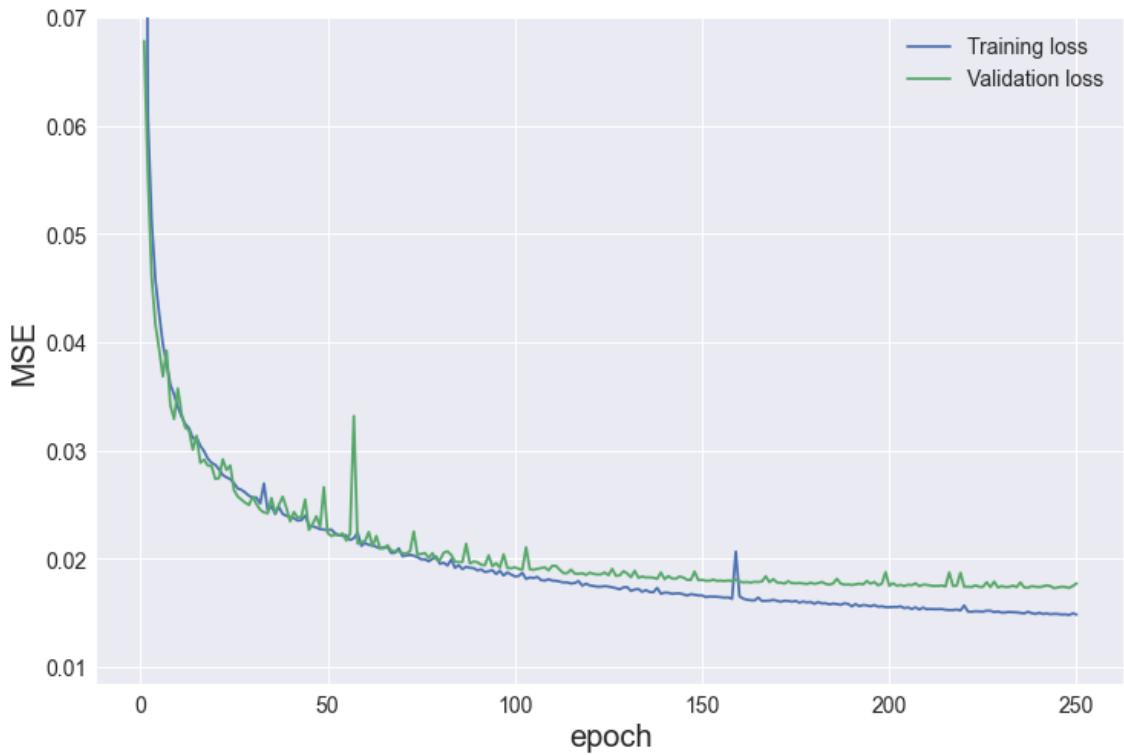
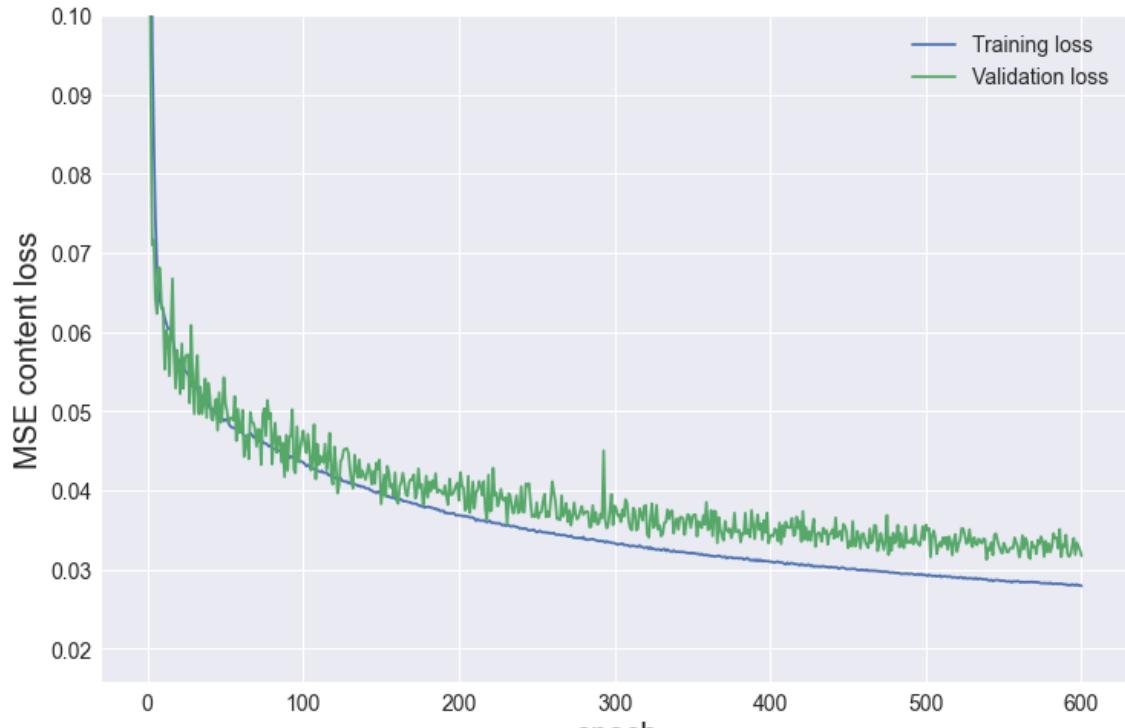
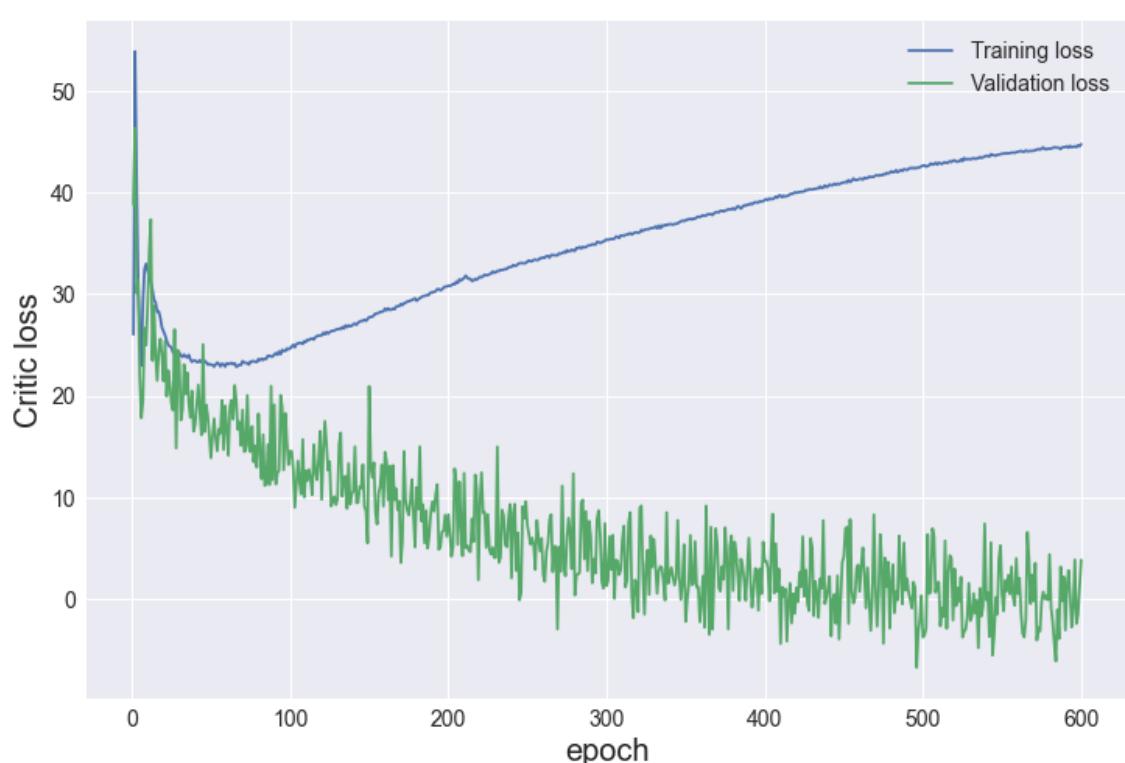


Figure C.3: Training and validation loss curves for CNN2 with HR static input.

C.2.2 WGAN models

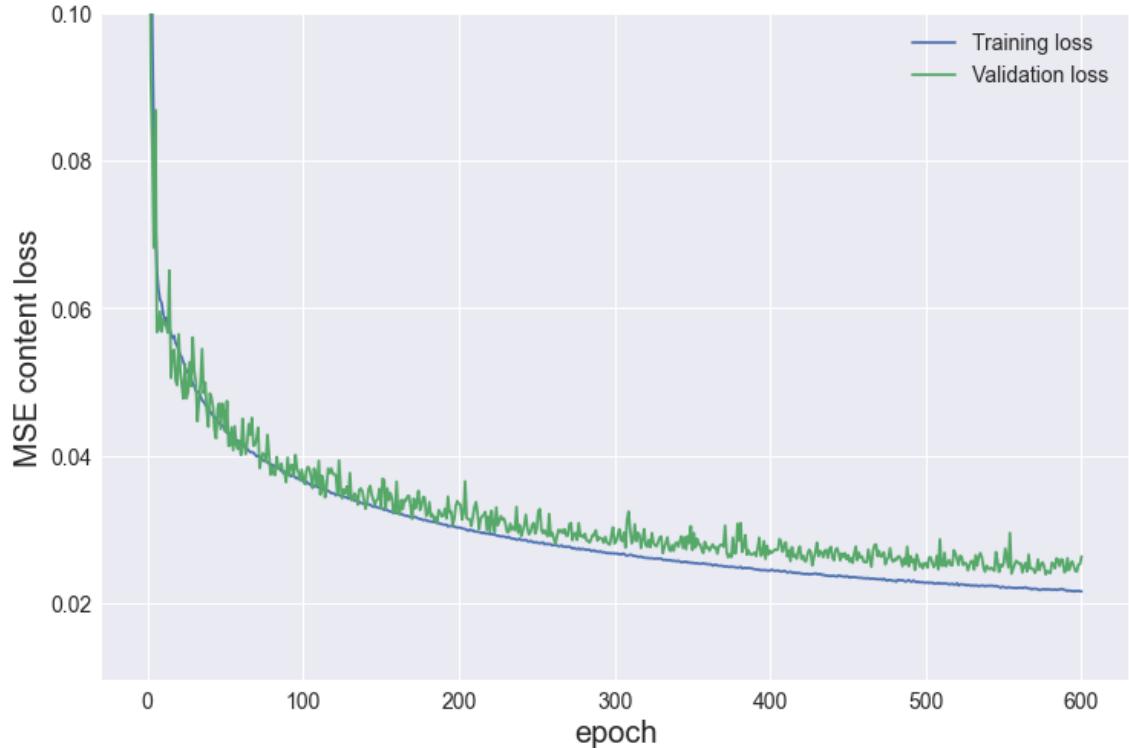


(a)

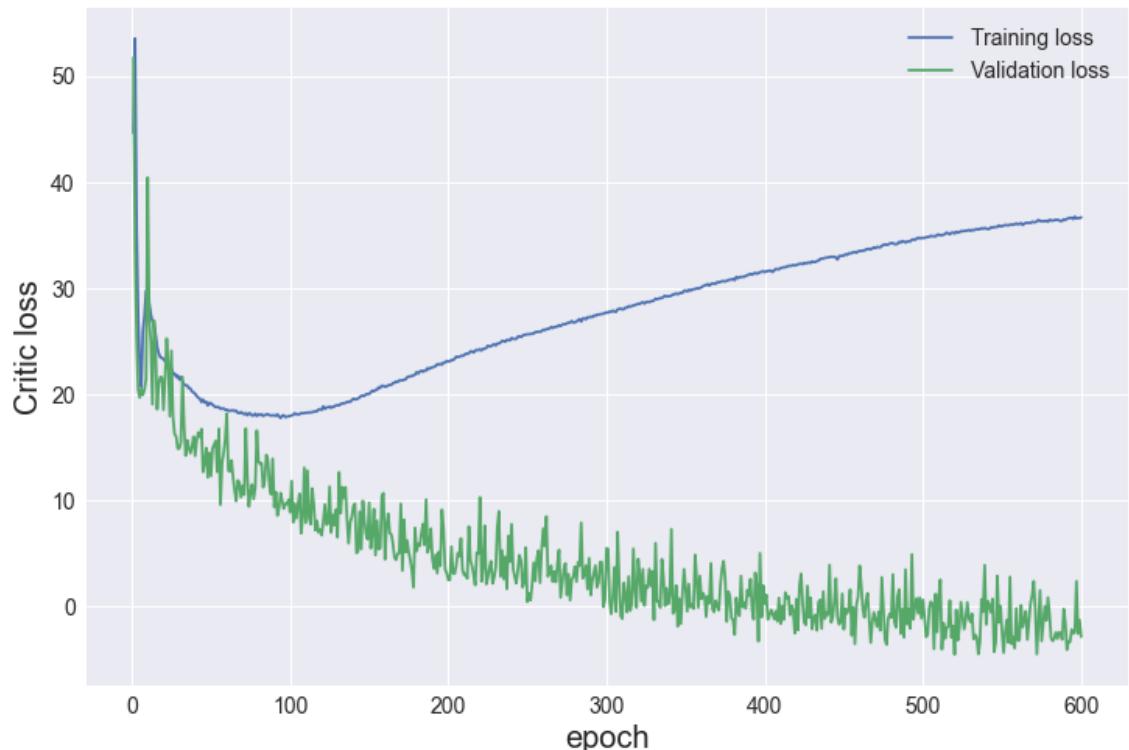


(b)

Figure C.4: Training and validation loss curves for WGAN.



(a)



(b)

Figure C.5: Training and validation loss curves for WGAN - HR static.

Statement of Authentication

I hereby confirm that this Master's Thesis is entirely my own work. I confirm that no part of the written submission has been copied from either a book or any other source, including the internet, except where such sections are clearly marked as quotations, and the sources have been correctly identified within the text or in the list of references. Moreover, I confirm that I have followed the guidelines of good scientific practice of the University of Oldenburg ("Leitlinien guter wissenschaftlicher Praxis der Carl von Ossietzky Universität Oldenburg").

Erklärung

Hiermit versichere ich, dass ich diese Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe. Außerdem versichere ich, dass ich die allgemeinen Prinzipien wissenschaftlicher Arbeit und Veröffentlichung, wie sie in den Leitlinien guter wissenschaftlicher Praxis der Carl von Ossietzky Universität Oldenburg festgelegt sind, befolgt habe.

Location: Reykjavik Iceland

Date: 24.06.2022

Signature: Gísli Þjóður Helgason

Technical
University of
Denmark

Frederiksborgvej, 339
4000 Roskilde
Tlf. 4525 1700

www.vindenergi.dtu.dk