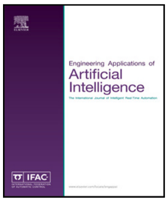
Engineering Applications of Artificial Intelligence 139 (2025) 109533

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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



An image-to-image adversarial network to generate high resolution wind data over complex terrains from weather predictions

Jaime Milla-Val [a], Carlos Montañés [a], Norberto Fueyo [b,*]

[a] *Nabladot S.L., Salvador Allende 75, 50015 Zaragoza, Spain*
[b] *Instituto de Investigación en Ingeniería de Aragón (I3A), University of Zaragoza, María de Luna 3, 50018 Zaragoza, Spain*

ARTICLE INFO

Keywords:
Microscale wind prediction
Computational fluid dynamic
Numerical weather prediction
Image-to-image
Deep learning
conditional Generative Adversarial Network

ABSTRACT

In this work, we propose a Machine Learning method to predict detailed wind fields over extensive, complex terrains. The ability to predict local wind fields is becoming increasingly important for a range of applications, including sports in Nature, large outdoors events, light-aircraft flying, or the management of natural disasters. The intricate nature of wind dynamics, particularly in regions with complex orography such as a mountain range, presents a major challenge to traditional forecasting models. This work presents an efficient way to predict local wind conditions with a high resolution, similar to that of Computational Fluid Dynamics (CFD), in large geographical areas with complex terrain, using the results from relatively coarse (and therefore economical) data from Numerical Weather Prediction (NWP). To achieve this goal, we developed a conditional Generative Adversarial Neural network model (cGAN) to convert NWP data into CFD-like simulations. We apply the method to a rugged region in the Pyrenees mountain range in Spain. The results show that the proposed model outperforms traditional Machine Learning methods, such as Support Vector Machines (SVM), in terms of accuracy and computational efficiency. The method is four orders of magnitude *faster than* traditional CFD. *Mean Average Errors of* $1.36\ \mathrm{m/s}$ *for wind speed and 18.73° for wind direction are obtained with the proposed approach.*

1. Introduction

Understanding and modelling wind local fields is becoming a matter of increasing importance for various applications, especially for domains with a complex terrain. Sediment transport is driven by local wind patterns; Wakes et al. (2010) used numerical modelling of wind over complex dune topography to predict the motion of sediments for land planning purposes. Wind plays a crucial role in snow accumulation in mountainous regions, as Gerber et al. (2017) show in a case study in Switzerland using a multi-scale non-hydrostatic atmospheric simulation and prediction tool. They concluded that a more detailed simulation of the flow and of additional processes, such as avalanches, are required to better match the experimental data. Later, Vionnet et al. (2021) combined mesoscale simulations with down-scaling techniques for snow redistribution modelling in the Canadian Rockies. Their study showed mismatches between simulated results and validation data on snow accumulation, especially in scenarios driven by wind fields that did not capture lee-side flow recirculations. The propagation of wildfires is also highly affected by wind. Sutherland et al. (2023) implemented a spatial-dependent factor to better simulate wind speed in fire situations with a diverse vegetation canopy. As the demand for energy rises around the world, and climate-change evidence mounts, renewable energy sources are essential to ensure sustainable energy production, and wind energy plays a substantial role in covering the demand. Accurate, reliable, and fast tools are needed for wind field modelling and wind turbine wake development assessment to improve and optimise wind farm operation and design (Elgendi et al., 2023).

Computational Fluid Dynamics (CFD) provides the necessary local detail for the wind field, with resolutions of meters or less. Blocken et al. (2015) modelled with CFD the complex terrain of a northwestern harbour in Spain for boat manoeuvring and pollution dispersion applications. On the other hand, Numeric Weather Predictions (NWP) are used to simulate atmospheric wind conditions over a relatively large area, and has resolutions of up to a few hundred meters. Mughal et al. (2017) assessed the sensitivity and validated the use of NWP software to simulate the speed and direction of wind on complex terrain in east Africa; Prósper et al. (2019) also applied NWP to wind power forecasting in an onshore wind farm with complex topography in northwestern Spain.

Both NWP and CFD can be used in a combined fashion; NWP can model the general atmospheric situation and provide boundary

[*] Corresponding author.
E-mail address: Norberto.Fueyo@unizar.es (N. Fueyo).

https://doi.org/10.1016/j.engappai.2024.109533
Received 10 April 2024; Received in revised form 22 September 2024; Accepted 21 October 2024
Available online 11 November 2024

conditions for the CFD prediction of the local wind field for real scenarios (Temel et al., 2018). CFD produces highly detailed results but with a considerably greater computational cost than NWP, which provides lower resolution data. For fast, cheap, and accurate wind field modelling applications, it would be ideal to obtain the finer resolution results of CFD with the computational speed of NWP. This can be achieved with machine learning techniques and is the rationale of the present study.

2. Literature review

Deep neural networks and machine learning have a great potential to enhance wind simulation, particularly when combined with CFD (Vinuesa and Brunton, 2022). The main focus is to obtain CFD-like results while reducing computational time. Yousif et al. (2023) used a combination of transformers and super-resolution networks for a synthetic-inflow generator to spatially developing turbulent boundary layers; and Jiang et al. (2023a) presented a transformer-based decoder for flow field prediction in airfoil design. In the context of wind energy production, image recognition methods have been employed to identify and avert premature deterioration of wind turbine blades (Yang et al., 2021). Fang et al. (2023) utilised a speech recognition technique in their landslide prediction modelling for early warning systems in Turkey. For wind simulation and prediction in larger areas, a combination of linear regression models and multi-variable LSTM and NARX networks for short-term forecasting have been studied in the Andes Mountains (López and Arboleya, 2022); Chen et al. (2021) also used an LSTM in combination with a CNN for short-term forecast of 2-D regional wind speed.

Generative Adversarial Networks (GAN), along with their variants, are powerful machine learning frameworks that have been used in several wind-related applications. Deep GAN were used for the design of new aerofoils of vertical-axis wind turbines (Santos et al., 2023). Another variant of GAN that uses a Wasserstein distance was employed by Zhang et al. (2024) to generate stochastic scenarios of wind power production. In fault diagnosis detection for wind turbine gearboxes, where data are often difficult to obtain, Liang et al. (2023) used GAN for data augmentation. Along a similar line of research, a GAN oversampling method was applied by Yang et al. (2023), achieving fault classification with small datasets. Ye et al. (2024) developed a day-ahead prediction model for wind power generation that combined self-attention with TimeGAN for temporal forecast. Behara and Saha (2024) proposed an incremental GAN model that introduces innovative solutions to enhance the precision and reliability of wind-turbine position analysis, providing wind characterisation to optimise wind energy generation. For single wind turbine wake prediction, Li et al. (2024) developed a model combining transformers and conditional Generative Adversarial Networks (cGAN) (Mirza and Osindero, 2014). Zhang and Zhao (2022) used deep convolutional cGAN to model wake development in wind farms.

One of the applications of cGAN is image-to-image translation. In this work, we train a cGAN image-to-image model to obtain CFD results from cheaper and coarser NWP simulations over complex terrains. Image-to-image models are designed for tasks that involve transforming an input image into an output image; converting images from one domain to another, such as converting satellite images to maps or turning sketches into realistic images (Isola et al., 2016). This allows to obtain highly detailed wind field data without incurring the computational expenses of CFD, saving computer resources and time. We use the Pix2Pix model proposed by Isola et al. (2016). It is an image-to-image deep neural network, and it belongs to the family of conditional Generative Adversarial Network (cGAN) (Mirza and Osindero, 2014). We selected the Pix2Pix model for our study because our objective is to translate NWP data (represented as images) into corresponding CFD data (also represented as images), rather than generating arbitrary CFD images. Pix2Pix, as a cGAN model, is well suited for this type of image-to-image translation task, allowing us to condition the generation process on specific input data (*i.e.*, meteorological conditions from NWP). Moreover, there exists a successful case study (for a façade dataset Tyleček and Šára, 2013) with a similar amount of training samples (400 versus our 193).

The use of other models was considered. Although StarGAN (Choi et al., 2018) offers multi-domain image translation, it is more complex and suited for style transfer and domain adaptation. Our focus was on a straightforward translation task, and we found no clear evidence of a superior performance of StarGAN for our specific purpose. The StyleGAN (Karras et al., 2019) model is primarily focused on style transfer and does not require paired images, which is not suitable for our goal of translating specific NWP data to CFD images. While diffusion models like Palette (Saharia et al., 2022) show promise, they are generally harder to train and, critically, usually require larger datasets for stability. Given our limited dataset, Pix2Pix was a more practical choice. The suitability of Pix2Pix for conditional image-to-image translation aligns well with the objectives of our study. From the seminal papers of the other considered models (Isola et al., 2016; Choi et al., 2018; Karras et al., 2019; Saharia et al., 2022), the tasks described in the Pix2Pix paper were the most appropriate for the objective and rationale of our study.

Image-to-image translation has been used recently to tackle a range of problems. Jiang et al. (2023b) used image-to-image translation, *viz.* the StyleGAN model (Karras et al., 2019), to create anime scenes; Dalva et al. (2023) for face editing; and Sun et al. (2023) to enhance underwater images. Image-to-image translation models have been employed also for applications involving unpaired medical physics images (Chen et al., 2023). Romero et al. (2024) used an image-to-image model that uses wind farm layouts and undisturbed wind field as the input to obtain the disturbed wind field caused by the wind farm for the optimisation of the wind farm layout. No recent works have been found that use image-to-image models to translate wind fields from the mesoscale into the highly detailed CFD scale; this study fills this gap.

Table 1 presents a chronological summary of the main relevant literature.

One of the novelties of this approach is the convenient arrangement of the CFD and NWP data as matrices that can be interpreted as images. Understanding the wind-field data as an image broadens the applicability of AI methods that have been developed in the field of image detection. A relevant difference from previous published works is that we do not intend to predict or forecast wind; the main objective is to train a deep learning model to translate NWP results to CFD results. An additional challenge, compared with other applications of machine learning in this field, which often use idealised test cases, is the need to model an actual, expansive, geographically complex domain, and real weather conditions.

3. Methods

This piece of research seeks to find an economical way to use relatively coarse data resulting from Numerical Weather Prediction (NWP) simulations to predict local wind conditions with a high resolution (similar to that achieved with Computational Fluid Dynamics, CFD), in large geographical areas with complex terrains.

In order to achieve CFD-like resolution from NWP-like input, we propose the use of supervised neural network models. Specifically, the proposed method uses the Pix2Pix (Isola et al., 2016) model, which belongs to the class of conditional Generative Adversarial Networks (cGANs) (Mirza and Osindero, 2014). This model was originally designed for image-to-image translation, and its ability to synthesise pictures from label maps, reconstruct objects from edge maps, and colourise images, among other capabilities, has been shown to be effective (Isola et al., 2016). In this work, we aim for the Pix2Pix model to perform a similar function to obtain a high-resolution CFD-like "image" of the wind field from the coarse NWP one, at a fraction of the computational cost.

Table 1
Literature review summary in chronological order.

Author	Method	Application
Yang et al. (2021)	Image recognition	Avert premature deterioration of wind turbine blades
Chen et al. (2021)	LSTM + CNN	Short-term forecast of 2-D regional wind speed
Zhang and Zhao (2022)	cGAN	Modelling wake development in wind farms
López and Arboleya (2022)	LSTM + NARXF	Short-term forecast of wind in the Andes Mountains
Yousif et al. (2023)	Transformers	Synthetic-inflow generator to spatially developing turbulent boundary layers
Jiang et al. (2023a)	Transformers	Flow field prediction in aerofoil design
Fang et al. (2023)	Speech recognition	Landslide prediction modelling for early warning systems in Turkey
Santos et al. (2023)	GAN	Aerofoils design of vertical-axis wind turbines
Liang et al. (2023)	GAN	Data augmentation in fault diagnosis detection for wind turbine gearboxes
Yang et al. (2023)	GAN	Data augmentation in fault diagnosis detection for wind turbine gearboxes
Jiang et al. (2023b)	Image-to-image	Create anime scenes
Dalva et al. (2023)	Image-to-image	Face editing
Sun et al. (2023)	Image-to-image	Enhance underwater images
Chen et al. (2023)	Image-to-image	Applications in unpaired medical physics images
Zhang et al. (2024)	Wasserstein-GAN	Generate stochastic scenarios of wind power production
Ye et al. (2024)	TimeGAN	Day-ahead prediction of wind power generation
Behara and Saha (2024)	GAN	Wind characterisation to optimise wind energy generation
Li et al. (2024)	Transformers + cGAN	Single wind turbine wake prediction
Romero et al. (2024)	Image-to-image	Optimisation of the wind farm layout

The present authors have in the past explored the use for this problem of more classic approaches (Milla-Val et al., 2024), such as Support Vector Machine (SVM) (Bishop, 2006), Random Forest (RF) (Breiman, 2001), K Nearest Neighbours (KNn) (Goldberger et al., 2004) or linear regression with Stochastic Gradient Descendant (SGD) (Robbins and Monro, 1951). The two main advantages of using a cGAN over these classic models previously explored are: improved scalability and dispensing with the need for feature engineering. Scalability is improved because cGANs can greatly benefit from the use of GPUs, which results in reduced computational times and affords larger training datasets. Feature engineering refers to transformations made to the original dataset in order to present it in a certain advantageous manner to the model; an example is the manipulation of the velocity components to obtain the wind speed as the absolute value of the velocity vector. With SVM, the information that is passed to the model has to be carefully selected and its features properly designed; otherwise, there is a risk of falling into the so-called Curse of Dimensionality (Bellman, 1966), resulting in a very high dimensional, and hence computationally expensive, space. In contrast, feature engineering is not needed with these convolutional models, as they take as input the whole domain at once, and the model itself detects and identifies the areas of interest for modelling.

Fig. 1 presents a flowchart with an overview of the proposed method. We use reanalysis data for the year 2018 as input for the NWP simulations and obtain 8760 hourly events. These 8760 hourly events undergo a data selection process, from which we choose 193 events that are either frequent or significant (see Section 3.1 below). From the NWP simulations of these selected events we obtain meteorological results that are used both as boundary conditions for the CFD calculations *and* as input for the cGAN model; terrain information is used in the CFD model, but not passed to the cGAN model. We then process our NWP and CFD results to present the data on a square grid of 128 × 128 points; this data grid can be interpreted as a typical digital image, which is suitable for the proposed cGAN model. This cGAN model is made of two independent AI models; the generator which consists of a convolutional encoder–decoder tuple and the discriminator. The generator and the discriminator are trained as a tandem in an adversarial fashion via two different loss functions: the discriminator tries to identify ground truth data (coming from CFD) as *true* and generated data (coming from the generator) as *fake*; the generator tries to make the discriminator identify the generated data as *true* and make these generated data as close as possible to the ground truth data (via a Mean Absolute Error metric).

3.1. Data sources for the model

The methodology followed for data generation is briefly outlined below; it is the same as used in Milla-Val et al. (2024), where full details can be found. Firstly, an NWP simulation is carried out with WRF v4.1 (Skamarock et al., 2019), using the dataset `ds090.0` (National Centers for Environmental Prediction et al., 1994) as input for the year 2018. This results in 8760 hourly weather events being simulated, from which the most frequent ones and those that are deemed to be "representative" are selected. Representative events are chosen in the manner described in Milla-Val et al. (2024); in essence, hourly events for a whole year are classified into wind-speed and wind-direction "bins" according to event wind speed and direction at a central point in the domain. Then, representative events are chosen at random from some of these bins, starting with the most populated bins, so that the selected events span the complete range of weather conditions.

Subsequently, these frequent and representative events are used as boundary conditions to run detailed CFD simulations. We use OpenFOAM v6 (Weller et al., 1998) for solving the flow in the domain of interest. The flow is modelled as isothermal, and we solve the Reynolds-Averaged Navier–Stokes equations of continuity and momentum using the SIMPLE algorithm for pressure-velocity coupling. The CFD mesh consists of approximately 250 thousand cells; it is made of octrees, and octree refinement is successively used in the three cell layers closest to the terrain. The mesh is pseudo-cylindrical, with the lateral boundary consisting of 16 planar faces. A top view of the CFD domain is shown in Fig. 3(b). As boundary conditions, OpenFOAM's `noSlip` condition was used on the terrain surface, `zeroGradient` for the top boundary, and `freeStream` for the lateral boundaries; this latter condition selects for the velocity components at the boundary cells, and depending on the flow direction, either `zeroGradient` or a fixed value interpolated from the NWP simulations. The OpenFOAM dictionaries which fully define the CFD problem are provided in Milla-Val (2024a).

NWP and CFD simulations calculate their results in different meshes. To develop the cGAN model, both sets of results are interpolated onto a common structured mesh, which we call the *auxiliary mesh*. This auxiliary mesh consists of horizontally equidistant points arranged in a square grid pattern; vertically, both the CFD domain and the auxiliary mesh extend from the surface level up to 5000 m high. The data obtained from NWP and CFD for the same area and same time period is used to train the cGAN-based model capable of reproducing CFD-like data from NWP predictions.

The modelled domain is a ridge in the Pyrenees mountain range in Spain; in Fig. 2 a photograph of the actual landscape is presented; and in Fig. 3(a), an image of its rendering in the CFD mesh. As an indication

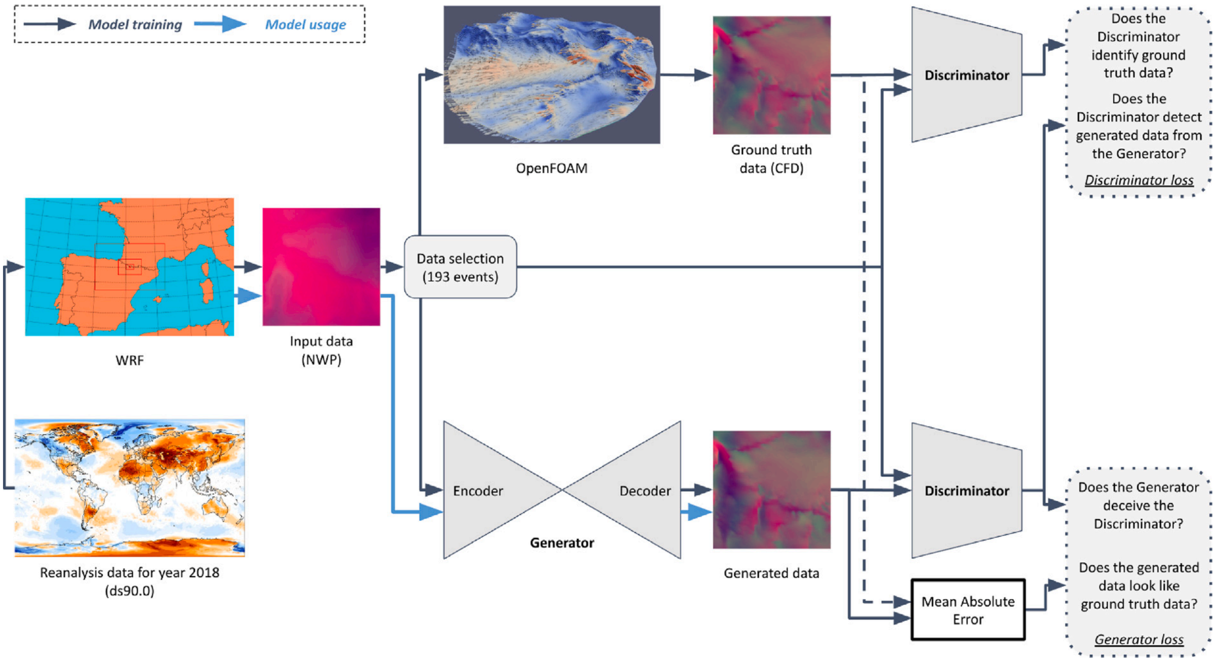

Model training
Model usage
OpenFOAM
Ground truth data (CFD)
Discriminator
Does the Discriminator identify ground truth data?
Does the Discriminator detect generated data from the Generator?
Discriminator loss
WRF
Input data (NWP)
Data selection (193 events)
Reanalysis data for year 2018 (ds90.0)
Encoder
Decoder
Generator
Generated data
Discriminator
Mean Absolute Error
Does the Generator deceive the Discriminator?
Does the generated data look like ground truth data?
Generator loss

Fig. 1. Overview of the proposed method and the training process. Black arrows denote model training and blue arrows model usage.

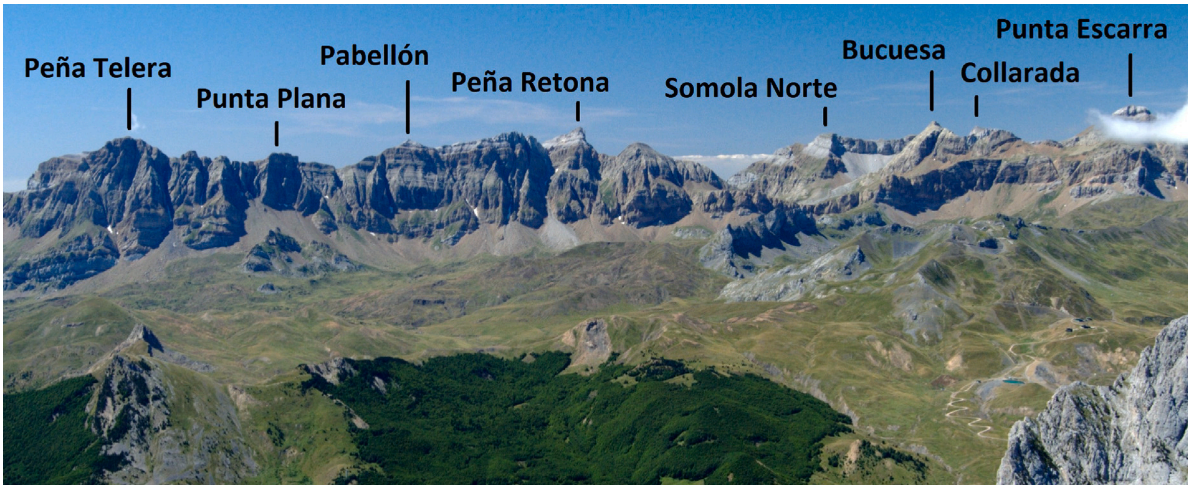


Fig. 2. Photograph of the domain: "Sierra de la Partacua", Pyrenees mountain range, Spain.

of the orographic complexity, peak elevations are around 1300 m with respect to the lowest altitude in the domain. We select this complex terrain in order to make the study more exacting, intentionally avoiding simpler domains which would result in simpler CFD simulations. The ground layer of the auxiliary mesh (the shadowed square in Fig. 3(c)) is a 5 km-wide square grid of 128 × 128 points inside, and concentric with, the cylindrical CFD domain (Fig. 3(b)).

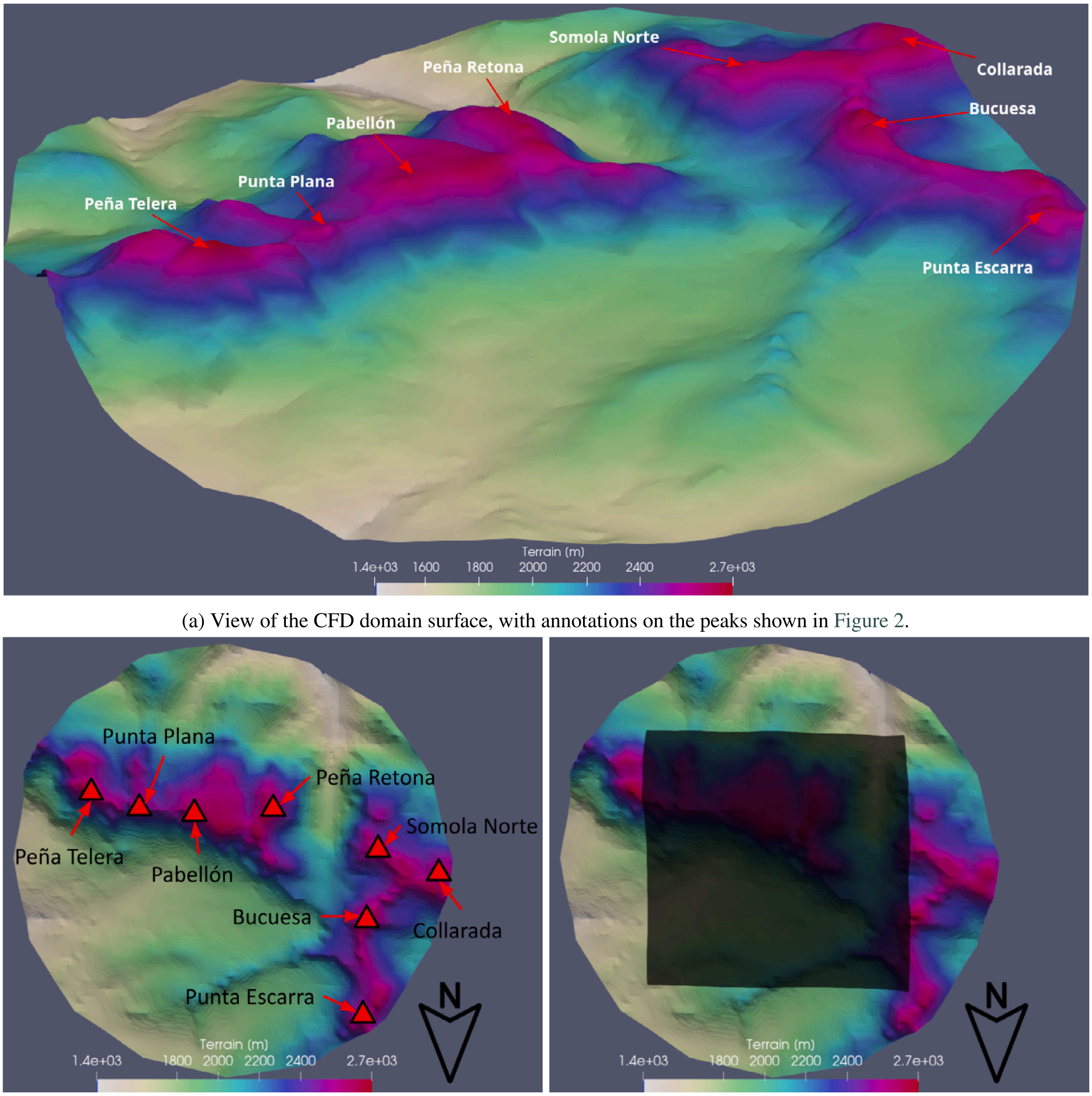
The ground level is the most challenging one to model, and for this reason it is the focus of the result analysis in this study. Also, the velocity field at this lowest level is very relevant for numerous applications, including wind farm performance prediction, human comfort in activities in Nature, and wind loads in structural analysis.

With this configuration for the auxiliary mesh, the data can be interpreted as an ensemble of paired "images" with a 128 × 128 resolution; each pair corresponds to one of the simulated events, and is composed of a coarse "image" from the NWP calculation and a fine one from the CFD simulation. We suggest that the problem so stated can be treated as an image-to-image translation one, and therefore be approached using Pix2Pix (Isola et al., 2016).

3.2. Interpreting data as an "image"

Typically, a digital image is represented by a matrix with dimensions `resX`×`resY`×`channels`; `resX`×`resY` is the number of pixels along each of the two dimensions of the picture, and `channels` represent the pixel colour. This `channels` dimension typically stores three values; for instance, in a RGB colour model, these three values are the levels of Red, Green, and Blue in the pixel.

When our NWP and CFD simulations are processed, we obtain data on a square grid of 128 × 128 points. Then, the data are arranged as an ensemble (one instance for each simulated event) of 128 × 128 images with two channels, each channel corresponding to one of the velocity components U and V. (We do not model with our cGAN the vertical component of the velocity, W, because it is small in magnitude



(a) View of the CFD domain surface, with annotations on the peaks shown in Figure 2.

(b) Top view of the CFD domain at the ground level.

(c) Same view as in Figure 3b showing the auxiliary mesh; the shadowed square area represents the auxiliary mesh at the ground level.

Fig. 3. In Figs. 3(a) and 3(b) the CFD mesh surface is presented with annotations on the peaks shown in the real footage (Fig. 2). Fig. 3(c) shows how the auxiliary mesh (shadowed square) lays within the CFD domain boundaries.

and of little relevance in practical applications.) This data arrangement can be thought of as one from a conventional digital image with dimensions $\texttt{resX} \times \texttt{resY} \times \texttt{channels}$; now the "pixels" $\texttt{resX} \times \texttt{resY} = 128 \times 128$ are the points in the spatial domain in our simulations, and the "colour" `channels` are the velocity components at each point. Therefore cGANs, as proposed for this study, appears to be a suitable paradigm as it was initially designed to work with images.

Fig. 4 exemplifies the interpretation of the data as an image. In the first row, input data to the model (the NWP results) are presented; in the second row, the targets (the CFD results) are shown. Column-wise, we present from left to right: the RGB representation of an event (the first and second channels corresponding to the U and V velocity components; the third channel is assigned a constant value so that we can use libraries that assume that inputs and targets are 3-colour-channel images); the corresponding wind speed; and the wind direction. (Wind speed and wind direction are obtained from U and V by post-processing.)

3.3. Training process

A full year (*viz* 2018) is simulated with an hourly time resolution using an NWP model. Among the 8760 hourly data events available from these simulations, 193 events are selected and simulated in CFD to build the training dataset. Additionally, a test dataset is built by randomly selecting another 80 hourly events in the same year. These events are also simulated with CFD, but they are not seen by the model during training.

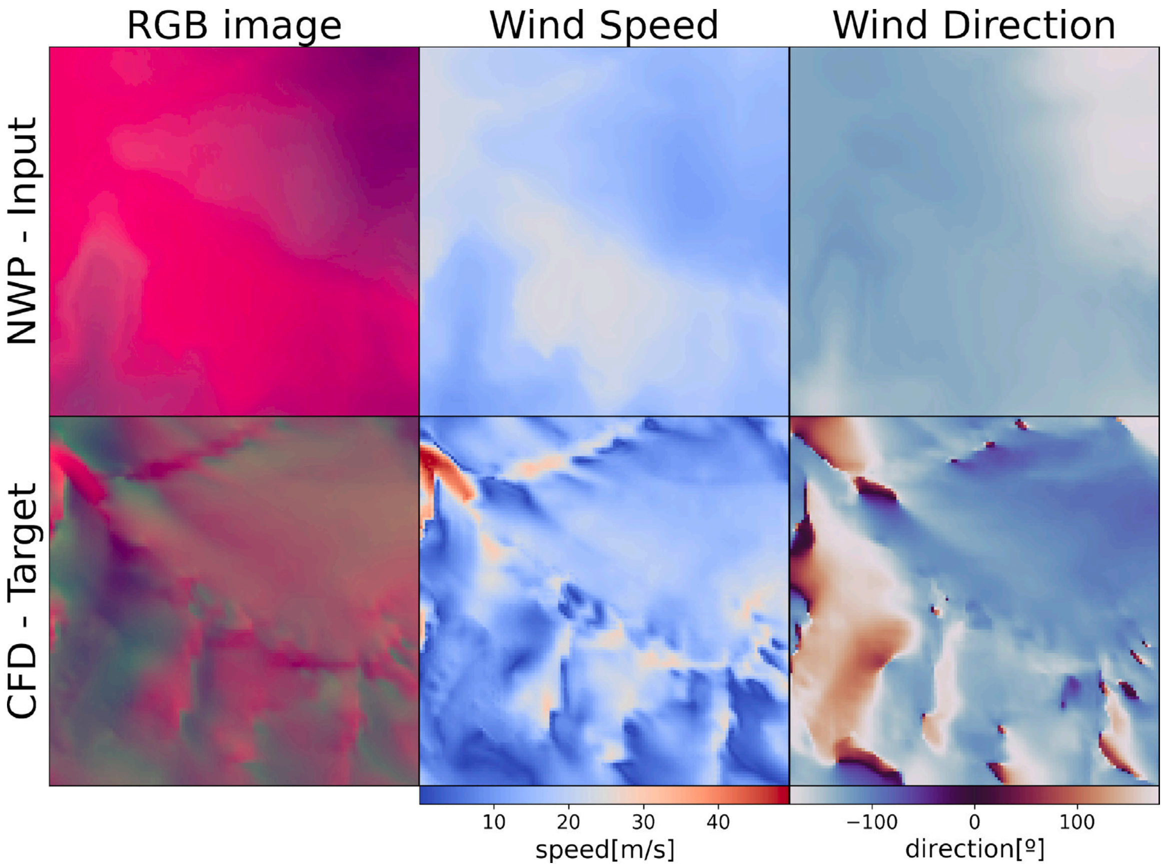


Fig. 4. Model inputs and targets. Top row: data input to our model (NWP results); bottom row: target data (CFD results). Columns from left to right: RGB representation of the two horizontal wind components, wind-speed magnitude and wind direction.

3.4. AI model

This work uses the Pix2Pix model (Isola et al., 2016), with certain modifications to make it suitable for our data. The Pix2Pix model is a conditional Generative Adversarial Network (cGAN) used for image-to-image translation, among other problems. This model is especially useful for three main reasons: it was tested on a relatively small dataset, the CMP facade database (Tyleček and Šára, 2013) (around 400 samples), which is similar to the one used in this work (around 200 samples); it is specifically designed to address image-to-image translation problems, which is the approach we adopted to correlate NWP results with CFD ones. The Pix2Pix model was selected for this study to translate NWP data into corresponding CFD data because it is well suited for conditional image-to-image translation tasks. Pix2Pix leverages conditional GANs to use specific input data (meteorological conditions from NWP) to generate CFD "images". Other models like StarGAN (Choi et al., 2018), StyleGAN (Karras et al., 2019), and Palette (Saharia et al., 2022) were considered, but found to be less suitable due to their complexity, their focus on style transfer, their use of unpaired images, or their need for larger datasets. Pix2Pix aligns well with the objectives of the study and dataset constraints.

A Generative Adversarial Network (GAN) (Goodfellow et al., 2014) is a type of artificial intelligence algorithm composed of two neural networks, a generator and a discriminator, which compete against each other to generate new data that are indistinguishable from the real one. The generator produces fresh data, while the discriminator evaluates them and decides whether they are genuine or artificial. The generator and the discriminator are trained in tandem, with the generator attempting to deceive the discriminator and the discriminator attempting to identify the generated data as false. A Conditional Generative Adversarial Network (cGAN) (Mirza and Osindero, 2014), such as the one used in this work, is an extension of the GAN model, in which the generation process is conditioned on additional information. This conditioning is accomplished in this work by feeding the discriminator the target data, that is, the CFD results.

The generator has a U-net-based architecture (Ronneberger et al., 2015), composed of an encoder (downsampler) and a decoder (upsampler) with skip connections between them. The downsampler block consists of a convolution operation and an activation function; in this work, the LeakyReLU (Maas et al., 2013) is replaced by a PReLU (He et al., 2015) and the filter size is increased to 5 (from the original 4). The upsampler block consists of a transposed convolution, dropout (applied to the first three blocks), and an activation function; the ReLU (Nair and Hinton, 2010) activation is replaced by a PReLU and the filter size is also increased to 5. The batch normalisation layers of these blocks that were present in the original model are removed. The original Pix2Pix model is designed for 256 × 256 images; however, it is adapted for 128 × 128 data by having 7 downsampler blocks and 6 upsampler blocks in the generator instead of the original 8 and 7, respectively.

The discriminator is a convolutional PatchGAN classifier (proposed in Isola et al. (2016)). This means that the discriminator output is not just a single value that indicates whether an image is real or not; instead, it produces a 7 × 7 matrix that reveals which parts of the generated image are more likely to be fabricated. The only change made to the discriminator is the increase of the filter size of the first three downsampler blocks to 7 (from the original 4). The generator and discriminator losses are similar to those proposed in the original work (Isola et al., 2016).

The loss for the generator is composed of two parts: a cross-entropy element that attempts to deceive the discriminator; and an L1 distance, which is the mean absolute error between the target data (CFD results)

Table 2
Summary of model hyperparameters.

Model agent	Element	Value
Generator	Downsampler-block: activation function	PReLU
	Downsampler-block: filter size	5
	Downsampler-block: batch normalisation	No
	Upsampler-block: activation function	PReLU
	Upsampler-block: filter size	5
	Upsampler-block: batch normalisation	No
Discriminator	Downsampler-block: activation function	PReLU
	Downsampler-block: filter size	7
	Downsampler-block: batch normalisation	Yes
General	λ	100
	Learning rate	$2 \cdot 10^{-4}$
	Batch size	32

and the generated product. Similarly as in Isola et al. (2016), the parameter λ, which balances the L1 loss and the cross-entropy term, is set to 100 as decided by the original authors. The loss for the discriminator consists of two cross-entropy elements: one that prevents it from being tricked by the generator; and another that teaches it what a real image looks like. To build the models, the TensorFlow 2 suite (v2.8) is used (Abadi et al., 2015). An Adam optimiser (Kingma and Ba, 2014) with a constant learning rate of $2 \cdot 10^{-4}$ and a batch size of 32 is used. No hyperparameter tuning is performed.

Table 2 summarises all the model hyperparameters; and a GitHub repository is provided with the full model coding and full details of the model architecture (Milla-Val, 2024a).

3.5. Metrics

In order to evaluate the accuracy of our approach, two metrics are calculated: the Mean Absolute Error (MAE) and the Structural Similarity Index Measure (SSIM) (Wang et al., 2004); several studies use the same metrics to evaluate their results; for instance, Stefenon et al. (2021) for forecasting electric grid insulator contamination; and Qian et al. (2014) to develop a multi-scale SSIM weighted metric.

MAE is computed, from the results produced by the trained generator and the ground truth given by the CFD simulation for the following variables: wind speed, $\text{MAE}_{\text{speed}}$, direction, MAE_{dir}, and the U and V velocity components, MAE_{UV}. The definition of MAE is:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \quad (1)$$

where n is the total number of observations, y_i is the actual value for the ith observation and $\hat{y}_i$ is the predicted value for the ith observation. For each predicted event, the points with a low wind speed (below 2 m/s) are excluded from the MAE calculations. Low wind speeds are not of great relevance for practical applications: they do not pose a threat to structures or buildings, are not strong enough to power most wind turbines, and they are often not noticed by, or annoying to, people. Furthermore, at these low speeds slight variations in the U or V wind components may lead to large errors in wind direction, the significance of which would not be commensurate with the small importance the low speed itself.

The Structural Similarity Index (SSIM) is calculated for a pair of images; in this case, the one produced by the generator and the one that corresponds to the CFD field. The SSIM definition for single channel images is given by:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (2)$$

where:

- x and y are the two images to be compared.
- μ_x and μ_y are the mean intensities of x and y, respectively.

  $$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad \mu_y = \frac{1}{N}\sum_{i=1}^{N} y_i$$

  where i represents a pixel and N is the total number of pixels.
- σ_x^2 and σ_y^2 are the variances of x and y, respectively.

  $$\sigma_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2 \qquad \sigma_y^2 = \frac{1}{N-1}\sum_{i=1}^{N}(y_i - \mu_y)^2$$

- σ_{xy} is the covariance of x and y.

  $$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$$

- C_1 and C_2 are small constants to stabilise the division with a weak denominator:

  $$C_1 = (K_1 L)^2 \quad \text{and} \quad C_2 = (K_2 L)^2$$

  where L is the dynamic range of the pixel values and $K_1 = 0.01$ and $K_2 = 0.03$.

SSIM gives a better understanding of whether the results produced by the machine learning model are similar to those that a CFD model would produce, and how close the generated "image" is to being CFD-like. SSIM is determined by three factors: luminance, contrast, and structure. Originally, it was designed for grayscale images, but can be easily extended to more channels by averaging the results from Eq. (2) for each channel. The SSIM is in the range $\in [0, 1]$, with values closer to 1 meaning a better match between the two images.

4. Results

The model is trained using 193 events (80% for actual training and 20% for validation) and tested on 80 additional events not used for training. The input data consist of NWP results, while the targets are CFD simulations, both in the aforementioned mountainous region in the Pyrenees range. Fig. 5 shows the results from a CFD simulation in this complex terrain; the image depicts the flow field on a surface above the terrain level, through glyphs that show the direction and speed of the local wind. The impact of the terrain complexity on the wind field can be observed; the wind blowing West to East (right to left in the image) creates an unique pattern. Bigger glyphs and more reddish colours indicate a strong local wind. For these weather conditions, some mountain peaks to the West (to the right in the image) exhibit strong winds; downstream of the peaks, wakes are developed as the air flows to the plains, where relatively open free flow can be seen. Other mountain peaks in the middle of the domain and the canyon, however, exhibit weaker winds. This reflects how the interaction between the complex terrain and the real weather conditions generates an intricate and unique wind pattern; this is the kind of pattern that we try to replicate with the cGAN model as an alternative to CFD. The output from the model is a 128 × 128 matrix with values for the two horizontal wind components, U and V, which can also be interpreted algorithmically as an image with two channels. The results of a previous model by Milla-Val et al. (2024) are presented for comparison. This previous model is a Support Vector Machine (SVM) that was trained pointwise (with very little information from the surrounding area) on a grid of 30 × 30 points, which is then re-scaled to 128 × 128 by linear interpolation to ease the comparison. The code for training the Pix2Pix model and the OpenFOAM dictionaries for the CFD simulations can be accessed in Milla-Val (2024a); training and test datasets are provided in Milla-Val (2024b).

In Table 3 the metrics for the test set for both models are presented: the mean absolute errors for speed, direction and both horizontal components and the SSIM are shown for the previous SVM model and for the Pix2Pix one.

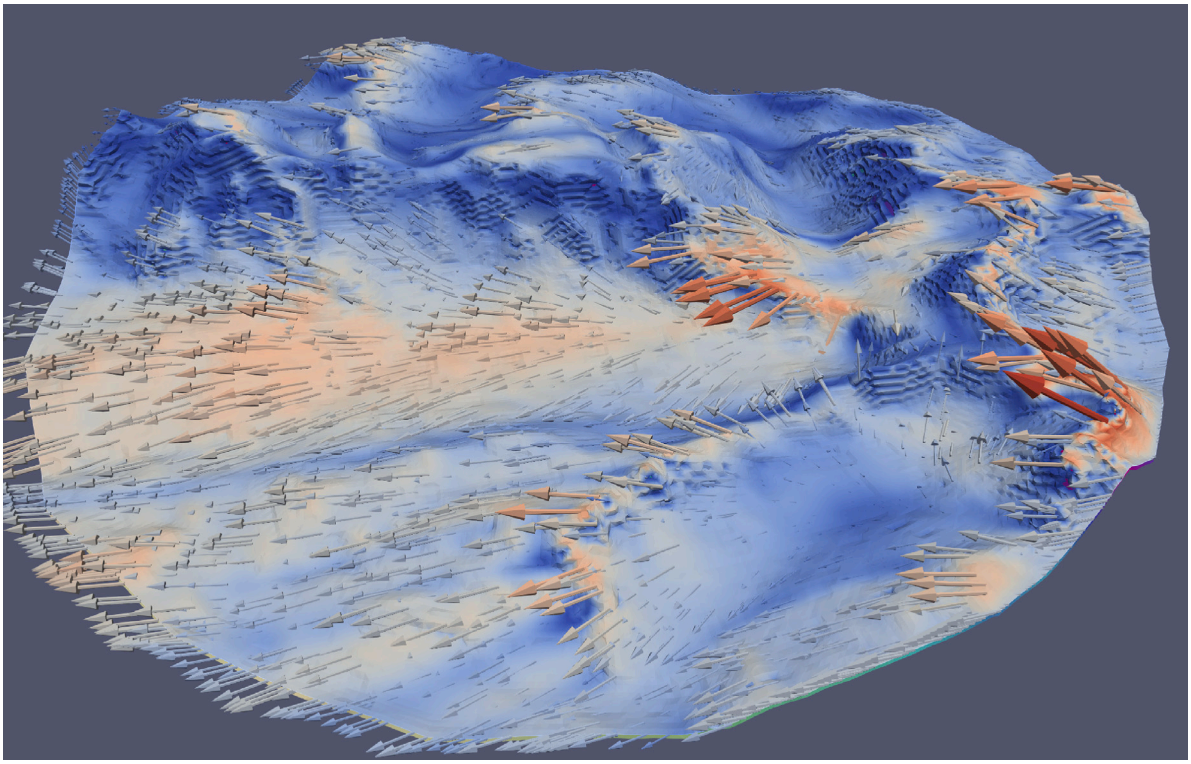
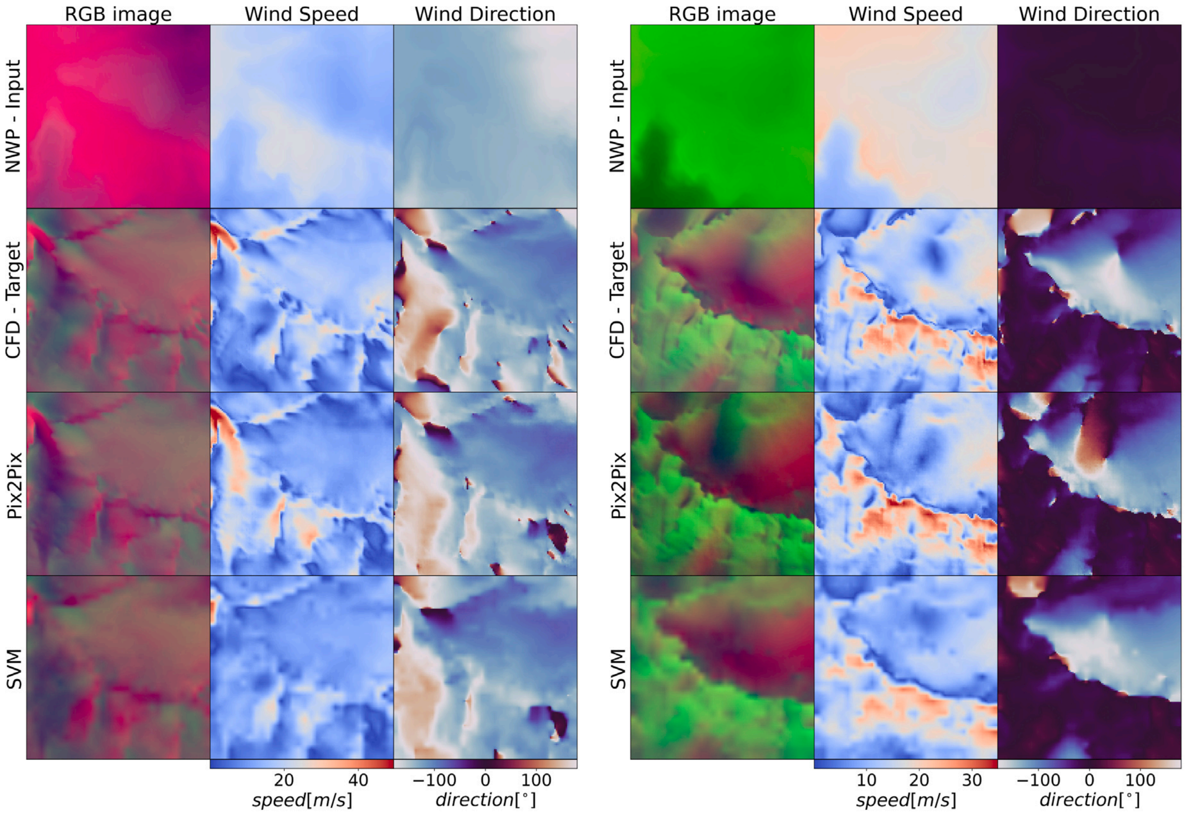


Fig. 5. CFD simulation of one of the weather events, showing the complex terrain and velocity field. The flow field is on a surface above the terrain level, and is a typical “targets” for the cGAN model.



(a) Event date and time: 2018-01-01 at 03:00.

(b) Event date and time: 2018-11-08 at 20:00.

Fig. 6. Typical results for two events from the test dataset. First row, input data (NWP results); second row, the target values (CFD results); third row, results from the present Pix2Pix model; fourth row, results from previous SVM model for comparison. First column: RGB representation of U and V velocity components; second column, wind speed; third column, wind direction (0° is North, 90° is East, −90° is West and −180° and 180° are South).

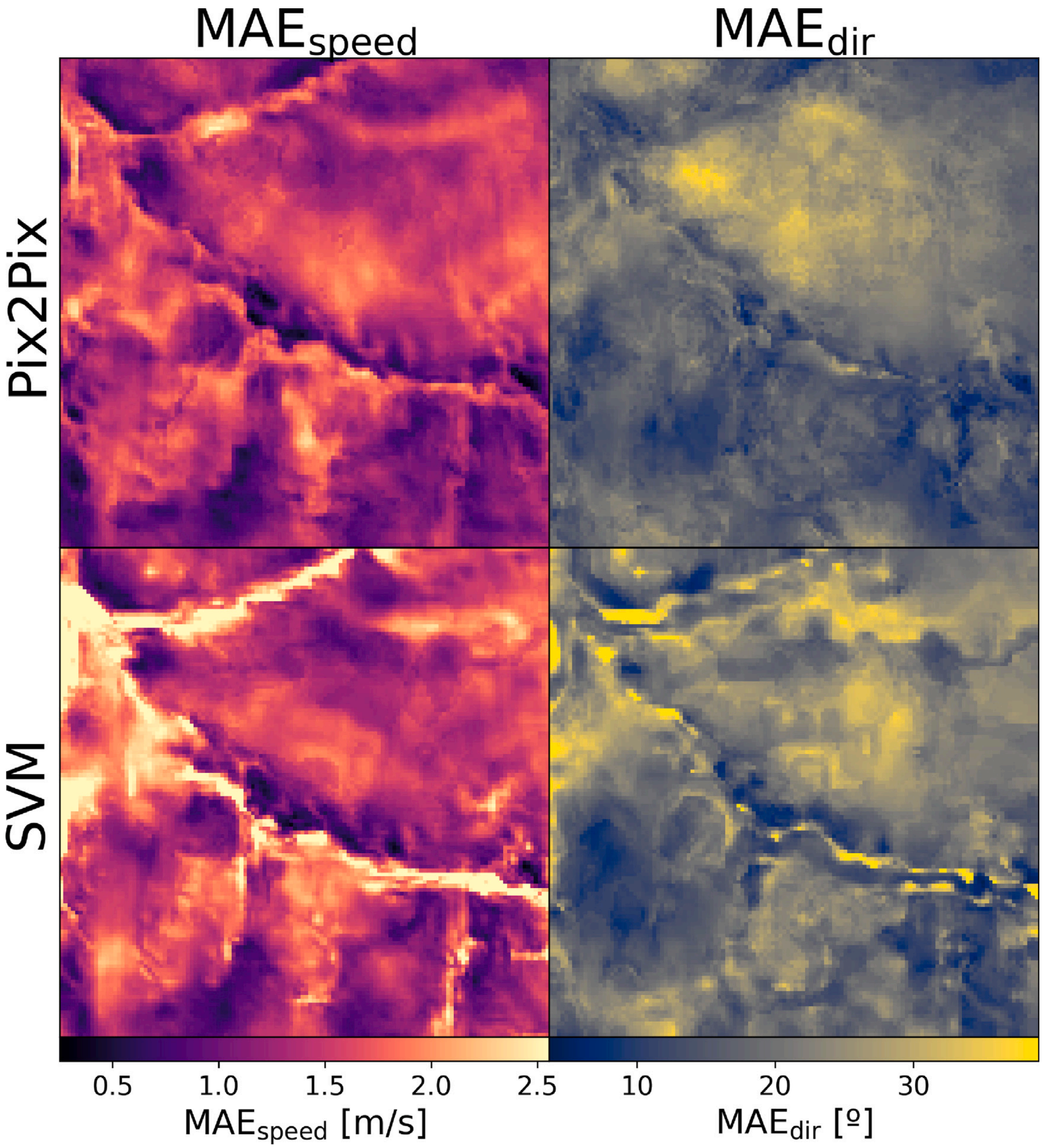


Fig. 7. Spatial $\text{MAE}_{\text{speed}}$ and MAE_{dir} values for SVM and Pix2Pix models averaged over the 80 test events. Colour scales clipped to values in the Pix2Pix model.

Table 3
MAE and SSIM metrics for the Pix2Pix and SVM models.

Metric	SVM	Pix2Pix
MAE_U [m/s]	1.55	1.35
MAE_V [m/s]	1.56	1.36
$\text{MAE}_{\text{speed}}$ [m/s]	1.55	1.36
MAE_{dir} [°]	20.91	18.73
SSIM	0.68	0.78

Fig. 6 shows that the Pix2Pix model is able to generate, downstream of the peaks, wakes that are also observed in the ground-truth CFD results, but do not appear so clearly in the SVM results. In Appendix A, additional test events generated by the Pix2Pix model are presented as supplementary material (Fig. 9).

Fig. 7 presents the spatial distribution of $\text{MAE}_{\text{speed}}$ and MAE_{dir} values for the SVM and Pix2Pix models averaged over all 80 events in the test set.

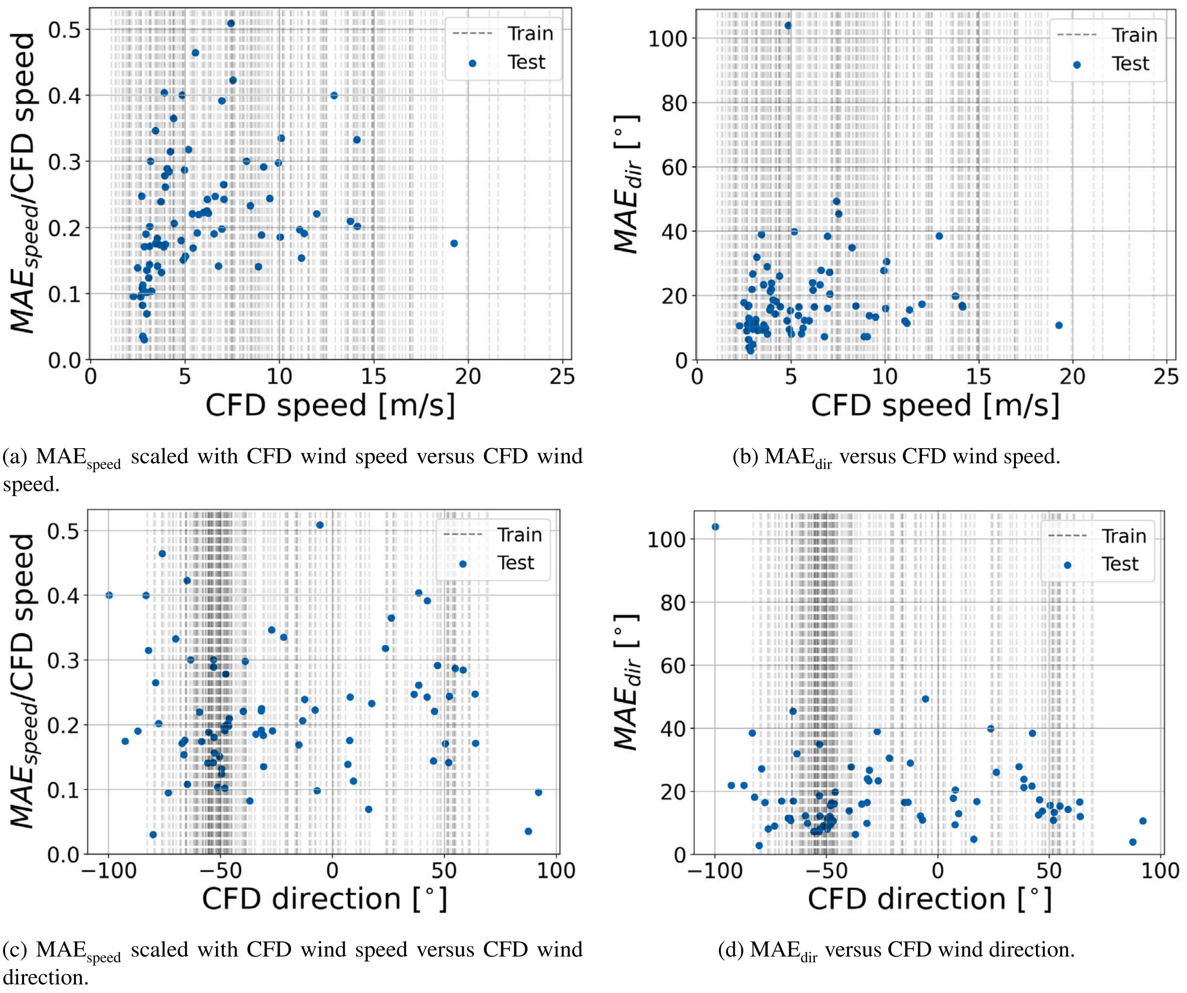
To study a possible correlation between the CFD events and the errors obtained by the Pix2Pix model, the errors, $\text{MAE}_{\text{speed}}$ and MAE_{dir}, for the Pix2Pix model and the CFD speed and direction are averaged for each event of the test dataset and presented in Fig. 8; here $\text{MAE}_{\text{speed}}$, scaled with the CFD speed of the test events, and MAE_{dir} are shown versus the wind speed and direction for the CFD.

Table 4 presents the computational times associated with model building (including NWP simulations, CFD simulations, and model training) as well as model utilisation.

5. Discussion

The improved SSIM (Table 3) indicates that the predicted results have a more CFD-like structure in the new Pix2Pix model than they had in the SVM one. Therefore, the present Pix2Pix model is a major step forward in improving the accuracy of the machine learning predictions.

The patterns that the Pix2Pix model generates (Fig. 6) are more similar to those expected from a CFD simulation compared to the SVM



(a) $\text{MAE}_{\text{speed}}$ scaled with CFD wind speed versus CFD wind speed.

(b) MAE_{dir} versus CFD wind speed.

(c) $\text{MAE}_{\text{speed}}$ scaled with CFD wind speed versus CFD wind direction.

(d) MAE_{dir} versus CFD wind direction.

Fig. 8. MAE from the Pix2Pix model and CFD wind speed and direction averaged for each test event. The grey vertical lines indicate that there is a training event for such CFD wind speed or direction.

Table 4
Computational times for model building and model usage: Times for the simulation of the 273 CFD events, for a whole year of NWP hourly events and for training and data retrieval with the Pix2Pix model. CPU indicates a 6-core Intel i7-6800k; GPU indicates an NVIDIA GeForce RTX 4060 Ti.

	Per event	Total
CFD (CPU)	4 h	1092 h (273 events)
NWP (CPU)	0.25 h	2190 h (8760 events)
Pix2Pix training (GPU)	–	0.5 h
Pix2Pix retrieval (GPU)	<1 s	–

output, as confirmed by the larger SSIM value (Table 3). The SVM model is trained on individual points, without using their neighbouring points. This may be the reason why it performs poorly in preserving spatial correlation and coherence. In contrast, the convolution operations of Pix2Pix are specifically designed to consider surrounding information, which is why it is so successful in capturing spatial dependencies.

It is evident from Fig. 7 that the Pix2Pix model does not accurately match the CFD results in certain areas of the domain; for instance, wind speed errors are generally larger windward of the orographic features, and smaller leeward. Also, a north-central area can be seen where the wind direction errors are significantly larger. This is likely to be caused by the lack of orographic constraints in this open region of the domain, which increases the number of possible outputs from the CFD simulations starting from similar boundary conditions, making it harder for the model to predict the wind direction. Examples of this behaviour can be seen in wake development in this area in Fig. 6.

In Fig. 8 the MAE values for the Pix2Pix model are presented versus the CFD wind speed and direction averaged for the test events. An outlier is spotted for $\text{MAE}_{\text{dir}} \sim 100°$ (Figs. 8(b) and 8(d)) but no counterpart is observed in the $\text{MAE}_{\text{speed}}$ plots (Figs. 8(a), 8(c)); paying special attention to Fig. 8(d), it can be seen that there are almost no training data with the CFD wind direction of ~100° and the lack of training points in this specific area is making this result so adverse. As a future work, with more available computer resources, more CFD simulations could be conducted, increasing the training dataset and solving this scarce populated regions, improving the overall model performance. Despite of this outlier event, there are several more test events in the unpopulated training event region whose performance is in concordance with the rest of test events. This indicates that the Pix2Pix model is able to some extent to performance Apart from that, there is no strong observation that indicates somehow a correlation between CFD wind speed or direction and the outcome of the Pix2Pix model may exist.

Next, we report the computational times for model building (NWP simulations, CFD simulations, and model training) and model usage (Table 4). CPU times are reported as core-hours on a 6-core Intel i7-6800k CPU, while the GPU times (reported for the training and data retrieval operations of the Pix2Pix model) are on an NVIDIA GeForce RTX 4060 Ti. The NWP mesoscale model (which consists of four nested domains) takes 0.25 core-hours for each hourly event and a total of 2190 core-hours (or 91 days) for a full year of 8760 events.

Approximately 4 core-hours are needed to simulate each CFD event, resulting in a total of 1092 h (or 45 days) to complete the simulations for all 273 events in the training and test datasets. The Pix2Pix model takes 30 GPU-min and 1500 epochs for training, and data retrieval takes under 1 s. Compared to the 40 min (on a 6-core machine) required for each CFD simulation the Pix2Pix model has a speed-up factor of four orders of magnitude.

6. Limitations

CFD simulations on this very large scale are computational expensive, and this has had an impact on the amount of training and test events that could be generated with reasonable computational resources. While the results are overall satisfactory, it has been shown that improving the dataset with the addition of events may be beneficial for model performance (Milla-Val et al., 2024); the enlargement of the data set is foreseen as a future activity.

The current rationale of this research line is to translate NWP images into CFD images. These images are 2D surfaces close to the domain terrain and that is why conventional image-to-image models using 2D convolution operations are used. Therefore, only wind information at a given height level is used to train the model; using higher order convolution operations (3D, 4D...), as adopted by video analysis models, would allow to benefit from wind field information at different heights. This in turn would make the model heavier, more expensive to train, and slower at retrieval time but could potentially provide better results.

In this study, the usage of an image-to-image model has been proven successful in a region with complex orography. However, the model is specifically trained with paired data from NWP and CFD simulations of this specific mountainous region. Although more research is needed in this regard, it is unlikely that transfer learning (*i.e.:* a model trained in a mountainous region working in an urban area) can be applied to this paradigm. As seen in the process of generating the model, some outliers appear when the training dataset does not adequately encompass all possible weather conditions; the same may be true when the effect in the flow field of the terrain features is not properly represented in the training dataset. For a new geographical region, there might be means of using a pre-existing model for another region to speed up model building, such as a warm start-up for the training (*i.e.* starting the training from previous model weights instead of random initial values); nevertheless, it is likely that a new model and a region-specific dataset will be required for the methodology to work for a different geographical area. StarGAN (Choi et al., 2018) is a multi-domain image-to-image model that may avoid the inconvenience of having to train a different model, albeit the region-specific dataset would still be needed.

7. Conclusions

We have presented a method based on conditional Generative Adversarial Networks (cGAN), specifically Pix2Pix (Isola et al., 2016), that successfully uses Numerical Weather Prediction (NWP) data to generate high-resolution Computational Fluid Dynamics-like (CFD) results, without incurring the high computational cost of CFD simulations. The method considers the results from NWP and CFD as paired “images”, and attempts to predict the latter from the former.

The method is tested for generating detailed, CFD-like results for a challenging, orographically complex region in the Pyrenees mountain range in Spain. The results show that the proposed Pix2Pix model outperforms other traditional methods machine-learning methods such as Support Vector Machines (SVM) (Milla-Val et al., 2024), providing greater accuracy and better computational efficiency. Mean Average Errors are $\mathrm{MAE}_{\mathrm{speed}} = 1.36$ m/s for wind speed, $\mathrm{MAE}_{\mathrm{dir}} = 18.73^{\circ}$ for wind direction, and the Structural Similarity Index (SSIM) is 0.78 for the reported results. Traditional CFD simulations take nearly one hour to complete for this problem, but the present Pix2Pix model can reduce the computational time by four orders of magnitude.

The proposed methodology has been therefore successful in terms of accuracy and has significantly reduced computational times; however, there are still some areas that are worth exploring. We have used in this work an image-to-image model to “translate” NWP products to CFD-like results; but a similar relationship should exist in the reverse direction, from CFD to NWP. Bidirectional translations between NWP and CFD could be obtained using transfer style models, such as CycleGAN (Zhu et al., 2017). This may potentially further improve the performance of the methodology proposed in this work and help the neural network training process by uncovering the underlying relationship in the opposite direction of this “translation”, *viz* from the CFD domain to the NWP domain. The batch normalisation layer was removed from the proposed cGAN model (while it was present in the original Pix2Pix model) because its removal resulted in better results for the application in this study. Huang and Belongie (2017) and Park et al. (2019) proposed the AdaIN and SPADE paradigms, which may serve as an alternative to improve the performance of the traditional batch normalisation layers by preserving semantic information.

This paper has shown that the proposed approach is suitable for a region with a challenging mountainous landscape and intricate topography. Its usage in urban settings can be highly advantageous in applications that are becoming increasingly important, such as energy production in urban areas or building layout planning (Hao et al., 2023; Kwok and Hu, 2023).

In this work, we are treating the several data snapshots as an ensemble of steady-state simulations. However, the transient nature of wind evolution may be a critical aspect of its modelling. To incorporate these effects, transformers (Vaswani et al., 2017) can be used, as has been done in various CFD applications, such as the generation of synthetic inflow to spatially developing turbulent boundary layers (Yousif et al., 2023), and flow prediction for the design and optimisation of aerodynamic shapes (Jiang et al., 2023a).

CRediT authorship contribution statement

Jaime Milla-Val: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Data curation, Conceptualization. **Carlos Montañés:** Writing – review & editing, Supervision, Methodology, Conceptualization. **Norberto Fueyo:** Writing – review & editing, Supervision, Resources, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the authors used ChatGPT and Writefull occasionally for suggestions to improve the readability of some paragraphs. The authors reviewed any suggested wording and edited it as needed; they take full responsibility for the content of the publication.

Software and data availability

- Input dataset for the NWP software: dataset `ds090.0` (National Centers for Environmental Prediction et al., 1994).
- NWP software: WRF v4.1 (Skamarock et al., 2019)
- CFD software: OpenFOAM v6 (Weller et al., 1998)
- AI suite Python framework: TensorFlow v2.8 (Abadi et al., 2015)
- Training and test datasets: download from pCloud (Milla-Val, 2024b).
- Own code: the code implemented for this work can be accessed at this GitHub repository (Milla-Val, 2024a).

Funding

This work was partly funded by Grant DIN2019-010452 from MCIN/AEI/10.13039/501100011033, Spain; by the Departamento de Ciencia, Universidad y Sociedad del Conocimiento del Gobierno de Aragón, Spain, as Group T32_23R Tecnologías Fluidodinámicas; and by Project TED2021-131861B-I00, financed by MCIN/AEI/10.13039/501100011033 (Spain) and by the European Union "NextGenerationEU"/PRTR.

Declaration of competing interest

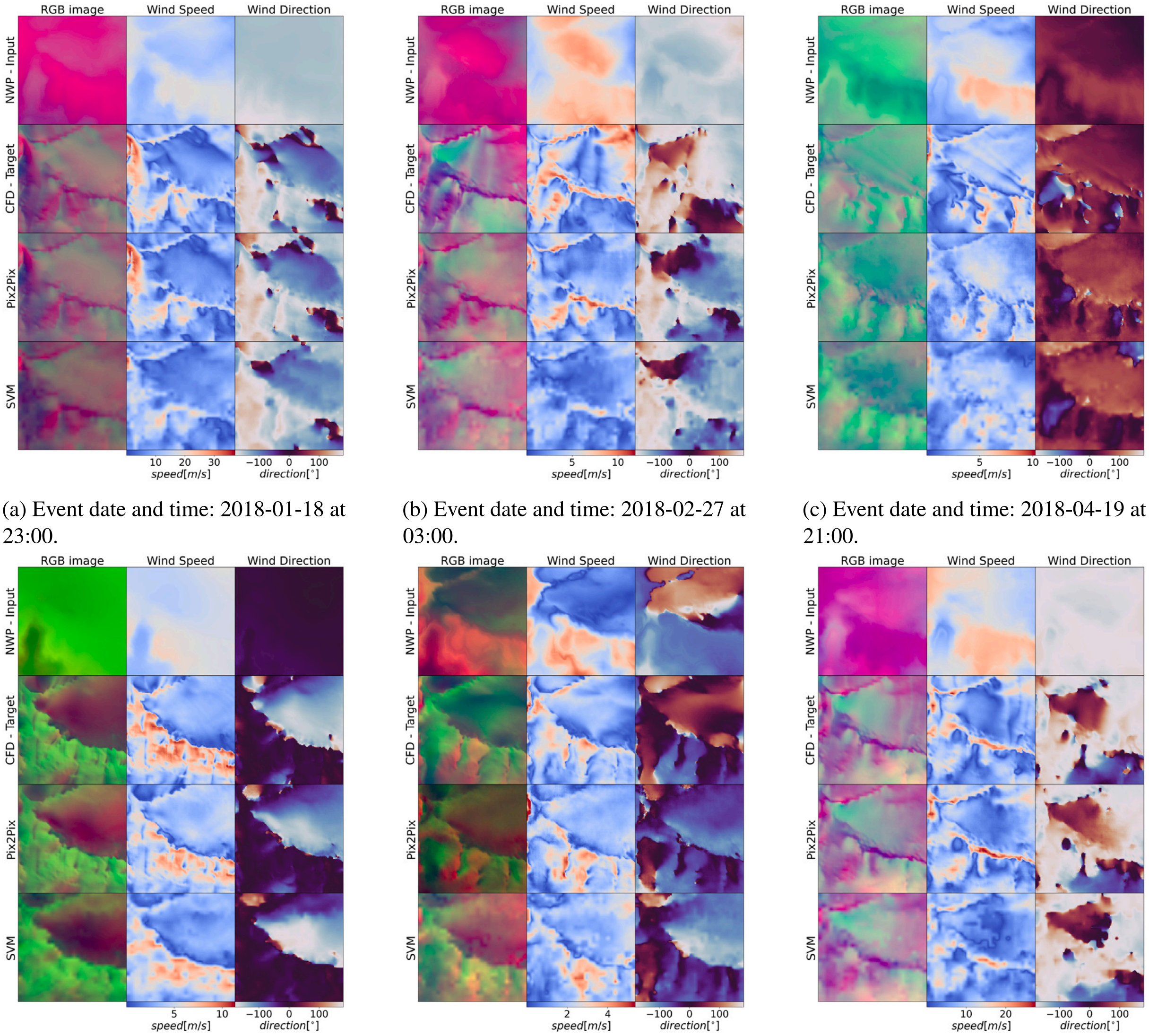
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Additional events from Pix2Pix

Additional events are presented in Fig. 9 to further analyse the results that the proposed Pix2Pix model provides. In addition, SVM results are shown to show the improvement of the present proposal over previous ones. The events in Fig. 9 are part of the 80 test events used to compute metrics and errors as a part of the model performance assessment. The events in the figure are distributed along the test year 2018 and have been selected to include dates in all the four seasons. They show a wide variety of weather conditions with strong and weak wind intensities (colour scale of the middle column) and several wind directions (third column).

Data availability

Data will be made available on request.



(a) Event date and time: 2018-01-18 at 23:00.

(b) Event date and time: 2018-02-27 at 03:00.

(c) Event date and time: 2018-04-19 at 21:00.

(d) Event date and time: 2018-07-31 at 12:00.

(e) Event date and time: 2018-11-07 at 21:00.

(f) Event date and time: 2018-12-10 at 06:00.

Fig. 9. Results for six events from the test dataset. First row, input data (NWP results); second row, the target values (CFD results); third row, results from the present Pix2Pix model; fourth row, results from a previous SVM model for comparison. First column: RGB representation of U and V velocity components; second column, wind speed; third column, wind direction (0° is North, 90° is East, −90° is West and −180° and 180° are South).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: https://www.tensorflow.org/. Software available from tensorflow.org.

Behara, R.K., Saha, A.K., 2024. Analysis of wind characteristics for grid-tied wind turbine generator using incremental generative adversarial network model. IEEE Access 12, 38315–38334. http://dx.doi.org/10.1109/ACCESS.2024.3372862.

Bellman, R., 1966. Dynamic programming. Science 153 (3731), 34–37.

Bishop, C.M., 2006. Sparse kernel machines. In: Pattern Recognition and Machine Learning. Springer, p. 758, chapter 7.

Blocken, B., van der Hout, A., Dekker, J., Weiler, O., 2015. CFD simulation of wind flow over natural complex terrain: Case study with validation by field measurements for Ria de Ferrol, Galicia, Spain. J. Wind Eng. Ind. Aerodyn. 147, 43–57. http://dx.doi.org/10.1016/j.jweia.2015.09.007.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. http://dx.doi.org/10.1023/A:1010933404324.

Chen, J., Chen, S., Wee, L., Dekker, A., Bermejo, I., 2023. Deep learning based unpaired image-to-image translation applications for medical physics: a systematic review. Phys. Med. Biol. 68 (5), 05TR01. http://dx.doi.org/10.1088/1361-6560/acba74.

Chen, Y., Wang, Y., Dong, Z., Su, J., Han, Z., Zhou, D., Zhao, Y., Bao, Y., 2021. 2-D regional short-term wind speed forecast based on CNN-LSTM deep learning model. Energy Convers. Manage. 244, 114451. http://dx.doi.org/10.1016/j.enconman.2021.114451.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., Choo, J., 2018. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. arXiv:1711.09020.

Dalva, Y., Pehlivan, H., Hatipoglu, O.I., Moran, C., Dundar, A., 2023. Image-to-image translation with disentangled latent vectors for face editing. IEEE Trans. Pattern Anal. Mach. Intell. 45 (12), 14777–14788. http://dx.doi.org/10.1109/TPAMI.2023.3308102.

Elgendi, M., AlMallahi, M., Abdelkhalig, A., Selim, M.Y., 2023. A review of wind turbines in complex terrain. Int. J. Thermofluids 17, 100289. http://dx.doi.org/10.1016/j.ijft.2023.100289.

Fang, Z., Tanyas, H., Gorum, T., Dahal, A., Wang, Y., Lombardo, L., 2023. Speech-recognition in landslide predictive modelling: A case for a next generation early warning system. Environ. Model. Softw. 170, 105833. http://dx.doi.org/10.1016/j.envsoft.2023.105833.

Gerber, F., Lehning, M., Hoch, S.W., Mott, R., 2017. A close-ridge small-scale atmospheric flow field and its influence on snow accumulation. J. Geophys. Res.: Atmos. 122 (15), 7737–7754. http://dx.doi.org/10.1002/2016JD026258.

Goldberger, J., Roweis, S., Hinton, G.E., Salakhutdinov, R., 2004. Neighbourhood components analysis. In: Advances in Neural Information Processing Systems. MIT Press, pp. 513–520.

Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial networks. http://dx.doi.org/10.48550/arXiv.1406.2661.

Hao, Y., Yang, W., Yin, K., 2023. Novel wind speed forecasting model based on a deep learning combined strategy in urban energy systems. Expert Syst. Appl. 219, 119636. http://dx.doi.org/10.1016/j.eswa.2023.119636.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. http://dx.doi.org/10.48550/arXiv.1502.01852.

Huang, X., Belongie, S., 2017. Arbitrary style transfer in real-time with adaptive instance normalization. arXiv:1703.06868.

Isola, P., Zhu, J.-Y., Zhou, T., Efros, A.A., 2016. Image-to-image translation with conditional adversarial networks. http://dx.doi.org/10.48550/arXiv.1611.07004.

Jiang, Y., Jiang, L., Yang, S., Loy, C.C., 2023b. Scenimefy: Learning to craft anime scene via semi-supervised image-to-image translation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. ICCV, pp. 7357–7367. http://dx.doi.org/10.48550/arXiv.2308.12968.

Jiang, J., Li, G., Jiang, Y., Zhang, L., Deng, X., 2023a. TransCFD: A transformer-based decoder for flow field prediction. Eng. Appl. Artif. Intell. 123, 106340. http://dx.doi.org/10.1016/j.engappai.2023.106340.

Karras, T., Laine, S., Aila, T., 2019. A style-based generator architecture for generative adversarial networks. arXiv:1812.04948.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. http://dx.doi.org/10.48550/arXiv.1412.6980.

Kwok, K., Hu, G., 2023. Wind energy system for buildings in an urban environment. J. Wind Eng. Ind. Aerodyn. 234, 105349. http://dx.doi.org/10.1016/j.jweia.2023.105349.

Li, H., Yang, Q., Li, T., 2024. Wind turbine wake prediction modelling based on transformer-mixed conditional generative adversarial network. Energy 291, 130403. http://dx.doi.org/10.1016/j.energy.2024.130403.

Liang, P., Deng, C., Yuan, X., Zhang, L., 2023. A deep capsule neural network with data augmentation generative adversarial networks for single and simultaneous fault diagnosis of wind turbine gearbox. ISA Trans. 135, 462–475. http://dx.doi.org/10.1016/j.isatra.2022.10.008.

López, G., Arboleya, P., 2022. Short-term wind speed forecasting over complex terrain using linear regression models and multivariable LSTM and NARX networks in the Andes Mountains, Ecuador. Renew. Energy 183, 351–368. http://dx.doi.org/10.1016/j.renene.2021.10.070.

Maas, A.L., Hannun, A.Y., Ng, A.Y., et al., 2013. Rectifier nonlinearities improve neural network acoustic models. In: International Conference on Machine Learning. p. 3.

Milla-Val, J., 2024a. image2image-nwp2cfd-piri. https://github.com/nabladot/image2image-nwp2cfd-piri.git. (Accessed 06 August 2024).

Milla-Val, J., 2024b. Training dataset: dataPiri1h_resMean. https://e.pcloud.link/publink/show?code=XZh6isZnu95fxzeqHFP7Qxlc7crtYdp2CJk. (Accessed 06 August 2024).

Milla-Val, J., Montañés, C., Fueyo, N., 2024. Economical microscale predictions of wind over complex terrain from mesoscale simulations using machine learning. Model. Earth Syst. Environ. 10 (1), 1407–1421. http://dx.doi.org/10.1007/s40808-023-01851-x.

Mirza, M., Osindero, S., 2014. Conditional generative adversarial nets. http://dx.doi.org/10.48550/arXiv.1411.1784.

Mughal, M.O., Lynch, M., Yu, F., McGann, B., Jeanneret, F., Sutton, J., 2017. Wind modelling, validation and sensitivity study using Weather Research and Forecasting model in complex terrain. Environ. Model. Softw. 90, 107–125. http://dx.doi.org/10.1016/j.envsoft.2017.01.009.

Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning. ICML-10, pp. 807–814.

National Centers for Environmental Prediction, National Weather Service, NOAA, U.S. Department of Commerce, 1994. NCEP/NCAR Global Reanalysis Products, 1948-continuing. URL: https://rda.ucar.edu/datasets/ds090.0/.

Park, T., Liu, M.-Y., Wang, T.-C., Zhu, J.-Y., 2019. Semantic image synthesis with spatially-adaptive normalization. arXiv:1903.07291.

Prósper, M.A., Otero-Casal, C., Fernández, F.C., Miguez-Macho, G., 2019. Wind power forecasting for a real onshore wind farm on complex terrain using WRF high resolution simulations. Renew. Energy 135, 674–686. http://dx.doi.org/10.1016/j.renene.2018.12.047.

Qian, F., Guo, J., Sun, T., Wang, T., 2014. Multi-scale SSIM metric based on weighted wavelet decomposition. Optik 125 (20), 6205–6209. http://dx.doi.org/10.1016/j.ijleo.2014.06.134.

Robbins, H., Monro, S., 1951. A stochastic approximation method. Ann. Math. Stat. 22 (3), 400–407. http://dx.doi.org/10.1214/aoms/1177729586.

Romero, D.A., Hasanpoor, S., Antonini, E.G.A., Amon, C.H., 2024. Predicting wind farm wake losses with deep convolutional hierarchical encoder–decoder neural networks. APL Mach. Learn. 2 (1), 016111. http://dx.doi.org/10.1063/5.0168973.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-net: Convolutional networks for biomedical image segmentation. http://dx.doi.org/10.48550/arXiv.1505.04597.

Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., Norouzi, M., 2022. Palette: Image-to-image diffusion models. In: ACM SIGGRAPH 2022 Conference Proceedings. SIGGRAPH '22, Association for Computing Machinery, New York, NY, USA, pp. 1–10. http://dx.doi.org/10.1145/3528233.3530757.

Santos, G.B., Pantaleão, A.V., Salviano, L.O., 2023. Using deep generative adversarial network to explore novel airfoil designs for vertical-axis wind turbines. Energy Convers. Manage. 282, 116849. http://dx.doi.org/10.1016/j.enconman.2023.116849.

Skamarock, W., Klemp, J., Dudhia, J., Gill, D., Zhiquan, L., Berner, J., Wang, W., Powers, J., Duda, M.G., Barker, D.M., Huang, X.-Y., 2019. A Description of the Advanced Research WRF Model Version 4. NCAR Technical Note NCAR/TN-475+STR, p. 145. http://dx.doi.org/10.5065/1dfh-6p97.

Stefenon, S.F., Ribeiro, M.H.D.M., Nied, A., Mariani, V.C., Coelho, L.D.S., Leithardt, V.R.Q., Silva, L.A., Seman, L.O., 2021. Hybrid wavelet stacking ensemble model for insulators contamination forecasting. IEEE Access 9, 66387–66397. http://dx.doi.org/10.1109/ACCESS.2021.3076410.

Sun, B., Mei, Y., Yan, N., Chen, Y., 2023. UMGAN: Underwater image enhancement network for unpaired image-to-image translation. J. Mar. Sci. Eng. 11 (2), http://dx.doi.org/10.3390/jmse11020447.

Sutherland, D., Rashid, M.A., Hilton, J.E., Moinuddin, K.A., 2023. Implementation of spatially-varying wind adjustment factor for wildfire simulations. Environ. Model. Softw. 163, 105660. http://dx.doi.org/10.1016/j.envsoft.2023.105660.

Temel, O., Bricteux, L., van Beeck, J., 2018. Coupled WRF-OpenFOAM study of wind flow over complex terrain. J. Wind Eng. Ind. Aerodyn. 174, 152–169. http://dx.doi.org/10.1016/j.jweia.2018.01.002.

Tyleček, R., Šára, R., 2013. Spatial pattern templates for recognition of objects with regular structure. In: Weickert, J., Hein, M., Schiele, B. (Eds.), Pattern Recognition. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 364–374. http://dx.doi.org/10.1007/978-3-642-40602-7_39.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is all you need. arXiv:1706.03762.

Vinuesa, R., Brunton, S.L., 2022. Enhancing computational fluid dynamics with machine learning. Nat. Comput. Sci. 2 (6), 358–366. http://dx.doi.org/10.1038/s43588-022-00264-7.

Vionnet, V., Marsh, C.B., Menounos, B., Gascoin, S., Wayand, N.E., Shea, J., Mukherjee, K., Pomeroy, J.W., 2021. Multi-scale snowdrift-permitting modelling of mountain snowpack. Cryosphere 15 (2), 743–769. http://dx.doi.org/10.5194/tc-15-743-2021.

Wakes, S.J., Maegli, T., Dickinson, K.J., Hilton, M.J., 2010. Numerical modelling of wind flow over a complex topography. Environ. Model. Softw. 25 (2), 237–247. http://dx.doi.org/10.1016/j.envsoft.2009.08.003.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13 (4), 600–612. http://dx.doi.org/10.1109/TIP.2003.819861.

Weller, H.G., Tabor, G., Jasak, H., Fureby, C., 1998. A tensorial approach to computational continuum mechanics using object-oriented techniques. Comput. Phys. 12 (6), 620. http://dx.doi.org/10.1063/1.168744.

Yang, X., Zhang, Y., Lv, W., Wang, D., 2021. Image recognition of wind turbine blade damage based on a deep learning model with transfer learning and an ensemble learning classifier. Renew. Energy 163, 386–397. http://dx.doi.org/10.1016/j.renene.2020.08.125.

Yang, S., Zhou, Y., Chen, X., Deng, C., Li, C., 2023. Fault diagnosis of wind turbines with generative adversarial network-based oversampling method. Meas. Sci. Technol. 34 (4), 044004. http://dx.doi.org/10.1088/1361-6501/acad20.

Ye, L., Peng, Y., Li, Y., Li, Z., 2024. A novel informer-time-series generative adversarial networks for day-ahead scenario generation of wind power. Appl. Energy 364, 123182. http://dx.doi.org/10.1016/j.apenergy.2024.123182.

Yousif, M.Z., Zhang, M., Yu, L., Vinuesa, R., Lim, H., 2023. A transformer-based synthetic-inflow generator for spatially developing turbulent boundary layers. J. Fluid Mech. 957, A6. http://dx.doi.org/10.1017/jfm.2022.1088.

Zhang, X., Li, D., Fu, X., 2024. A novel wasserstein generative adversarial network for stochastic wind power output scenario generation. IET Renew. Power Gener. http://dx.doi.org/10.1049/rpg2.12932.

Zhang, J., Zhao, X., 2022. Wind farm wake modeling based on deep convolutional conditional generative adversarial network. Energy 238, 121747. http://dx.doi.org/10.1016/j.energy.2021.121747.

Zhu, J.-Y., Park, T., Isola, P., Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. http://dx.doi.org/10.48550/arXiv.1703.10593.