



# Wind turbine wake detection and characterisation utilising blade loads and SCADA data: a generalised approach

Piotr Fojcik<sup>1</sup>, Edward Hart<sup>1</sup>, and Emil Hedevang<sup>2</sup>

<sup>1</sup>Electronic & Electrical Engineering, University of Strathclyde,  
99 George Street, Glasgow, G1 1RD, United Kingdom

<sup>2</sup>Siemens Gamesa Renewable Energy, Borupvej 16, 7330 Brande, Denmark

**Correspondence:** Piotr Fojcik (piotr.fojcik@strath.ac.uk)

Received: 31 January 2025 – Discussion started: 12 February 2025

Revised: 1 June 2025 – Accepted: 24 June 2025 – Published: 11 September 2025

**Abstract.** Large offshore wind farms face operational challenges due to turbine wakes, which can reduce energy yield and increase structural fatigue. These problems may be mitigated through wind farm flow control techniques, which require reliable wake detection (recognising the presence of a clear wake) and characterisation (parametric description of a wake's properties) as prerequisites. This paper presents a novel three-stage framework for generalised wake detection and characterisation. First, a regression model utilises blade loads, pitch and rotor rotational speed data to estimate the wind speed distribution across the rotor plane. Second, a convolutional neural network undertakes pattern recognition analysis to perform wake detection, classifying rotor-plane wind estimates as “fully impinged”, “left impinged”, “right impinged” or “not impinged”. Third, where wake impingement is detected, two-dimensional Gaussian fitting is undertaken to provide a parametric wake characterisation, providing outputs of the wake centre location and wake lateral width. The framework is trained and tested in a simulation environment incorporating the Mann turbulence model, the dynamic wake meandering (DWM) model for generating wakes and an industrial-grade aeroelastic solver. The testing is undertaken under a wide range of wind conditions, with mean ambient wind speeds from 5–15 m s<sup>-1</sup>, turbulence intensities from 3 %–9 % and a full range of wind directions. Results show high accuracy of wind field estimation, with the mean RMSE across all test cases being 3.7 % when normalised by mean ambient wind speed. The wake detection model accurately identifies the presence of a wake for approximately 77 % of tested wind fields, with subpar performance observed for more extreme conditions or those at the limits of the training data used. The final wake characterisation stage is shown to flexibly adapt to changing wind conditions, successfully tracking the wake's position even for more turbulent conditions. The proposed framework therefore demonstrates strong potential as a generalised approach to wake detection and characterisation.

## 1 Introduction

The wind industry has advanced significantly in recent years, reaching a total worldwide installed capacity of 906 GW at the end of 2022 (Hutchinson and Zhao, 2023). Despite being a mature field, some challenges still remain to be solved. Due to the increasing size of modern wind farms, there is a newly posed challenge of effectively operating clusters of hundreds of multi-megawatt machines. For this reason, the current fo-

cus for both research and industry is shifting towards a farm-level approach for operations, control and maintenance.

A wind turbine interacts with the incoming wind flow, extracting energy and creating a wake: a downstream region of decreased wind speed and increased turbulence. Due to their varied impact on the farm's performance, wakes play a crucial role in turbine operation. A turbine affected by wakes from upstream machines has a lower energy yield due to a wake-generated velocity deficit in the upcoming flow

(Adaramola and Krogstad, 2011; Barthelmie et al., 2010). Having the rotor experience a more turbulent wind field leads to uneven aerodynamic loading on the machine, inducing severe fatigue cycles in its structure (Churchfield et al., 2012). A turbine under waked flow conditions is also prone to experiencing additional loads due to large-scale motions of wake meandering, further enhancing its fatigue (Madsen et al., 2010; Larsen et al., 2013).

Via various control techniques it is possible to mitigate the negatives of waked flow conditions, optimising turbine operation for reliable and effective performance. The earliest developments in this area likely date back to a study by Steinbuch et al. (1988), where the authors implement axial induction control to increase the energy yield by reducing wake effects. Numerous methods for optimising wind farm flow follow; some examples include wake steering by introducing yaw offsets (Howland et al., 2020; SGRE, 2019) or dynamic individual pitch control (Frederik et al., 2020) and dynamic induction control (Munters and Meyers, 2018) to induce enhanced mixing in the wake.

To use the above-mentioned approaches in a closed-loop control scheme, dynamic information on whether the impinging wake is being successfully redirected or dispersed is required (Raach et al., 2016). Moreover, before starting the control action, there first needs to be confirmation that a turbine is indeed wake-affected to facilitate an intervention to its normal operating cycle. For the sake of the discussion in this work, we refer to these two flow control prerequisites as *wake detection* and *wake characterisation*. The former refers to the action of recognising that a given turbine is experiencing a clear wake impingement from a nearby machine, in such a way that it has a significant (in both magnitude and time) influence on its performance. The latter refers to the action of identifying the properties of the wake using a simplified parametric representation and monitoring how the values of these parameters change in time.

There is a growing effort in the research community to develop new wake detection and characterisation techniques. Up-to-date approaches focus on implementing remote sensing techniques or using a turbine's operational data. The former is usually achieved with the use of light detection and ranging (lidar) devices, with recent literature featuring numerous examples of their successful application in wake characterisation (Bingöl et al., 2010; Trujillo et al., 2011; Conti et al., 2020; Lio et al., 2020). A recent study (Raach et al., 2016) investigates the use of lidar-based wake characterisation directly in a closed-loop wake steering scheme, showing promising results. Despite achieving good performance, the lidar-based approaches unavoidably rely on additional hardware, which is not currently present at most wind farms and represents a significant additional cost. A different way of tackling the problem is to employ the relationship between the wind field and turbine response. The basic idea behind this approach is that a variation in the incoming flow can be correlated with changes in signals such as blade

root bending moments and generator speed. The widely discussed method introduced by Bottasso and Schreiber (2018) relies on using out-of-plane blade root bending moments to approximate the local wind speed in each of the rotor quadrants, thus allowing for partial wake detection. The method is tested in the field (Schreiber et al., 2020) using a setup of two wind turbines and a met mast, providing a proof of concept for the qualitative wake detection. The idea of mapping between the blade loads and local wind speeds has also been used by others; some examples include studies by Kim et al. (2023), Simley and Pao (2016), and Liu et al. (2021). A study by Onnen et al. (2022) pushes this approach further, implementing out-of-plane blade loads to achieve accurate characterisation of the meandering wake deficit with the use of an extended Kalman filter (EKF). The authors also discuss the potential of using the estimation uncertainty metric to detect the wake presence. The method is validated via wind tunnel (Onnen et al., 2023) and field (Onnen et al., 2025) experiments, showing a strong correlation between the estimated wake position and the reference lidar measurements. A similar blade-load-based approach is also considered by Dong et al. (2021), where the authors investigate wake characterisation with an EKF setup within a simulation environment. In another study by Farrell et al. (2022), the authors implement a recurrent neural network trained with experimental and simulation data to estimate the lateral position of the wake centre.

All in all, we identify the research gap as follows: existing wake estimation methods primarily focus on the wake characterisation aspect, with both lidar- and load-based approaches offering high accuracy. Up-to-date wake detection studies have analysed wake impingement in a scenario with a single upwind turbine at a fixed distance, for a fixed set of wind directions. This handles only part of the overall problem in a realistic setting, where the turbine experiences a full range of wind directions. For the majority of inflow angles, the turbine is not directly affected by a nearby turbine's wake; it does however receive a large amount of random flow fluctuations due to operating in the highly turbulent atmospheric boundary layer. Without robust knowledge of whether the turbine operates under wake impingement or not, there is a risk of implementing a wake characterisation scheme on a naturally occurring turbulent eddy, hence providing incorrect and misleading inputs to the control system.

In this paper, we seek to bridge the above-identified gap by proposing a generalised three-stage approach that performs wind field reconstruction, wake detection and wake characterisation within a single framework. Firstly, the instantaneous wind field interacting with the rotor is estimated using turbine operational and load data. Secondly, a neural network approach is implemented to classify whether the estimated wind field represents a case of “full”, “partial” or “no” wake impingement. Thirdly, if clear wake impingement is detected, a parametric wake model is fitted in order to characterise the centre and lateral width of the cur-

rently observed wake. A novel end-to-end methodology is presented, which aims to provide both a demonstrator and a performance benchmark for generalised wake detection and characterisation methods of this type. The trained models are tested across a full range of wind directions within a virtual offshore wind farm, evaluating the method's performance for wakes shed at various upstream distances. The simulation environment incorporates Mann turbulence wind boxes, the dynamic wake meandering (DWM) model for generating wake interactions and aeroelastic code to compute the turbine response. This work focuses on developing a novel solution that can confidently assert when a turbine is impinged by a wake from a nearby machine – a key factor for farm-wide wake steering control.

The paper is structured as follows: Sect. 2 discusses the methodology used for the framework's development and training; Sect. 3 presents the results of the framework validation; Sect. 4 considers the discussion on the framework applicability and limitations; and, lastly, Sect. 5 covers the conclusion and addresses future work.

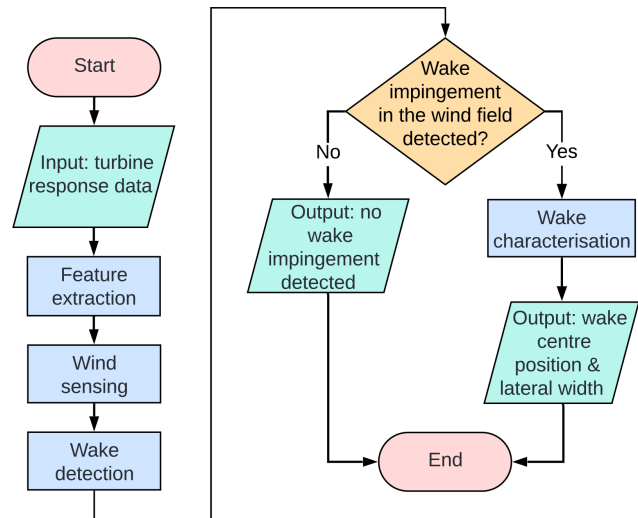
## 2 Methodology

Section 2.1 discusses the layout of the developed framework and its high-level characteristics. The details of the implementation are described in Sect. 2.2 to 2.5.

### 2.1 A summary of the proposed framework

Figure 1 shows the information flow between the models within the proposed wake estimation framework. Figure 2 presents two examples that illustrate the framework's performance when (a) the wind field does not have a clear wake impingement from a nearby turbine and (b) when it does. The high-level details on the individual models that the framework consists of are discussed in the following.

The first constituent model in the developed framework is a wind field estimator, which, when provided with turbine response time series as the input, is capable of producing the wind field reconstruction, thus showing the flow at the rotor as the output. This process is hereafter referred to as *wind sensing*. A successfully generated wind field estimation provides sufficient information to analyse the flow field and extract the desired wake impingement information. For this framework, we propose performing wake detection using a convolutional neural network (CNN), which, after being trained, can distinguish flow with a well-defined wake deficit from a nearby turbine. If a likely wake impingement is detected, a suitable algorithm aiming to estimate the wake's properties is to be implemented. The ultimate framework output data are two time series of the wake centre position and wake lateral width, describing the horizontal distance where the degree of wake impingement is significant. The combination of these two time series provides a simplified two-dimensional representation of a meandering wake deficit.

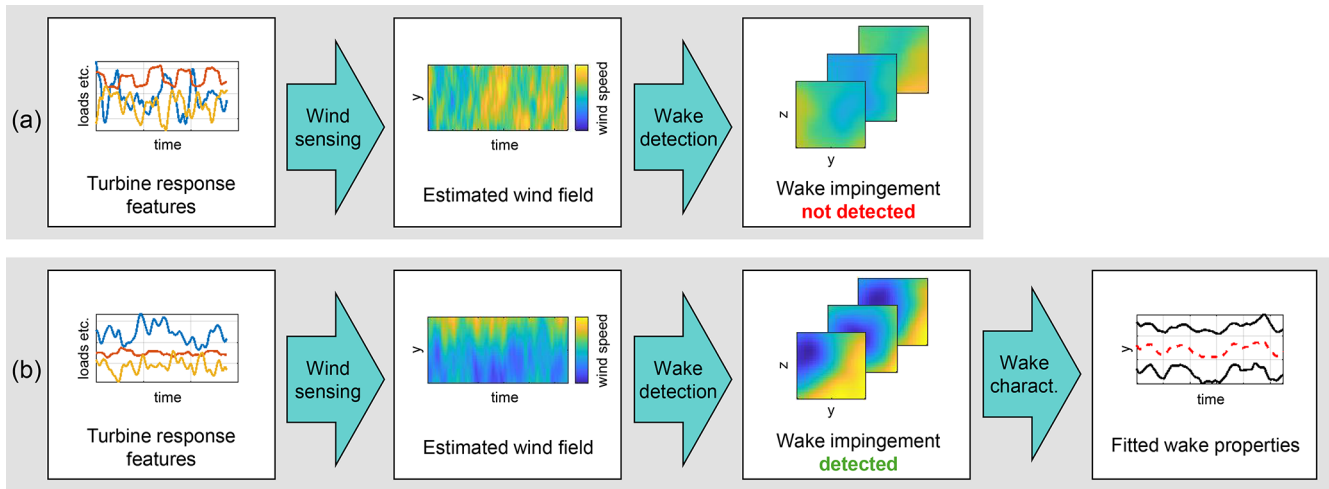


**Figure 1.** A flowchart describing the proposed wake estimation framework.

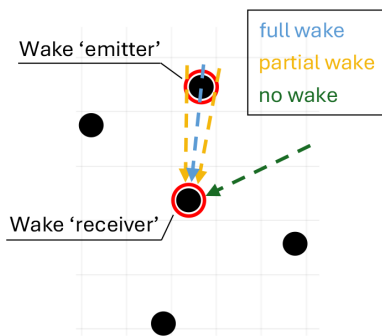
This representation can serve as an input to wind farm flow control techniques that are informed by the lateral properties of the impinging wake, such as those that use yaw as a control action (Howland et al., 2020; SGRE, 2019). Focusing specifically on the lateral properties is also motivated by the fact that the wake meandering is more pronounced in the transverse direction than in the vertical direction, with the ratio reflecting the turbulence intensities of the respective components (Yang and Sotiropoulos, 2019). The developed methodology also allows us to extract a time series of a peak wake deficit  $\Delta U$  value; however this is not discussed in this work.

### 2.2 Framework implementation: training data acquisition

The simulation environment allows the modelling of the operation of a given turbine at any location within the wind farm layout, capturing wake impingement effects from multiple turbines simultaneously. While this aspect plays a key role during testing, training is limited to interactions between two devices located on the farm border: a first-row wake “emitter” and a second-row wake “receiver”, spaced approximately 6 rotor diameters apart. The role of the “receiver” is to provide the turbine response data for training. Four wind directions, representing four distinctive wake impingement scenarios, as seen from the front of the rotor, are defined as follows: (a) fully impinged, with the turbine directly downstream of the closest neighbour; (b) partially impinged, with the wake impacting the left side of the rotor; (c) partially impinged, with the wake impacting the right side of the rotor; and (d) no impingement. The wind direction differs by  $5^\circ$  between the fully and partially impinged cases. This setup allows us to clearly differentiate between the effects of full and partial wake impingement and standalone atmospheric



**Figure 2.** Two examples showing the framework’s performance. **(a)** The incoming wind field does not have a clear wake deficit; thus the framework does not go into the wake characterisation. **(b)** There is a clear wake deficit; thus the final step of wake characterisation is implemented.



**Figure 3.** Simulated wind farm with highlighted locations of two turbines of interest and selected wind directions for the training process.

turbulence. It ensures that only one wake impacts the “re-ceiver” turbine; including multiple wakes would complicate the impingement effects and make it more challenging to define wake detection classes (see Sect. 2.4). The partial layout of the wind farm and the wind directions defining the four classes are shown in Fig. 3.

A dataset comprised of 1200 simulations is produced, with simulation parameters and run counts presented in Table 1. Ambient wind speed  $U_{\text{amb}}$  and ambient turbulence intensity  $I_{\text{amb}}$  values are varied to ensure that the simulated conditions represent a multitude of different offshore wind states. Each simulation consists of generating a unique wind field and calculating the corresponding time series of instantaneous turbine response. Synthetic wind fields are acquired by generating ambient turbulence boxes using the Mann spectral tensor model (Mann, 1994) and superimposing them with wakes generated with the SGRE’s in-house implementation of the DWM model (Larsen et al., 2007). The version of the model

**Table 1.** Training simulation subsets. For each wind direction, 300 simulations capturing all  $U_{\text{amb}}$  and  $I_{\text{amb}}$  combinations are performed.

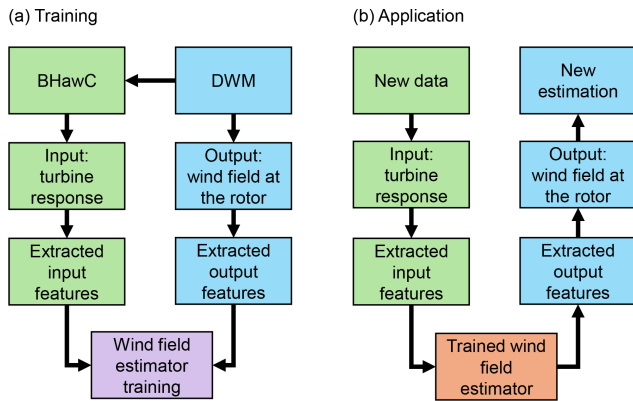
Wind direction [°]	Impingement case	$U_{\text{amb}}$ [m s <sup>-1</sup> ]	$I_{\text{amb}}$ [%]	No. of simulations			
9 4 14 45	Full Partial left Partial right None	5–25 (step size 0.2)	3, 5, 7	300			
Total: 1200							

used follows the parameterisation suggested in the latest IEC standard (IEC, 2019). In its essence, the DWM model regards the wake as a passive tracer being moved laterally and vertically by stochastic turbulence with a characteristic length of twice the rotor diameter or larger. The turbine operation for the simulated conditions is calculated using the aeroelastic code BHawC. Since ambient wind fields are acquired with the Mann turbulence model, the atmospheric conditions are assumed to be neutral (Mann, 1994). The presence of vertical wind shear is simulated using the wind profile power law, with a shear exponent value of 0.07 selected for all cases. Turbines are simulated with no yaw angle misalignment.

### 2.3 Framework implementation: wind sensing

#### 2.3.1 Wind sensing input and output data selection

Figure 4 illustrates the training of the wind sensing model and demonstrates its application scheme. By providing a varied dataset of turbine response time series and the corresponding wind field representations, the estimator “learns”



**Figure 4.** Simplified diagram visualising the flow of data during (a) training and (b) application of the wind sensing model.

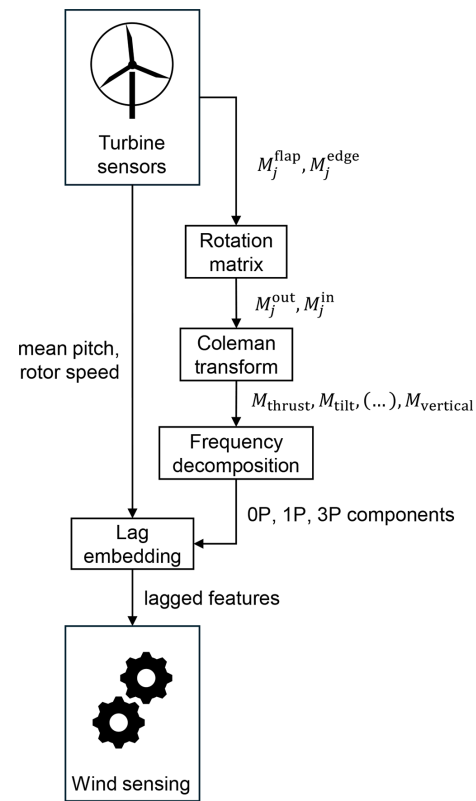
to approximate the wind speed distribution at the rotor from the turbine sensor data.

The aeroelastic code provides access to an extensive set of turbine response signals, such as component accelerations or generator speed. A conditional dependence analysis (readers further interested in the specific method used are referred to Azadkia and Chatterjee, 2021) is undertaken to determine which data channels are best coupled with the variations in the flow. The algorithm tests different combinations of turbine signals, estimating the likelihood that wind speed can be deterministically predicted from them. The following combination with the strongest overall correlation to wind speed is identified and used in wind sensing: blade root bending moments in both the flap-wise and the edge-wise directions; mean pitch across all three blades, simply calculated as  $\bar{\beta} = (\beta_1 + \beta_2 + \beta_3)/3$ ; and rotor rotational speed  $\omega$ . Rotor azimuth angle and individual pitch angles are also extracted to use for transformations of other inputs.

The wind field is considered to be a three-dimensional Cartesian grid, with three axes:  $X$ , representing streamwise distance;  $Y$ , representing lateral distance; and  $Z$ , representing vertical distance. To represent the wind speed distribution in the flow, each point on that grid stores the local values of three wind speed components,  $U$ ,  $V$  and  $W$ , which correspond to the spatial variation in the  $X$ ,  $Y$  and  $Z$  axes, respectively. For this specific application, the wind field is represented with the spatial distribution of the longitudinal wind speed component  $U$ , as it is the one that is primarily impacted by wake deficit (Dimitrov et al., 2017). All models treat the wind field as a time series of  $YZ$  slices, showing the chronological sequence of  $U$  distributions at the rotor plane.

### 2.3.2 Wind sensing input processing

Selected turbine response time series are processed to extract the optimal features from the raw data. The processes described in this section are applied during both training and performance testing – they are an integral part of the



**Figure 5.** Block diagram of the turbine response processing.  $M_j^{\text{flap/edge}}$  – flap/edge blade root bending moments,  $M_j^{\text{out/in}}$  – in- and out-of-plane blade root bending moments,  $j$  – blade index,  $M_{\text{thrust/tilt}/(\dots)/\text{vertical}}$  – rotor loads. A detailed description can be found in the text.

data “pipeline”. Figure 5 illustrates the transformations taking place in the feature extraction.

First, using the pitch signals, the edge- and flap-wise blade root bending moments are transformed to in- and out-of-plane coordinates. Performing the rotation between the two frames is calculated for a given blade  $j$  as follows:

$$\begin{bmatrix} M_j^{\text{out}} \\ M_j^{\text{in}} \end{bmatrix} = \begin{bmatrix} \cos(\beta_j) & \sin(\beta_j) \\ -\sin(\beta_j) & \cos(\beta_j) \end{bmatrix} \begin{bmatrix} M_j^{\text{flap}} \\ M_j^{\text{edge}} \end{bmatrix}, \quad (1)$$

where  $M_j^{\text{out}}$  and  $M_j^{\text{in}}$  are out-of- and in-plane root bending moments, respectively;  $M_j^{\text{flap}}$  and  $M_j^{\text{edge}}$  are flap and edge root bending moments, respectively; and  $\beta_j$  is the instantaneous pitch angle.

Next, using the azimuth angle time series, the Coleman transformation (Coleman, 1943) is applied to the blade load data. It allows us to effectively map the separate, rotary-frame bending moments from each blade onto a stationary frame of reference. As a result, the dynamic response to the variation in the wind can be expressed with six variables describing the loads for an entire rotor:  $M_{\text{thrust}}$ ,  $M_{\text{tilt}}$  and  $M_{\text{yaw}}$  (for out-of-rotor-plane movement) and  $M_{\text{torque}}$ ,  $M_{\text{lateral}}$  and



$M_{\text{vertical}}$  (for in-rotor-plane movement). These are referred to as *rotor loads*. The Coleman transformation is defined as follows:

$$\mathbf{C}(\psi) = \begin{bmatrix} 1/3 & 0 & 0 \\ 0 & 2/3 & 0 \\ 0 & 0 & 2/3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1 \\ \cos(\psi) & \cos\left(\psi + \frac{2\pi}{3}\right) & \cos\left(\psi + \frac{4\pi}{3}\right) \\ \sin(\psi) & \sin\left(\psi + \frac{2\pi}{3}\right) & \sin\left(\psi + \frac{4\pi}{3}\right) \end{bmatrix}, \quad (2)$$

where  $\psi$  is the rotor azimuth angle. It is used to transform the blade loads as follows:

$$\begin{bmatrix} M_{\text{thrust}} \\ M_{\text{tilt}} \\ M_{\text{yaw}} \end{bmatrix} = \mathbf{C}(\psi) \begin{bmatrix} M_{1-3}^{\text{out}} \\ M_{2-3}^{\text{out}} \\ M_{3-3}^{\text{out}} \end{bmatrix}, \quad (3)$$

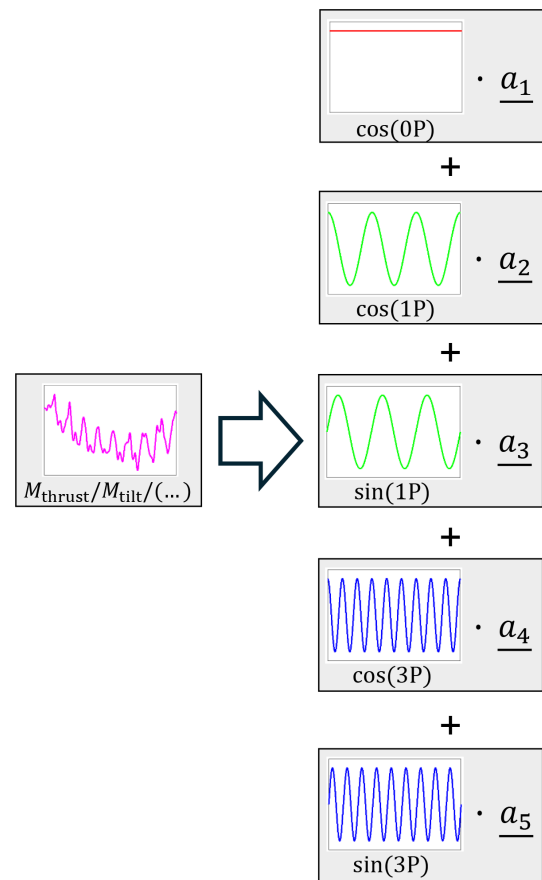
$$\begin{bmatrix} M_{\text{torque}} \\ M_{\text{lateral}} \\ M_{\text{vertical}} \end{bmatrix} = \mathbf{C}(\psi) \begin{bmatrix} M_{1-3}^{\text{in}} \\ M_{2-3}^{\text{in}} \\ M_{3-3}^{\text{in}} \end{bmatrix}, \quad (4)$$

where  $M_{1-3}^{\text{out}}$  is the out-of-plane bending moment, and  $M_{1-3}^{\text{in}}$  is the in-plane bending moment for blade indices 1–3.

The rotor load time series are decomposed into their key frequency components. This process is visualised in Fig. 6. The five frequency components selected via a conditional dependence study (Azadkia and Chatterjee, 2021) are  $\cos(0P)$ , used to describe the mean signal value;  $\cos(1P)$  and  $\sin(1P)$ , used to describe the once-per-revolution variation; and, finally,  $\cos(3P)$  and  $\sin(3P)$ , used to describe the thrice-per-revolution variation. A sliding window function with a width of three revolutions is applied, projecting the original load time series onto each of the frequency components. The Fourier coefficient vectors ( $\mathbf{a}_1$ – $\mathbf{a}_5$ ) obtained from these projections are stored, producing a coefficient time series for each rotor load.

In order to capture the short-term temporal dependencies and patterns in the time series of all wind sensing inputs, the features are embedded with their lagged values. For each sample in a time series, two additional features expressing the past value of the curve are added. These lagged features are obtained by shifting the time series 4 and 8 s from the current time stamp. This effectively makes the estimation more stable and noise resistant, as the wind slice is reconstructed with turbine response across several seconds. Specific lag values used are determined by testing the framework's performance with a few different configurations and choosing the one that gives the best overall results.

Due to implementing the transformations discussed in this section, using the turbine response data as wind sensing input effectively stops being a time series modelling task; instead, it becomes a time-independent regression task. Considering the entire feature extraction process, at each point in time, a 96-dimensional input vector captures and encodes the tur-



**Figure 6.** An example of rotor load time series decomposition into frequency component coefficient vectors ( $\mathbf{a}_1$ – $\mathbf{a}_5$ ), capturing the contributions of each Fourier mode at each point in time.

bine's current operating state. Table 2 serves as a summary of these features.

### 2.3.3 Wind sensing output processing

The three-dimensional wind field representations are processed for data compression and efficient extraction of their key features for the best application in wind sensing. The wind field is represented with a sequence of  $YZ$  slices showing the  $U$  distribution at each point along the  $X$  axis. The spatial features of each slice are extracted, capturing the key characteristics of the wind field in a more compact form. The feature extraction is performed using the two-dimensional discrete cosine transform (DCT) (Ahmed et al., 1974). The result is similar to applying the principal component analysis; however, as opposed to this far more complex method, the DCT is a fixed transform and does not require any training. The DCT effectively converts the wind slice into a collection of cosine functions oscillating at different frequencies in both the vertical and the horizontal directions. Each DCT coefficient corresponds to a specific frequency in the transformed domain and describes the contribution of each fre-

**Table 2.** A summary of features extracted from turbine response.

Original turbine response signal	Preprocessing methods	Resultant no. of features
$M_{1-3}^{\text{out}}, M_{1-3}^{\text{in}}$	Rotation matrix, Coleman transform, frequency decomposition, lag embedding	90
$\bar{\beta}$	Lag embedding	3
$\omega$	Lag embedding	3
		Total: 96

quency in the distribution of  $U$  on the given  $YZ$  slice. This concept, applied to an example wind field  $YZ$  slice, is presented in Fig. 7.

The lower-order frequencies primarily capture the large-scale structures of the field, while the higher-order frequencies are more associated with encoding smaller-scale, random fluctuations. It can be seen that the most significant values can be found in the upper-left corner of the diagram, meaning the majority of flow distribution is being kept by the lower orders of DCT coefficients. For this reason, the key information on the flow can be easily extracted by storing only the lower-order coefficients and disregarding the higher-order coefficients. Truncating the collection of DCT-order coefficients essentially acts as a spatial low-pass filter. In this work, retention of eight DCT coefficients is found to enhance computational efficiencies while allowing for good wind sensing accuracy. The right plot in Fig. 7 shows the effect of truncating the higher frequencies; it can be noticed how the wind slice is significantly “smoothed”, leaving only the major  $U$  fluctuations.

### 2.3.4 Training the wind sensing estimator

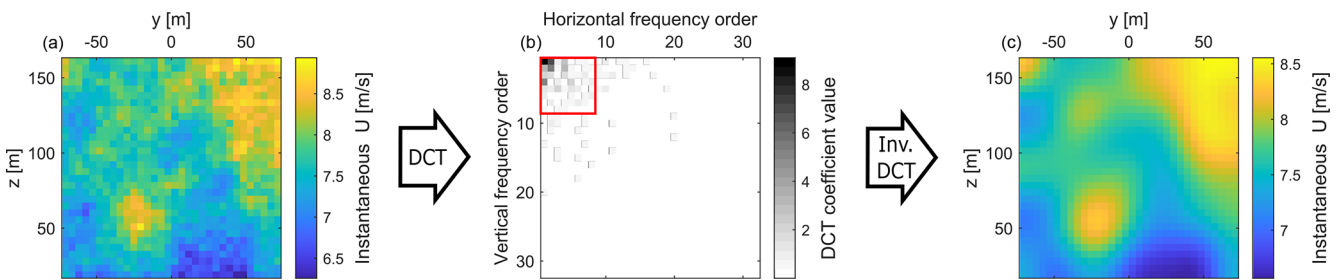
Having extracted the features from both the inputs and the outputs, two-thirds of data are used for the wind field estimator training, while the remaining one-third is put aside for generating wind field estimations for wake detector testing. The current implementation utilises a localised linear regression approach (Cleveland et al., 1988), where the estimator training is performed for multiple localised models. The pre-processed turbine response data are a time series of multi-dimensional input points, with each dimension representing a different feature (see Table 2). Two of these dimensions, namely unlagged  $\bar{\beta}$  and unlagged  $\omega$  (these being the key variables indicating the turbine’s operating point), are used to define a projection plane, allowing input points to be mapped and binned by their values. A set of evenly spaced points on this plane is selected, and a local linear regression model is trained at each. Training data are assigned to models based on proximity within a defined radius, with overlapping points contributing to multiple estimators. Predictions for new inputs are made using the model nearest to their location on the projection plane. As a result, the training process builds

functions that correlate the inputs to outputs only at a narrow section of data distribution. This eliminates the need to fit a global non-linear function that would singlehandedly need to describe the complex relationship between wind and turbine signals.

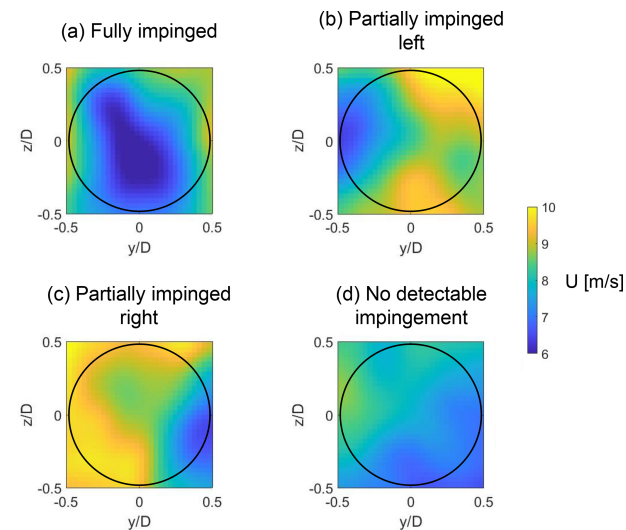
### 2.4 Framework implementation: wake detection

Considering that wake detection can essentially be reduced to a classification task, we propose dealing with it by implementing a convolutional neural network. The overview of this deep learning technique can be found in Appendix A. The approach is to operate on the  $YZ$  snapshot samples (so-called “front view”), which show the estimated instantaneous wind component  $U$  across the whole rotor area. This allows for clear differentiation between impinged and non-impinged conditions. To accurately capture the different forms of wake impingement, four distinct classes are defined. These are as follows: (a) fully impinged, with the turbine directly downstream of the closest neighbour; (b) partially impinged left, with the wake impacting the left side of the rotor; (c) partially impinged right, with the wake impacting the right side of the rotor; and (d) no detectable impingement.

The turbine response time series from the remaining 400 simulations, set aside during the wind sensing estimator training, are now used for generating wind field estimations. These estimations form the basis of a new dataset for CNN training. The training samples are labelled according to the wind direction specified in their respective simulation setup. In turn, the four classes directly relate to the four wind directions used during the training data acquisition (see Sect. 2.2). An extensive manual review of the produced samples allows for the following conclusion: in the available dataset, a clear, non-dissipated wake is only visible for lower values of the selected ambient wind speed range – typically up to  $14\text{--}16\text{ m s}^{-1}$ , depending on the ambient turbulence intensity. Simulations with higher wind speeds exhibit stronger wake meandering, which results in the wake deficit often being absent from the  $YZ$  snapshot samples. The wake deficit depth is also weaker due to the smaller thrust force in the above-rated operation. Consequently, many samples labelled “fully” or “partially impinged” do not, in fact, exhibit a wake. The aim of the detector is to reliably recognise



**Figure 7.** An example showing how the YZ slice of the wind field (a) is transformed into a matrix of DCT components (b) and then back into a truncated slice of wind through an inverse DCT (c). The red square indicates which DCT coefficients are stored for the implemented wind field representation.



**Figure 8.** Example front-view YZ snapshot samples representing four classes defined for the neural network training. The rotor area is indicated with the black circle.

whether the turbine is likely experiencing a wake impingement from a nearby machine. To ensure this robustness, the training data must exclude the aforementioned mislabelled samples. With that in mind, the CNN is trained with data from simulations where the mean ambient wind speed  $U_{\text{amb}}$  is between 5 and 15 m s<sup>−1</sup> while maintaining the full range of turbulence intensity values (3 %, 5 % and 7 %). The YZ snapshot samples are taken every 10 s from the available training data, resulting in 11 200 labelled samples. Figure 8 shows selected YZ samples being appropriate examples of each one of the four classes.

The implemented CNN architecture is presented in Table 3. The input layer has the same resolution as the YZ slice of estimated wind. The three convolution layers have an increasing number of kernels, allowing the network to progressively learn more complex features at deeper levels (Goodfellow et al., 2016). A batch normalisation layer is added after each of the convolution layers for the sake of normalisation of weight gradients and neuron activations, thus speeding up

**Table 3.** Implemented CNN architecture.

Layer no.	Name	Details
1	Image input	Layer size = 32 × 32
2	Convolution	8 kernels with size = 3
3	Batch normalisation	
4	ReLU	
5	Max pooling	Pool size = 2, stride = 2
6	Convolution	16 kernels with size = 3
7	Batch normalisation	
8	ReLU	
9	Max pooling	Pool size = 2, stride = 2
10	Convolution	32 kernels with size = 3
11	Batch normalisation	
12	Fully connected	
13	Output normalisation	Type: softmax
14	Classification	Layer size = 4

the training process. Two max pooling layers reduce the spatial size of the feature maps. The activation function is selected as the rectified linear unit (ReLU). The final three layers transform the learned features into predictions, connecting all of the neurons, normalising outputs into probabilities and performing final classification. An extended description of the layers used can be found in Appendix A. The optimal hyperparameter values are chosen based on trial and error, with the CNN accuracy monitored through integrated testing.

The training is conducted using MATLAB’s Deep Learning Toolbox. A total of 90 % of the available labelled data (10 080 samples) are used for training. The samples are shuffled before the training process and after each one of the four epochs. The training is repeated several times to make sure there are no deviations. The remaining 10 % of samples are used for an integrated testing cycle, which automatically occurs after the training is complete. Figure 9 shows its results in the form of a confusion matrix. As can be observed, the main source of error is the misclassifications between the “fully impinged” and “no detectable impingement” classes. This phenomenon is further investigated in Sects. 3 and 4,



where the performance of each of the models is analysed in more detail under a variety of conditions.

## 2.5 Framework implementation: wake characterisation

The final part of the developed framework is the identification of the key wake parameters in cases where a clear wake impingement is detected. This is undertaken by scanning for the characteristic velocity deficit shape in the appropriate wind field representations.

### 2.5.1 Two-dimensional Gaussian fitting

The “tracking” of wake position and width is achieved by implementing a two-dimensional Gaussian fitting scheme based on a least squares algorithm. This method is a commonly used solution in the literature for dynamic wake property analysis (Abkar and Porté-Agel, 2015; Trujillo et al., 2011; Conti et al., 2020). First, the wake deficit is defined for every sample as follows:

$$U_{\text{def}}(y, z) = U(y, z) - U_{\text{amb}}, \quad (5)$$

where  $U(y, z)$  is the instantaneous wind speed distribution across the  $YZ$  plane, and  $U_{\text{amb}}$  is the mean ambient wind speed value.

A general bivariate two-dimensional Gaussian function for spatial variables of  $y$  and  $z$  is expressed as

$$f_{\text{bivariate}}(y, z) = \frac{A}{2\pi\sigma_y\sigma_z\sqrt{1-\rho^2}} \exp\left[-\frac{1}{2(1-\rho^2)}\left(\frac{(y-y_c)^2}{\sigma_y^2} - \frac{2\rho(y-y_c)(z-z_c)}{\sigma_y\sigma_z} + \frac{(z-z_c)^2}{\sigma_z^2}\right)\right], \quad (6)$$

where  $A$  is the amplitude equal to the height of the peak;  $\sigma_y$  and  $\sigma_z$  are standard variations along the  $Y$  and  $Z$  axes, respectively;  $\rho$  is the correlation coefficient between the function's spread in  $Y$  and  $Z$ ; and  $y_c$  and  $z_c$  are the means along  $Y$  and  $Z$ , respectively.

The time series of wake deficits from Eq. (5) is fitted to Eq. (6) through a Levenberg–Marquardt non-linear least squares algorithm implemented with MATLAB's Optimization Toolbox. Fitted Gaussian parameters are then used to identify wake properties: the function peak at  $(y_c, z_c)$  can be interpreted as the position of the wake centre;  $\sigma_y$  and  $\sigma_z$  describe the spread of the wake along the  $Y$  and  $Z$  axes, respectively; and  $\rho$  can be used to calculate the lengths of the semi-minor and semi-major axes of the wake ellipse and their orientation with respect to the  $YZ$  axes. The initial guess for  $\sigma_y$  and  $\sigma_z$  is defined as half of the rotor diameter  $D$  in order to best reflect the typical width expected of a wake.

As explained earlier, the framework outputs refer solely to the lateral wake characteristics, which are expressed by the

variation on the  $Y$  axis. Out of all fitted parameters, only two are required to describe the lateral wake variation:  $y_c$  and  $\sigma_y$ . To parametrically represent the size of the wake deficit, it is assumed that the lateral half width corresponds to the standard deviation in the  $Y$  direction. With that in mind, the wake extends laterally for  $2\sigma_y$ , with the fitted centre  $y_c$  in the middle. Figure 10 shows an example of the bivariate two-dimensional Gaussian fit applied on a  $YZ$  wind field sample, with the key characterised properties indicated.

### 2.5.2 Filtering with moving average window

The lateral wake half-width and centre position time series are smoothed using a moving average filter to reduce high-frequency noise from sample-by-sample variability while preserving slower variations associated with wake meandering. This way, the ultimate characterised wake property time series  $y_c^{\text{filt}}$  and  $\sigma_y^{\text{filt}}$  are defined as follows:

$$\begin{bmatrix} y_c^{\text{filt}}(t) \\ \sigma_y^{\text{filt}}(t) \end{bmatrix} = \frac{1}{\tau} \int_{t-\tau}^t \begin{bmatrix} y_c(\theta) \\ \sigma_y(\theta) \end{bmatrix} d\theta. \quad (7)$$

The variable  $\tau$  stands for window size in seconds, which is calculated separately for every wind field in the following manner:

$$\tau = \frac{L_{\text{meand}}}{U_{\text{hh}}}, \quad (8)$$

where  $L_{\text{meand}}$  is the characteristic meandering length equal to two rotor diameters (with the value corresponding to findings from the literature, used in several wake models like the DWM itself, Larsen et al., 2007), and  $U_{\text{hh}}$  is the mean  $U$  value at hub height.

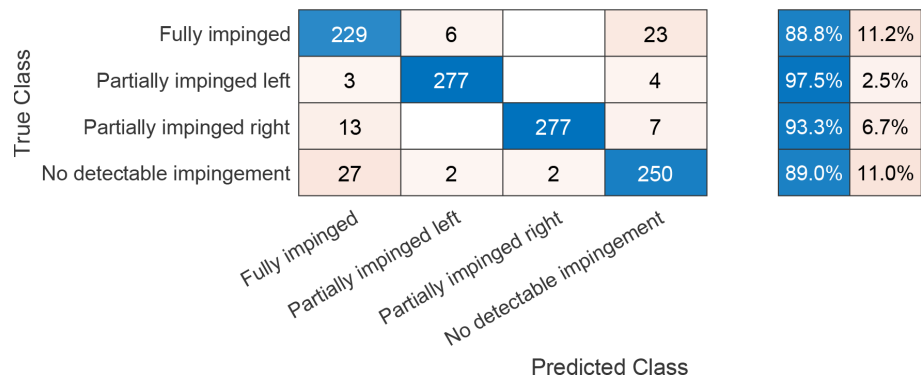
It should be noted that in this method of moving average filtering, the averaging window is not centred at the current sample; instead, the window extends from the past sample occurring  $\tau$  seconds ago to the current sample. As a result, the filtered wake centre and lateral half-width time series experience a small lag of  $\tau/2$ .

### 2.5.3 Treatment of non-impinged samples

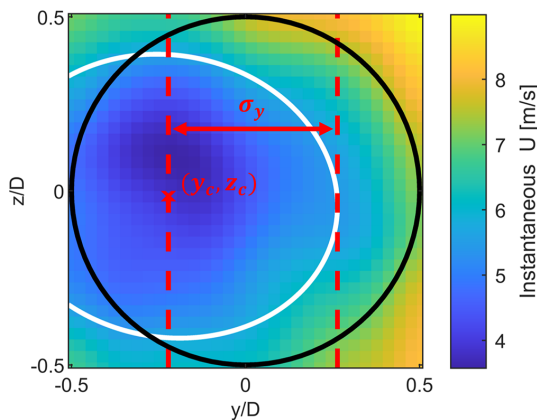
To satisfy the assumption that wake characterisation should not be performed for wind samples without clear impingement, the filtered wake properties are treated for a given time stamp  $i$  as follows:

$$y_c^{\text{filt}}(i), \sigma_y^{\text{filt}}(i) = \begin{cases} y_c^{\text{filt}}(i), \sigma_y^{\text{filt}}(i), & \text{if class } (i) = \text{fully/partially impinged} \\ \text{NaN}, & \text{if class } (i) = \text{no detectable impingement} \end{cases} \quad (9)$$

The assignment of NaN (not a number) values ensures that the samples identified as non-impinged are not being represented with a two-dimensional Gaussian wake in the framework output.



**Figure 9.** Confusion matrix for the integrated testing of the trained CNN. Each row represents the instances of a true class, whereas each column represents the instances of a predicted class. As a result, the values in the diagonal (blue) and off-diagonal (orange) cells correspond to correct and false classifications, respectively. Percentages of correct (blue) and false (orange) classifications within each true class are shown on the right.



**Figure 10.** An example of a bivariate two-dimensional Gaussian fit performance applied to a YZ wind field sample. The rotor outline is marked with the black line, the fitted wake ellipse is marked with the white line, the fitted wake centre at  $(y_c, z_c)$  is marked with the red x sign and the extent of the fitted lateral half width  $\sigma_y$  is marked with dashed lines.

### 3 Results

#### 3.1 Performance evaluation

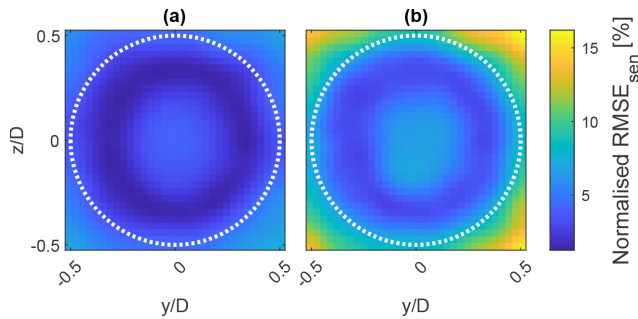
The framework’s performance is evaluated with 3600 new simulations divided into 10 subsets. The configurations with the number of runs are shown in Table 4. Each simulation is paired with the corresponding turbine response. All simulations are performed for one “receiver” turbine, which is more centrally located within the wind farm compared to the machine used for the training; by doing so, it experiences different types of impingement for different wind directions. Both qualitative and quantitative performance analysis is conducted for each of the three constituent models.

**Table 4.** Testing simulation subsets. For each  $U_{\text{amb}}$  and  $I_{\text{amb}}$  combination, 360 simulations capturing the full wind direction spectrum are performed.

$U_{\text{amb}}$ [m s <sup>-1</sup> ]	$I_{\text{amb}}$ [%]	Wind direction [°]	No. of simulations
5, 7, 9, 11, 13, 15	5	1–360 (step size 1)	2160
10	3, 5, 7, 9		1440
			Total: 3600

In this work, the reference for quantitative analysis of the models’ accuracy is the collection of the raw DWM-generated wind fields. Wind sensing accuracy analysis is a straightforward calculation of root mean square error (RMSE) between the  $U$  data points in simulated wind fields and the estimated equivalents. A reference for wake detection accuracy analysis is created via a new classifier trained analogically to the process described in Sect. 2.4, with the only difference being that the training dataset is derived from simulated wind fields, not the estimated ones. Without the bias from the wind field reconstruction, this classifier achieves approximately 99 % accuracy under integrated testing. As such, its classifications across all 3600 raw wind fields (with the wind sensing step not performed) are thus taken to be the “ground truth” reference. Finally, the reference for wake characterisation is obtained by processing a raw simulated wind field: a two-dimensional Gaussian is fitted, moving average filtering is applied and “non-impinged” wind slice samples are removed analogically to the process described in Sect. 2.5.

The wind sensing accuracy is first analysed by plotting the RMSE across the YZ plane for two example simulations from the testing dataset, presenting low and high ambient turbulence. This is shown in Sect. 3.2. The wind-direction-dependent wind sensing and wake detection performance is



**Figure 11.** Wind sensing RMSE plotted across YZ slice, normalised by the corresponding  $U_{\text{amb}}$  value of  $10 \text{ m s}^{-1}$ . (a)  $I_{\text{amb}} = 3 \%$ ; (b)  $I_{\text{amb}} = 9 \%$ . The rotor outline is marked with the dotted white line.

then analysed via a study of the sensitivity to  $U_{\text{amb}}$  and  $I_{\text{amb}}$ , comparing these models' mean accuracy across the entirety of the 10 subsets described above. This is discussed in Sect. 3.3. Section 3.4 presents detailed wake characterisation results. This is achieved by visualising the framework's performance across selected wind fields from the testing dataset, followed by computing the quantitative performance metrics of wake characterisation.

### 3.2 Wind sensing: YZ-wise accuracy

Figure 11 compares the YZ plane distribution of the wind sensing RMSE for low (a) and high (b) turbulence intensity. This distribution is obtained by calculating  $\text{RMSE}_{\text{sen}}$  at every YZ location over the full duration of the simulation  $n_t$ :

$$\text{RMSE}_{\text{sen}}(y, z) = \sqrt{\frac{1}{n_t} \sum_{i=0}^{n_t} (U_i^{\text{est}}(y, z) - U_i^{\text{DWM}}(y, z))^2}, \quad (10)$$

where  $U_i^{\text{est}}(y, z)$  and  $U_i^{\text{DWM}}(y, z)$  are the YZ-specific  $U$  at time  $i$  for the estimated and DWM wind field, respectively. Both plots show the error to be higher at the rotor centre and in the corners outside the rotor area. The high- $I_{\text{amb}}$  plot reports significantly higher RMSE, with the critical corner area reaching approximately twice the value of the low-turbulence case. The ring-like pattern of the area with the lowest error (dark blue in the plot) is repeated across all wind conditions, whose plots are not shown here for brevity.

### 3.3 Wind sensing and wake detection: ambient condition sensitivity study

For all 3600 simulations, the YZ distributions of wind sensing RMSE (such as the examples in Fig. 11) are averaged across the entire rotor plane to obtain a single simulation-specific mean value. Furthermore, to check how the RMSE changes with wind conditions, mean  $\overline{\text{RMSE}_{\text{sen}}}$  values averaging all

**Table 5.** A comparison of performance metrics in wind sensing and wake detection under varying  $U_{\text{amb}}$  and  $I_{\text{amb}}$ .  $\overline{\text{RMSE}_{\text{sen}}}$  is normalised by the corresponding  $U_{\text{amb}}$  value.

$U_{\text{amb}}$ [ $\text{m s}^{-1}$ ]	$I_{\text{amb}}$ [%]	Normalised $\overline{\text{RMSE}_{\text{sen}}}$ [%]	$\text{RMSE}_{\text{det}}$ [%]
5	5	6.7	67
7	5	3.9	18
9	5	3.3	14
11	5	3.1	11
13	5	2.7	17
15	5	2.3	23
10	3	2.2	7
10	5	3.1	20
10	7	4.3	19
10	9	5.6	30

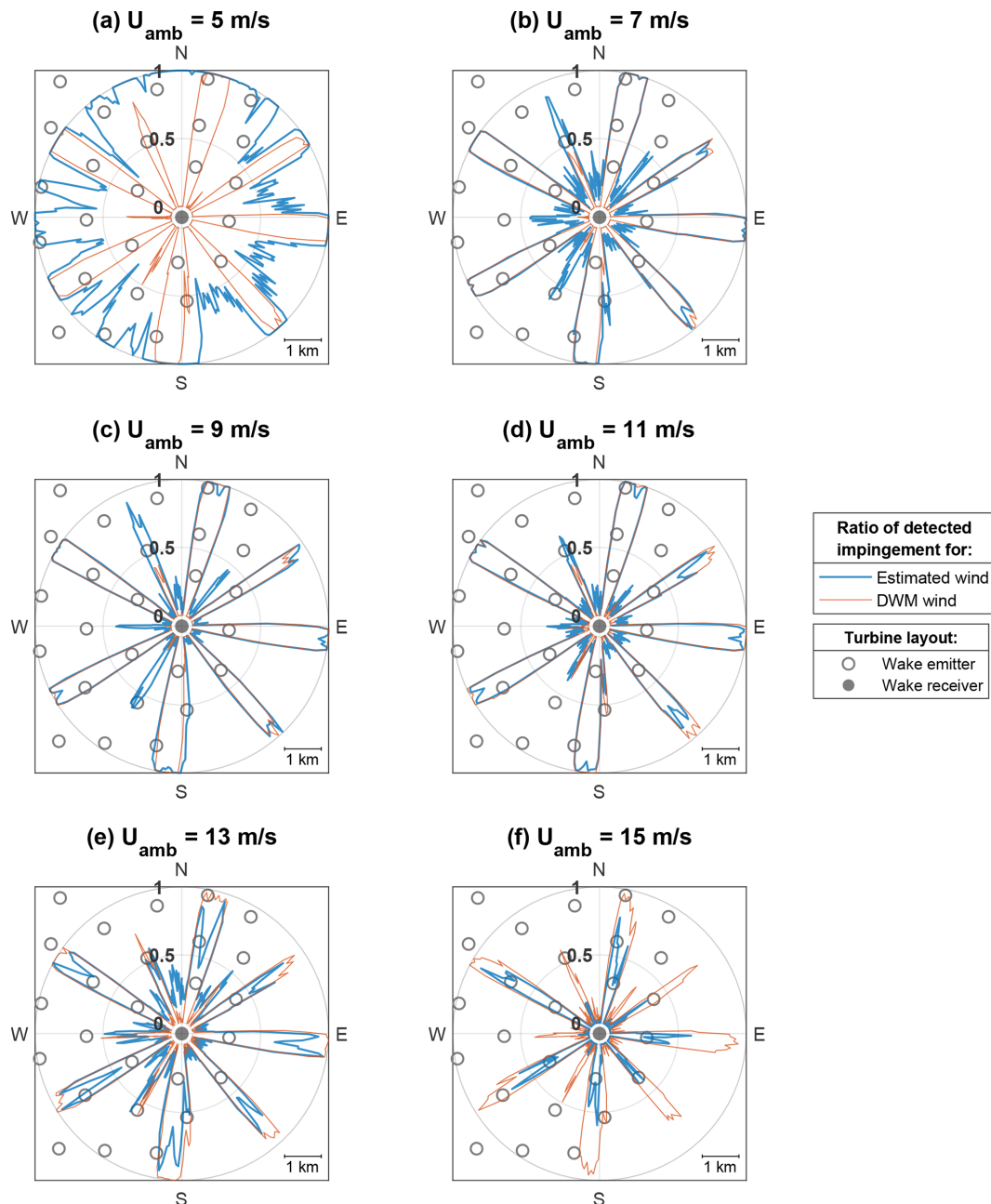
360 simulations with the same  $U_{\text{amb}}$  and  $I_{\text{amb}}$  are calculated. These values are presented in Table 5.

For each of the 3600 simulations, the proportion of samples (YZ wind slices at a point in time) classified as a clear wake impingement (“fully impinged”, “partially impinged left” or “partially impinged right”) is calculated. These simulation-specific impingement ratios are compared to their equivalents in the reference established by the “ground truth” classifier (see Sect. 3.1). This provides a quantitative measure of the wake detection error for a given wind direction. A single wake detection  $\text{RMSE}_{\text{det}}$  value is calculated across the entire wind direction range for each of the 10  $U_{\text{amb}}$  and  $I_{\text{amb}}$  combinations:

$$\text{RMSE}_{\text{det}} = \sqrt{\frac{1}{n_{\text{wd}}} \sum_{k=0}^{n_{\text{wd}}} (r_k^{\text{est}} - r_k^{\text{DWM}})^2}, \quad (11)$$

where  $n_{\text{wd}}$  is the number of wind directions (360 for the full wind direction range),  $r_k^{\text{est}}$  is the simulation-specific estimated impingement ratio and  $r_k^{\text{DWM}}$  is the simulation-specific impingement ratio from the raw DWM wind reference.

These values are presented in the rightmost column of Table 5. Figures 12 and 13 show the impingement ratios (both the DWM “ground truth” and the estimated values) across the full range of wind directions and under varying  $U_{\text{amb}}$  and  $I_{\text{amb}}$  conditions. For each  $U_{\text{amb}}$  and  $I_{\text{amb}}$  combination, the plot displays the wind farm layout with the wake “receiver” turbine in the centre. Polar values visualise the proportion of samples for a given wind direction classified as flow with a clear wake deficit. These proportions are shown as radial values between 0 (all simulation samples identified as “no detectable impingement”) and 1 (all simulation samples identified as “full impingement”, “partial impingement right” or “partial impingement left”).



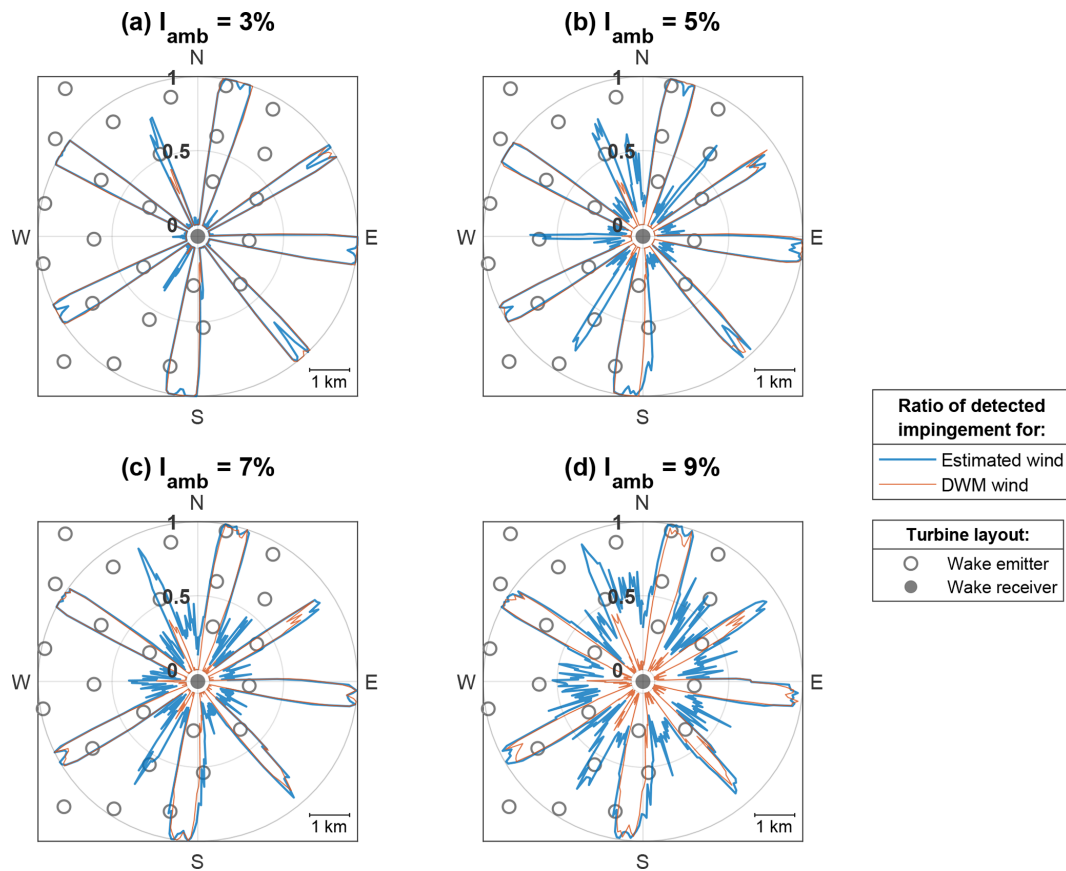
**Figure 12.** The wake detection model performance tested under a full wind direction range for six  $U_{\text{amb}}$  values between 5 and 15  $\text{m s}^{-1}$ .  $I_{\text{amb}} = 5\%$  for all cases. The radial values in blue and orange indicate the impingement ratio for a given wind direction. The wind farm layout is shown in the background.

### 3.4 Wake characterisation: time-wise accuracy

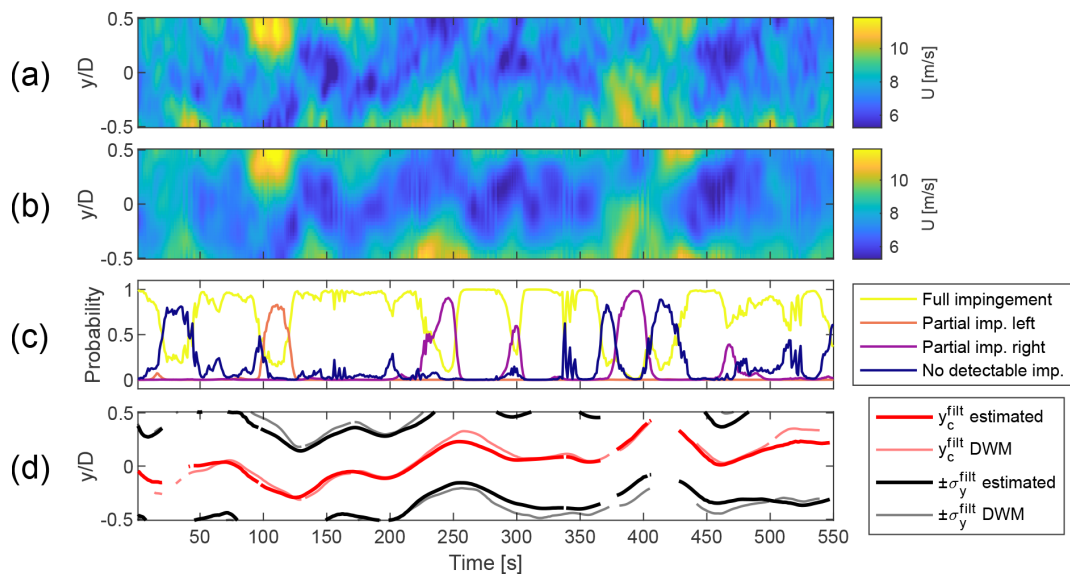
Figures 14 to 17 provide a detailed performance analysis of the framework across four example wind fields. In each figure, plots (a) and (b) show hub-height horizontal slices of the simulated and estimated  $U$  distributions, respectively. The (c) plots display direct wake detection outputs from the CNN's classification, assigning a value between 0 and 1 to each of the four classes, representing the probability of a

sample depicting a corresponding wake impingement case. These probabilities sum up to 1, so a near-perfect class representation has a confidence score near 1, with the other three values near 0. The (d) plots present the outputs of wake characterisation, showing the fitted and filtered lateral wake properties against the DWM reference (see Sect. 3.1).

Figures 14 and 15 compare the performance under high and low  $I_{\text{amb}}$ , respectively. They also exhibit different degrees of wake–rotor overlap, allowing us to investigate both

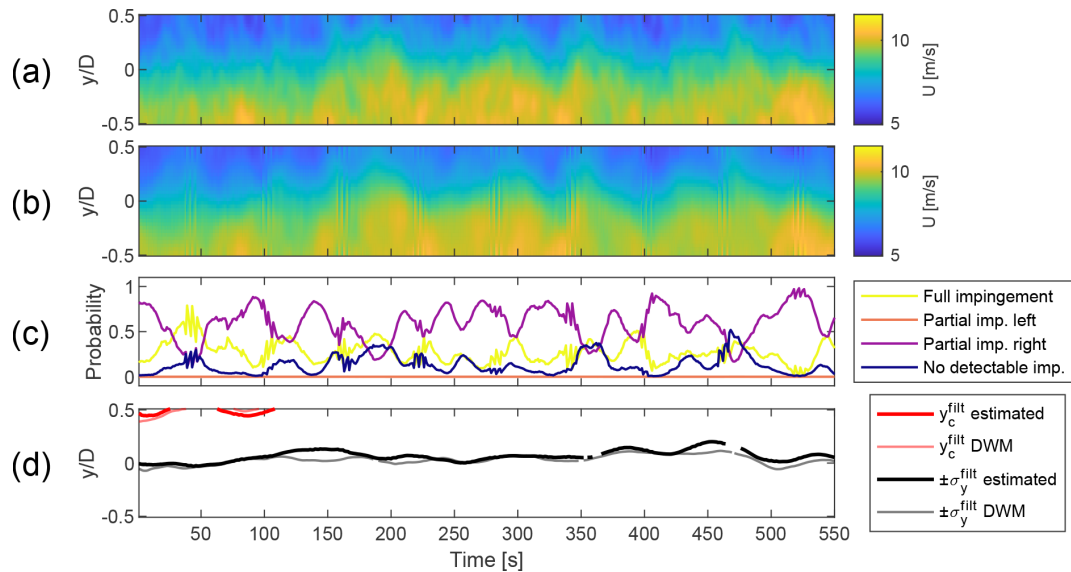


**Figure 13.** The wake detection model performance tested under a full wind direction range for four  $I_{\text{amb}}$  values between 3 % and 9 %.  $U_{\text{amb}} = 10 \text{ m s}^{-1}$  for all cases. The radial values in blue and orange indicate the impingement ratio for a given wind direction. The wind farm layout is shown in the background.

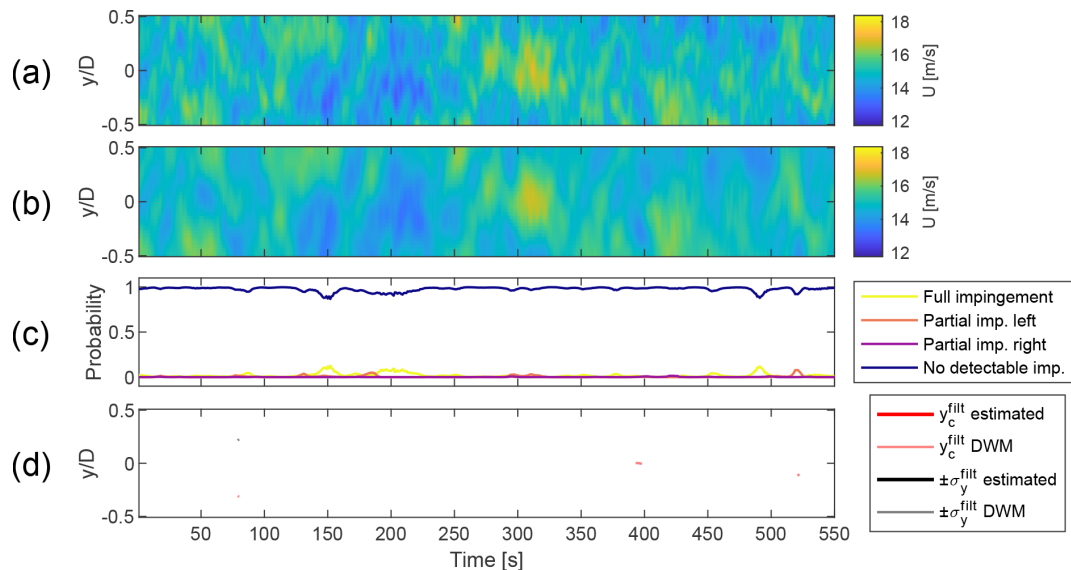


**Figure 14.**  $U_{\text{amb}} = 10 \text{ m s}^{-1}$ ,  $I_{\text{amb}} = 9 \%$ , fully impinged. (a) Horizontal slice of the simulated wind field at hub height. (b) Horizontal slice of the estimated wind field at hub height, obtained with the wind sensing model. (c) Classification scores obtained with the wake detection model. (d) Fitted wake properties obtained with the wake characterisation model next to the reference fitted on the raw DWM wind.





**Figure 15.**  $U_{\text{amb}} = 10 \text{ m s}^{-1}$ ,  $I_{\text{amb}} = 3 \%$ , partially impinged. (a) Horizontal slice of the simulated wind field at hub height. (b) Horizontal slice of the estimated wind field at hub height, obtained with the wind sensing model. (c) Classification scores obtained with the wake detection model. (d) Fitted wake properties obtained with the wake characterisation model next to the reference fitted on the raw DWM wind.

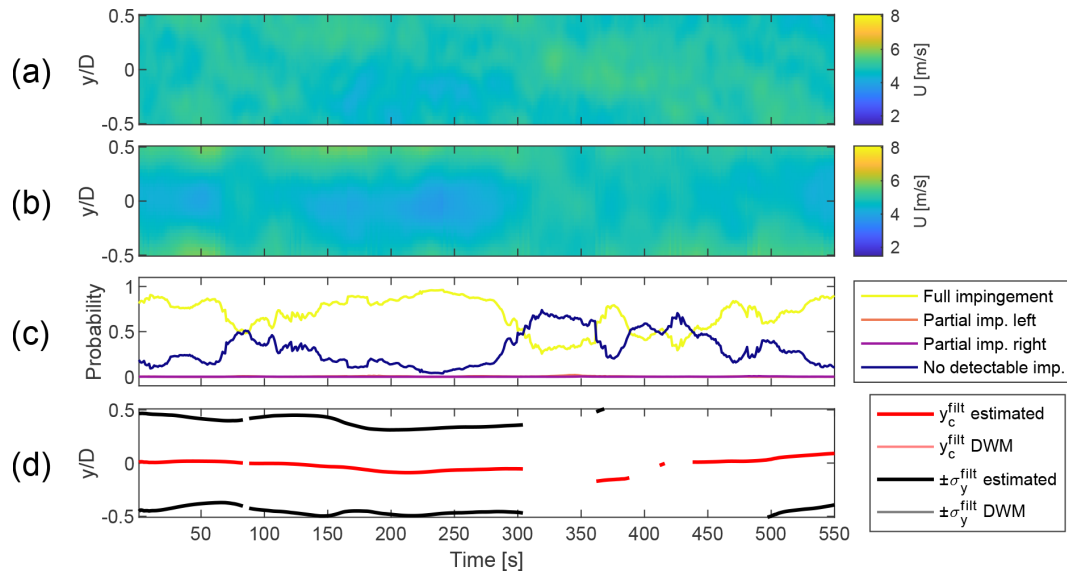


**Figure 16.**  $U_{\text{amb}} = 15 \text{ m s}^{-1}$ ,  $I_{\text{amb}} = 5 \%$ , no impingement. (a) Horizontal slice of the simulated wind field at hub height. (b) Horizontal slice of the estimated wind field at hub height, obtained with the wind sensing model. (c) Classification scores obtained with the wake detection model. (d) Fitted wake properties obtained with the wake characterisation model next to the reference fitted on the raw DWM wind.

full- and partial-impingement scenarios. Figures 16 and 17 show the performance in a scenario with no turbines upstream of the “receiver” device. They show the effects of operating under low- and high- $U_{\text{amb}}$  conditions, respectively.

Quantitative metrics for wake characterisation accuracy are computed as shown in Eqs. (12) and (13). Using the reference derived from raw DWM wind fields (see Sect. 3.1),

RMSE is calculated over the time series of both the lateral wake centre and the wake width. Samples classified as “no detectable impingement”, containing NaN values as discussed in Sect. 2.5, are excluded from the calculation.



**Figure 17.**  $U_{\text{amb}} = 5 \text{ m s}^{-1}$ ,  $I_{\text{amb}} = 5 \%$ , no impingement. (a) Horizontal slice of the simulated wind field at hub height. (b) Horizontal slice of the estimated wind field at hub height, obtained with the wind sensing model. (c) Classification scores obtained with the wake detection model. (d) Fitted wake properties obtained with the wake characterisation model next to the reference fitted on the raw DWM wind.

$$\text{RMSE}_{\text{char}}^{y_c} = \sqrt{\frac{1}{n_t} \sum_{i=0}^{n_t} \left( y_{c,i}^{\text{filt}}(\text{est}) - y_{c,i}^{\text{filt}}(\text{DWM}) \right)^2} \quad (12)$$

$$\text{RMSE}_{\text{char}}^{\sigma_y} = \sqrt{\frac{1}{n_t} \sum_{i=0}^{n_t} \left( \sigma_{y,i}^{\text{filt}}(\text{est}) - \sigma_{y,i}^{\text{filt}}(\text{DWM}) \right)^2} \quad (13)$$

Here  $n_t$  is the simulation duration;  $y_{c,i}^{\text{filt}}(\text{est})$  and  $\sigma_{y,i}^{\text{filt}}(\text{est})$  are the estimated  $y_c^{\text{filt}}$  and  $\sigma_y^{\text{filt}}$  at time  $i$ , respectively; and  $y_{c,i}^{\text{filt}}(\text{DWM})$  and  $\sigma_{y,i}^{\text{filt}}(\text{DWM})$  are  $y_c^{\text{filt}}$  and  $\sigma_y^{\text{filt}}$  at time  $i$  from the reference fitted on the raw DWM wind, respectively.

Table 6 shows the wake characterisation RMSE calculated for four example wind fields, comparing the model's accuracy for full and partial wake impingement under high and low  $I_{\text{amb}}$ . For brevity, only two of these wind fields are visualised in Figs. 14 and 15. Wind fields in which the DWM reference primarily shows “no detectable impingement” – such as those in Figs. 17 and 16 – are excluded from the RMSE analysis, as the reference wake characterisation mostly contains NaN values.

## 4 Discussion

### 4.1 Evaluation methods

The reference for wake detection is based on the classifications from a detector trained with simulated (rather than estimated) wind fields. This allows us to consider the effects of varied wake dispersion under different ambient conditions

**Table 6.** Performance metrics for wake characterisation.

Impingement case	$U_{\text{amb}}$ [m s <sup>-1</sup> ]	$I_{\text{amb}}$ [%]	$\text{RMSE}_{\text{char}}^{y_c}$ [D]	$\text{RMSE}_{\text{char}}^{\sigma_y}$ [D]
Full	10	9	0.05	0.09
Full	10	3	0.03	0.04
Partial	10	9	0.1	0.1
Partial	10	3	0.03	0.05

and arguably fits this analysis better than a general impingement definition based strictly on inflow angle. Indeed, to the best of the authors' knowledge, there is not a widely recognised definition of wake impingement (e.g. by means of reduced power output) that could be used here instead.

Wake characterisation accuracy is evaluated by processing the raw DWM wind with the methods from Sect. 2.5, thus extracting reference wake properties. This approach is preferred over taking the “ground truth” from the meandering wake centres applied internally in the DWM model, as turbulent fluctuations in the synthetic wind field – along with additional imposed turbulence – can cause the actual wake experienced by the turbine to deviate from the calculated position. Moreover, this method naturally accommodates the interaction of multiple, combined wakes. A reference derived by fitting a Gaussian profile on a simulated wind field has been used in other studies (Lejeune et al., 2022).

However, it should be noted that using the same methodology to both perform estimations and establish a reference, with the only major difference being the source of the wind field, carries a bias. This is especially relevant for calcu-

lated wake characterisation metrics; calculated RMSE values could be expected to increase with the use of a different reference approach. In future work, we aim to address this through validation with higher-fidelity simulations and wind field data.

## 4.2 Framework performance

### 4.2.1 Wind sensing

Although there are minor discrepancies for specific wind conditions, the developed wind sensing model presents generally good performance with low error between reference and estimated values. Figure 11 reveals that the wind field reconstruction quality is the highest at the radial distance of approximately half of the rotor radius. This is likely due to the fact that the flow impacting the mid-blade region has the largest influence on the root bending moments. The  $YZ$  locations outside of the rotor plane yield the highest RMSE due to blade sensors being unable to meaningfully react to the flow fluctuations from that region. Figures 14 and 15 show that major flow fluctuations are well captured in the estimated wind field representations. Some smoothing occurs for smaller-scale fluctuations, but this does not pose a major problem for wake detection. In fact, the attenuation of less significant turbulent eddies allows the wind sensing procedure to act as a spatial filter, focusing on larger flow structures like a wake deficit. The sensitivity study from Sect. 3.3 shows that the normalised  $\overline{\text{RMSE}}_{\text{sen}}$  rises with increasing  $I_{\text{amb}}$ , which can be attributed to the higher flow complexity in more turbulent wind. The cause of higher error in low- $U_{\text{amb}}$  cases is investigated in Fig. 17. The wind field reconstruction shows a minor  $U$  deficit that is not present in the simulated wind field, which is identified by the CNN as a wake impingement. This phenomenon occurs for the majority of wind directions at  $U_{\text{amb}} = 5 \text{ m s}^{-1}$  (see Fig. 12a), which in turn is the cause of high wake detection RMSE captured in Table 5. Looking at the high- $U_{\text{amb}}$  case in Fig. 16, it is apparent that a more heterogeneous wind field effectively disperses such anomalies. This emergence of “fake” wakes under low-wind-speed conditions is likely a result of the training dataset composition, 75 % of which shows wake impingement, leaving only 25 % for purely ambient turbulence cases. Using a single wind sensing estimator across these cases introduces a bias, as the model is predominantly trained on wake impingement conditions. The misclassification between “fully impinged” and “no detectable impingement” is quantified by the confusion matrix in Fig. 9.

### 4.2.2 Wake detection

Examining both Fig. 12 and Fig. 13, most plots show the wakes generated by upstream turbines to be correctly identified for the corresponding wind directions at the “receiver” turbine. There is a distinct difference in the proportion of samples identified as a wake depending on the distance

between the “emitter” and “receiver”. For turbines further away, the proportion of detected impingement is lower due to the wake losing its distinctive shape over a longer distance and interacting with the turbulent atmospheric boundary layer for a longer time. This relationship is evident in both the number of “impingement” samples per simulation (length of the blue/orange peak) and the number of wind directions for which the wake is detected (width of the blue/orange peak).

The superposition of wakes does not appear to have a significant effect on wake detection performance. Wakes from eastern turbines (a single machine upstream) do not yield substantially different wake detection results than wakes from deep inside the wind farm to the west. This is likely a consequence of the wake superposition approach used, where wind speed at each  $YZ$  grid point on the receiving turbine is derived based on the strongest influencing wake deficit. As a result, the primary aspect impacting the wake detection performance is the distance between the “receiver” and “emitter” machines. This is also due to limiting the training data to one pair of “emitter”–“receiver” turbines at a fixed distance.

As shown in Table 5, the wake detection RMSE largely follows the trends from the wind sensing RMSE. This is expected, as one model uses the outputs from the other. The highest error across all cases is reported for ambient wind speed of  $5 \text{ m s}^{-1}$ , which is due to the wind sensing anomaly described above. The wake detection is progressively more accurate with increasing  $U_{\text{amb}}$  (attributed to dispersion of the anomaly), until the RMSE begins to rise above  $11 \text{ m s}^{-1}$ . Visual inspection of Fig. 12e and f reveals that this is a consequence of increased estimation–reference mismatch at the wind directions with high impingement ratios. These wind speeds usually correspond to above-rated operation, where the wakes are less pronounced due to decreased thrust, likely making them more difficult to detect. The wake detection RMSE rises with increasing  $I_{\text{amb}}$  values; the plots in Fig. 13 show that the main source of the error is increasing “noise” of some wake impingement detected for all of the wind directions. This is expected, as more turbulent wind fields contain more large-scale ambient eddies that could be identified as a wake.

The sample-by-sample classification is presented in the (c) plots in Figs. 14 to 17. The CNN’s output aligns with the respective wind slice samples; for example, when wake meanders from the centre to the  $Y$ -positive part of the snapshot, the highest probability transitions from “full impingement” to “partial impingement left”. This indicates that the instantaneous wake position can be, at least to some extent, tracked even without a dedicated wake characterisation model.

### 4.2.3 Wake characterisation

Figures 14 and 15 show the sample-by-sample wake deficit characterisation for two example wind fields. Visual comparison between the simulated wind field in the (a) plots

and characterised wake in the (d) plots shows that the wake behaviour is appropriately captured in both cases. The estimated wake properties fit the reference values reasonably well. The time series of the fitted wake centre and lateral border are near continuous for lower  $I_{\text{amb}}$ , while higher  $I_{\text{amb}}$  yields a characterisation with a few gaps. This is due to the removal of samples classified as the “no detectable impingement” class, which happens more often when the deficit shape is distorted by increased ambient turbulence. Table 6 brings further insight, showing quantitative characterisation accuracy metrics for full and partial impingement under low and high  $I_{\text{amb}}$ . As expected, higher turbulence causes the estimated wake properties to diverge more from the reference, thus increasing the RMSE. For all wind fields, the error is higher for  $\sigma_y$  than for  $y_c$ , likely due to the wake centre being more easily identified by the least squares algorithm (it is simply the area with the lowest value). Error comparisons between full and partial impingement reveal mostly similar values, with the only outlier being  $\text{RMSE}_{\text{char}}^{y_c}$  at  $I_{\text{amb}} = 9\%$ , where the partial wake error is twice that of the full wake. This is a logical outcome, given the increased difficulty of characterisation when the wake meandering motion is strong and the centre is outside the  $YZ$  snapshot.

Figure 17d shows how the “fake” wake anomaly discussed above results in a characterisation performed on ambient turbulence. This effect is gone when  $U_{\text{amb}}$  is higher (see Fig. 16d), which is why the characterisation is missing for the entire wind field except a few isolated samples.

### 4.3 Applicability

Overall, the presented performance shows that the developed methodology is a promising solution to the challenging problem of generalised wake impingement estimation. The less accurate wake detection for  $U_{\text{amb}}$  values of 5 and  $15 \text{ m s}^{-1}$  is less critical after the consideration of specific application of this work. Firstly, at  $U_{\text{amb}} = 5 \text{ m s}^{-1}$  the turbine is usually close to its cut-in wind speed (Carrillo et al., 2013), meaning that for many cases, applying flow control would not be necessary. Secondly, the wind farm flow control brings the largest benefits for below-rated operation, where, due to higher energy extraction and lower wind, the wakes are more pronounced (Scott et al., 2024). These conditions align with the scenarios where the framework demonstrates optimal performance. Wind speeds above  $13 \text{ m s}^{-1}$  (where the framework performance deteriorates) are generally less frequent than lower-wind-speed conditions (Shu and Jesson, 2021), further mitigating the impact of these issues. A similar comment can be made with respect to wake detection RMSE increasing with  $I_{\text{amb}}$ . Wake steering control brings the largest benefits when the wind is less turbulent, which is where the framework shows the best performance.

The presented results also highlight that wake detection accuracy is highly dependent on the training data selection. This indicates that improved performance across a broader

range of conditions could be achieved by extension of the training dataset, e.g. by learning the wake deficit shape from various “receiver” turbines. One of the testing subsets has  $I_{\text{amb}}$  equal to 9 %, a value outside the training range of 3 %–7 %. While having one of the highest errors in the study, the wakes within this subset are nonetheless detected at correct wind directions. This highlights the framework’s ability to produce reasonable results for wind fields with ambient conditions “unknown” to the estimator. This is highly advantageous for the field application context, where the pre-deployment training procedure would only capture some of the wind effects experienced by the wind farm.

### 4.4 Current limitations

The present methodology treats each  $YZ$  wind snapshot entirely independently. It therefore seems clear that improved performance (and solutions for some of the problematic cases which are identified) could be readily achieved by extending the methodology to undertake a post-processing analysis which accounts for time variations in results. This could, for example, allow for the imputation of gaps that occur in characterised time series for more turbulent wakes (see Fig. 14); moreover, by considering classifications of time-adjacent snapshots, it could help in the correct detection of fully impinged cases which are misidentified as unimpinged. Kalman filtering appears to be a technique well suited to these possible methodological extensions.

It should also be highlighted that the application of a moving average filtering within the methodology results in predictions lagging behind the true wind field. This lag ranges from approximately 12 s for  $U_{\text{amb}} = 15 \text{ m s}^{-1}$  to approximately 58 s for  $U_{\text{amb}} = 5 \text{ m s}^{-1}$ . These values are a consequence of matching the filtering window width to the wind field’s typical meandering timescales (see Eqs. 7 and 8). In the context of wind farm flow control, these levels of lag are not necessarily problematic, and the lag may be removable by introducing methods such as Kalman filtering. Relevant literature (Luo et al., 2024; Zhou et al., 2023) shows that long short-term memory networks could potentially provide a short-term forecast of the wake dynamics, thus providing an alternative solution. Further research needs to be conducted to investigate these leads.

Note also that the range of ambient turbulence intensities used for framework testing falls between 3 % and 9 %, and the atmospheric conditions for both training and testing are considered neutral. Large offshore wind farms could experience higher levels of turbulence deep within the turbine grid due to multiple wake interactions (Shaw et al., 2022). These simplifications should be acknowledged, and other turbulence levels, along with incorporating both stable and unstable boundary layer conditions, should be analysed in future work. Similarly, turbine yaw offsets are also not considered in this work. The current analysis implemented a medium-fidelity wake model, and so further work should also extend



this framework to more realistic wake structures, for example, those obtained through large-eddy simulations.

## 5 Conclusions

The proposed method for generalised wake impingement detection and characterisation shows good performance for a wide range of tested wind conditions. The framework's effectiveness is sensitive to ambient wind speed and turbulence intensity levels, with the best performance observed for wind speeds between 7 and 13 m s<sup>-1</sup> and turbulence intensities ranging from 3 % to 7 %. Wind field reconstruction reports high accuracy with an all-around RMSE of 3.7 % when normalised by mean ambient wind speed. The wake detection model correctly responds to the wake presence; averaging the RMSE across all wind fields used in testing indicates that the correct wake impingement case is identified for approximately 77 % of the samples. Wake characterisation appropriately adapts to the meandering motion, achieving high accuracy; wake centre tracking reports an average RMSE of 0.08 and 0.03  $D$  for  $I_{\text{amb}}$  of 9 % and 3 %, respectively; and wake lateral border tracking reports an average RMSE of 0.1 and 0.05  $D$  for  $I_{\text{amb}}$  of 9 % and 3 %, respectively. This work provides a baseline concept for generalised wake impingement estimation, which, with further improvements, could greatly contribute to better-informed wind farm flow control. Current limitations are identified, with the most important including the shortcomings of the evaluation methods and anomalies in wind sensing. Next steps for the further development of this framework include validation of the current setup in progressively more realistic environments (including yaw misalignment, higher-fidelity wind fields and wakes, and utilisation of field data), an extension of the training dataset (towards accounting for more flow phenomena from more turbines at different distances), and methodological extension to capture time variance between individual two-dimensional snapshots and opportunities for imputation and forecasting of wake properties. Kalman filtering is identified as a likely route to extending the framework in this manner.

## Appendix A: Convolutional neural network

CNNs are a specialised type of deep learning method, optimised for working with grid-like data (most commonly two-dimensional images). Their design is perfectly suited for learning spatial features from input data, making them especially effective for image recognition and classification tasks. Due to several key achievements in recent years, they have revolutionised areas such as facial recognition, handwriting analysis and autonomous vehicles (Li et al., 2022). In the context of wind energy, some examples of CNN implementation include wind power prediction (Zhu et al., 2017; Harbola and Coors, 2019) and early fault detection and classification

(Rahimilarki et al., 2022). The input layer of the CNN is the size of the grid-like data it processes – in a simple example of a grey-scale image, it would be its width multiplied by its height. If the task at hand is to classify the image, for example, determining which handwritten digit the image shows, then the output layer would consist of 10 nodes (one for each digit), each holding a confidence value between 0 and 1. The actual processing occurs in the hidden layers located between the input and output layers, and for most CNNs, these layers are a combination of the following:

- *Convolutional layer.* This layer applies learnable filters (also called kernels) that slide across the input data, only working on a small patch of the data at a time. A dot product of the kernel and data at each position creates a measure of how close a patch of data resembles a given spatial feature, such as an edge or arch. Doing this for the entire input results in a feature map. Performing this operation through multiple iterations with many kernels, progressively optimising their weights, allows us to combine small features into larger ones and is ultimately the key to detecting patterns.
- *Activation function.* Normally following the convolution layers, various activation functions like the rectified linear unit (ReLU) introduce non-linearity, enabling the network to learn complex patterns. It is achieved by restricting the connections between the neurons when the weighted sum is too low.
- *Pooling layers.* The pooling layers reduce the spatial size of feature maps, decreasing computational complexity and helping to prevent overfitting. Max pooling is the most commonly used method of doing so, selecting the maximum value from each patch of the feature map.
- *Fully connected layer.* After several cycles of convolution–activation–pooling, this layer connects every neuron in one layer to every neuron in the next, combining features learned by previous layers to produce final predictions.

While designing the neural network, one needs to decide on the number and type of hidden layers implemented in its architecture, as well as on the hyperparameters such as the number and size of learnable filters. Moreover, as with all deep learning techniques, the performance of the CNN is heavily dependent on the quality of data used for its training.

**Code and data availability.** The simulation tools, wind farm model and other data used in this research are property of Siemens Gamesa Renewable Energy.



**Author contributions.** Conceptualisation: PF, EdH, EmH. Writing (original draft): PF. Writing (review and editing): PF, EdH, EmH. Analysis: PF. Visualisation: PF. Supervision: EdH, EmH.

**Competing interests.** The contact author has declared that none of the authors has any competing interests.

**Disclaimer.** Publisher's note: Copernicus Publications remains neutral with regard to jurisdictional claims made in the text, published maps, institutional affiliations, or any other geographical representation in this paper. While Copernicus Publications makes every effort to include appropriate place names, the final responsibility lies with the authors.

**Financial support.** This research has been funded by Siemens Gamesa Renewable Energy and the EPSRC Industrial CASE program (grant no. EP/W522260/1, voucher no. 210039).

**Review statement.** This paper was edited by Cristina Archer and reviewed by two anonymous referees.

## References

- Abkar, M. and Porté-Agel, F.: Influence of atmospheric stability on wind-turbine wakes: A large-eddy simulation study, *Phys. Fluids*, 27, 035104, <https://doi.org/10.1063/1.4913695>, 2015.
- Adaramola, M. and Krogstad, P.-A.: Experimental investigation of wake effects on wind turbine performance, *Renew. Energy*, 36, 2078–2086, <https://doi.org/10.1016/j.renene.2011.01.024>, 2011.
- Ahmed, N., Natarajan, T., and Rao, K.: Discrete Cosine Transform, *IEEE Transact. Comput.*, C-23, 90–93, <https://doi.org/10.1109/T-C.1974.223784>, 1974.
- Azadkia, M. and Chatterjee, S.: A simple measure of conditional dependence, *Ann. Stat.*, 49, 3070–3102, <https://doi.org/10.1214/21-AOS2073>, 2021.
- Barthelmie, R. J., Pryor, S. C., Frandsen, S. T., Hansen, K. S., Schepers, J. G., Rados, K., Schlez, W., Neubert, A., Jensen, L. E., and Neckelmann, S.: Quantifying the Impact of Wind Turbine Wakes on Power Output at Off-shore Wind Farms, *J. Atmos. Ocean. Tech.*, 27, 1302–1317, <https://doi.org/10.1175/2010JTECHA1398.1>, 2010.
- Bingöl, F., Mann, J., and Larsen, G. I.: one-dimensional scanning, *Wind Energy*, 13, 51–61, <https://doi.org/10.1002/we.352>, 2010.
- Bottasso, C. and Schreiber, J.: Online model updating by a wake detector for wind farm control, in: *IEEE 2018 Annual American Control Conference (ACC)*, 27–29 June 2018, Milwaukee, WI, USA, 676–681, ISBN 978-1-5386-5428-6, <https://doi.org/10.23919/ACC.2018.8431626>, 2018.
- Carrillo, C., Montaña, A. O., Cidrás, J., and Díaz-Dorado, E.: Review of power curve modelling for wind turbines, *Renew. Sustain. Energ. Rev.*, 21, 572–581, <https://doi.org/10.1016/j.rser.2013.01.012>, 2013.
- Churchfield, M. J., Lee, S., Michalakes, J., and Moriarty, P. J.: A numerical study of the effects of atmospheric and wake turbulence on wind turbine dynamics, *J. Turbulence*, 13, N14, <https://doi.org/10.1080/14685248.2012.668191>, 2012.
- Cleveland, W. S., Devlin, S. J., and Grosse, E.: Regression by local fitting, *J. Econometr.*, 37, 87–114, [https://doi.org/10.1016/0304-4076\(88\)90077-2](https://doi.org/10.1016/0304-4076(88)90077-2), 1988.
- Coleman, R. P.: Theory of Self-Excited Mechanical Oscillations of Hinged Rotor Blades, Langley Memorial Aeronautical Laboratory, <https://ntrs.nasa.gov/citations/19930093628> (last access: 1 September 2025), 1943.
- Conti, D., Dimitrov, N., Peña, A., and Herges, T.: Wind turbine wake characterization using the SpinnerLidar measurements, *J. Phys.: Conf. Ser.*, 1618, 062040, <https://doi.org/10.1088/1742-6596/1618/6/062040>, 2020.
- Dimitrov, N., Natarajan, A., and Mann, J.: Effects of normal and extreme turbulence spectral parameters on wind turbine loads, *Renew. Energy*, 101, 1180–1193, <https://doi.org/10.1016/j.renene.2016.10.001>, 2017.
- Dong, L., Lio, W. H., and Meng, F.: Wake position tracking using dynamic wake meandering model and rotor loads, *J. Renew. Sustain. Energ.*, 13, 023301, <https://doi.org/10.1063/5.0032917>, 2021.
- Farrell, W., Herges, T., Maniaci, D., and Brown, K.: Wake state estimation of downwind turbines using recurrent neural networks for inverse dynamics modelling, *J. Phys.: Conf. Ser.*, 2265, 032094, <https://doi.org/10.1088/1742-6596/2265/3/032094>, 2022.
- Frederik, J. A., Doekemeijer, B. M., Mulders, S. P., and van Wingerden, J.: The helix approach: Using dynamic individual pitch control to enhance wake mixing in wind farms, *Wind Energy*, 23, 1739–1751, <https://doi.org/10.1002/we.2513>, 2020.
- Goodfellow, I., Bengio, Y., and Courville, A.: *Deep Learning*, MIT Press, ISBN 9780262035613, 2016.
- Harbola, S. and Coors, V.: One dimensional convolutional neural network architectures for wind prediction, *Energ. Convers. Manage.*, 195, 70–75, <https://doi.org/10.1016/j.enconman.2019.05.007>, 2019.
- Howland, M. F., Ghate, A. S., Lele, S. K., and Dabiri, J. O.: Optimal closed-loop wake steering – Part 1: Conventionally neutral atmospheric boundary layer conditions, *Wind Energy Sci.*, 5, 1315–1338, <https://doi.org/10.5194/wes-5-1315-2020>, 2020.
- Hutchinson, M. and Zhao, F.: *Global Wind Report 2023*, Global Wind Energy Council, <https://www.apren.pt/contents/publicationsothers/gwec-global-offshore-wind-report-2023.pdf> (last access: 1 September 2025), 2023.
- IEC: International Standard IEC 61400-1: Wind energy generation systems – Part 1: Design requirements, 2019.
- Kim, K.-H., Bertelè, M., and Bottasso, C. L.: Wind inflow observation from load harmonics via neural networks: A simulation and field study, *Renew. Energy*, 204, 300–312, <https://doi.org/10.1016/j.renene.2022.12.051>, 2023.
- Larsen, G. C., Madsen, H. A., Bingöl, F., and Mann, J.: Dynamic wake meandering modelling, Risø National Laboratory, <https://orbit.dtu.dk/en/publications/dynamic-wake-meandering-modeling> (last access: 2 September 2025), ISBN 978-87-550-3602-4, 2007.
- Larsen, T. J., Madsen, H. A., Larsen, G. C., and Hansen, K. S.: Validation of the dynamic wake meander model for loads and power production in the Egmond aan Zee wind farm, *Wind Energy*, 16, 605–624, <https://doi.org/10.1002/we.1563>, 2013.

- Lejeune, M., Moens, M., and Chatelain, P.: A Meandering-Capturing Wake Model Coupled to Rotor-Based Flow-Sensing for Operational Wind Farm Flow Prediction, *Front. Energ. Res.*, 10, 884068, <https://doi.org/10.3389/fenrg.2022.884068>, 2022.
- Li, Z., Liu, F., Yang, W., Peng, S., and Zhou, J.: A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects, *IEEE T. Neural Netw. Learn. Syst.*, 33, 6999–7019, <https://doi.org/10.1109/TNNLS.2021.3084827>, 2022.
- Lio, W. H., Larsen, G. C., and Poulsen, N. K.: Dynamic wake tracking and characteristics estimation using a cost-effective LiDAR, *J. Phys.: Conf. Ser.*, 1618, 032036, <https://doi.org/10.1088/1742-6596/1618/3/032036>, 2020.
- Liu, Y., Pamososuryo, A. K., Ferrari, R., Hovgaard, T. G., and van Wingerden, J.-W.: Blade Effective Wind Speed Estimation: A Subspace Predictive Repetitive Estimator Approach, in: *IEEE 2021 European Control Conference (ECC)*, 29 June–2 July 2021, Delft, the Netherlands, 1205–1210, ISBN 978-9-4638-4236-5, <https://doi.org/10.23919/ECC54610.2021.9654981>, 2021.
- Luo, Z., Wang, L., Fu, Y., Xu, J., Yuan, J., and Tan, A. C.: Wind turbine dynamic wake flow estimation (DWFE) from sparse data via reduced-order modeling-based machine learning approach, *Renew. Energy*, 237, 121552, <https://doi.org/10.1016/j.renene.2024.121552>, 2024.
- Madsen, H. A., Larsen, G. C., Larsen, T. J., Trolborg, N., and Mikkelsen, R.: Calibration and Validation of the Dynamic Wake Meandering Model for Implementation in an Aeroelastic Code, *J. Sol. Energ. Eng.*, 132, 041014, <https://doi.org/10.1115/1.4002555>, 2010.
- Mann, J.: The Spatial Structure of Neutral Atmospheric Surface-Layer Turbulence, *J. Fluid Mech.*, 273, 141–168, 1994.
- Munters, W. and Meyers, J.: Towards practical dynamic induction control of wind farms: analysis of optimally controlled wind-farm boundary layers and sinusoidal induction control of first-row turbines, *Wind Energ. Sci.*, 3, 409–425, <https://doi.org/10.5194/wes-3-409-2018>, 2018.
- Onnen, D., Larsen, G. C., Lio, W. H., Liew, J. Y., Kühn, M., and Petrović, V.: Dynamic wake tracking based on wind turbine rotor loads and Kalman filtering, *J. Phys.: Conf. Ser.*, 2265, 022024, <https://doi.org/10.1088/1742-6596/2265/2/022024>, 2022.
- Onnen, D., Petrović, V., Neuhaus, L., Langidis, A., and Kühn, M.: Wind tunnel testing of wake tracking methods using a model turbine and tailored inflow patterns resembling a meandering wake, in: *IEEE 2023 American Control Conference (ACC)*, 31 May–2 June 2023, San Diego, CA, USA, 837–842, ISBN 979-8-3503-2806-6, <https://doi.org/10.23919/ACC55779.2023.10155916>, 2023.
- Onnen, D., Larsen, G. C., Lio, A. W. H., Hulsman, P., Kühn, M., and Petrović, V.: Field comparison of load-based wind turbine wake tracking with a scanning lidar reference, *Wind Energ. Sci. Discuss. [preprint]*, <https://doi.org/10.5194/wes-2024-188>, in review, 2025.
- Raach, S., Schlipf, D., and Cheng, P. W.: Lidar-based wake tracking for closed-loop wind farm control, *J. Phys.: Conf. Ser.*, 753, 052009, <https://doi.org/10.1088/1742-6596/753/5/052009>, 2016.
- Rahimilarki, R., Gao, Z., Jin, N., and Zhang, A.: Convolutional neural network fault classification based on time-series analysis for benchmark wind turbine machine, *Renew. Energy*, 185, 916–931, <https://doi.org/10.1016/j.renene.2021.12.056>, 2022.
- Schreiber, J., Bottasso, C. L., and Bertelè, M.: Field testing of a local wind inflow estimator and wake detector, *Wind Energ. Sci.*, 5, 867–884, <https://doi.org/10.5194/wes-5-867-2020>, 2020.
- Scott, R., Hamilton, N., Cal, R. B., and Moriarty, P.: Wind plant wake losses: Disconnect between turbine actuation and control of plant wakes with engineering wake models, *J. Renew. Sustain. Energ.*, 16, 043303, <https://doi.org/10.1063/5.0207013>, 2024.
- SGRE: Siemens Gamesa now able to actively dictate wind flow at offshore wind locations, <https://www.siemensgamesa.com/en-int/newsroom/2019/11/191126-siemens-gamesa-wake-adapt-en> (last access: 24 May 2024), 2019.
- Shaw, W. J., Berg, L. K., Debnath, M., Deskos, G., Draxl, C., Ghate, V. P., Hasager, C. B., Kotamarthi, R., Mirocha, J. D., Muradyan, P., Pringle, W. J., Turner, D. D., and Wilczak, J. M.: Scientific challenges to characterizing the wind resource in the marine atmospheric boundary layer, *Wind Energ. Sci.*, 7, 2307–2334, <https://doi.org/10.5194/wes-7-2307-2022>, 2022.
- Shu, Z. R. and Jesson, M.: Estimation of Weibull parameters for wind energy analysis across the UK, *J. Renew. Sustain. Energ.*, 13, 023303, <https://doi.org/10.1063/5.0038001>, 2021.
- Simley, E. and Pao, L. Y.: Evaluation of a wind speed estimator for effective hub-height and shear components, *Wind Energy*, 19, 167–184, <https://doi.org/10.1002/we.1817>, 2016.
- Steinbuch, M., de Boer, W., Bosgra, O., Peters, S., and Ploeg, J.: Optimal control of wind power plants, *J. Wind Eng. Indust. Aerodynam.*, 27, 237–246, [https://doi.org/10.1016/0167-6105\(88\)90039-6](https://doi.org/10.1016/0167-6105(88)90039-6), 1988.
- Trujillo, J., Bingöl, F., Larsen, G. C., Mann, J., and Kühn, M.: Light detection and ranging measurements of wake dynamics. Part II: two-dimensional scanning, *Wind Energy*, 14, 61–75, <https://doi.org/10.1002/we.402>, 2011.
- Yang, X. and Sotiropoulos, F.: A Review on the Meandering of Wind Turbine Wakes, *Energies*, 12, 4725, <https://doi.org/10.3390/en12244725>, 2019.
- Zhou, L., Wen, J., Wang, Z., Deng, P., and Zhang, H.: High-fidelity wind turbine wake velocity prediction by surrogate model based on d-POD and LSTM, *Energy*, 275, 127525, <https://doi.org/10.1016/j.energy.2023.127525>, 2023.
- Zhu, A., Li, X., Mo, Z., and Wu, R.: Wind power prediction based on a convolutional neural network, in: *2017 International Conference on Circuits, Devices and Systems (IC-CDS)*, 5–8 September 2017, Chengdu, China, 131–135, <https://doi.org/10.1109/ICCD.2017.8120465>, 2017.