Applied Mathematical Modelling 132 (2024) 41–56

Contents lists available at ScienceDirect

Applied Mathematical Modelling

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

Experimental and numerical gust identification using deep learning models

Kayal Lakshmanan [a], Davide Balatti [a], Hamed Haddad Khodaparast [a,*], Michael I. Friswell [a], Andrea Castrichini [b]

[a] Swansea University, Faculty of Science and Engineering, Bay Campus, Swansea, SA1 8EN, United Kingdom
[b] Airbus Operations Ltd., Filton, BS99 7AR, United Kingdom

ARTICLE INFO

Keywords:
Gust identification
Inverse method
Aeroelasticity
Deep learning

ABSTRACT

Identifying gusts and turbulence events is of primary importance for designing future gust load alleviation systems, calculating airframe load, and analysing incidents. Due to the impossibility of their direct measurement, indirect methods are used and ad hoc experiments are necessary to validate the methodology. This paper employs Convolutional Neural Network and Long Short Term Memory (CNN-LSTM) as well as CNN models for in-flight gust identification. Two aeroelastic models, with different levels of fidelity, representative of a civil and commercial aircraft, are used to generate gust responses to train and test the Deep Learning (DL) models. The results highlight the capability of both LSTM-CNN and CNN models in reconstructing gusts across the entire flight envelope of a civil commercial aircraft. The CNN model demonstrated its ability to identify gusts and turbulence when they occur concurrently, similar to real-world scenarios, in a significantly shorter amount of time. Furthermore, its application to wind tunnel gust response measurements, where the inflow has previously been characterised, demonstrated the effectiveness of the proposed methodology for experimental measurements.

1. Introduction

The accurate and effective estimation of aircraft loads is a critical issue in the aerospace sector. These loads' extreme values are often the principal factors for determining the size of an aircraft's structure. Consequently, they significantly impact the aircraft's performance. Inaccuracies in load estimations can result in suboptimal design choices, typically manifesting as overly conservative structures that are heavier than necessary. This excess weight increases energy consumption, toxic emissions, and noise pollution. Therefore, accurate load estimations are directly linked to achieving the aviation industry's objective of net-zero carbon emissions by 2050 [1].

In-flight, the most significant loads an aircraft encounters are due to gusts and manoeuvres. Aircraft designers must, therefore, ensure that their designs comply with airworthiness standards. These include the capability to withstand vertical and lateral discrete gusts, as well as turbulence [2]. Turbulence refers to air movement that an aircraft navigates through, which changes the dynamic response of the aircraft. This change takes place as the effective incidence on each aerodynamic surface is influenced by any component of the air's velocity (termed gust velocity) that is perpendicular to the flight path [3]. In recent years, there has been an

* Corresponding author.
E-mail address: h.haddadkhodaparast@swansea.ac.uk (H. Haddad Khodaparast).

https://doi.org/10.1016/j.apm.2024.04.034
Received 12 October 2023; Received in revised form 9 April 2024; Accepted 15 April 2024

Available online 18 April 2024
0307-904X/© 2024 The Author(s). Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

increased effort to develop new active or passive Gust Load Alleviation (GLA) systems, as evidenced by several studies [4–9]. These studies show the importance of these loads. Directly measuring the time responses of gust velocities, which cause gust loads on aircraft structures, is not feasible. As a result, it becomes necessary to rely on identification strategies.

A better knowledge of the actual gusts and turbulence to which the aircraft is subjected could improve different aeronautical practices (e.g. design of GLA systems, analysis of accidents or estimation of the loads on the airframe). Regardless of the active or passive nature of the GLA systems, they are designed based on '1 - cos' gust and the von Karman wind turbulence model [10]. Indeed, in future, the design of GLA systems could be driven by an extensive database of actual gusts and turbulence events recorded by flying aircraft. Moreover, aircraft are equipped with digital flight data recorders, which enable the recording of specific aircraft parameters. In analysing flight incidents, gust and turbulence events are essential in estimating limit loads [11]. Furthermore, the advent of Industry 4.0 has brought complex industrial systems that are more interconnected, smart, and autonomous. A Digital Twin can be used to manage an aircraft throughout its service life and ensure its structural integrity [12]. To reliably calculate structural loads, the identification of gusts and turbulence events is necessary.

In the 40s and 50s, gust profiles were studied using the discrete-gust approach. This method used the peak vertical acceleration measured on an aircraft during a gust event to extract gust parameters such as the maximum gust velocities or the distance parallel to the aeroplane's flight path for the gust to reach its peak velocity. The identified gusts were used in the design of new aircraft, although this approach was not able to identify the real air turbulence [13,14]. Later, in the 60s and 70s, a new approach based on frequency response functions was proposed. Its use was limited due to its complexity in defining the frequency response functions [13]. In 1999, a Monte-Carlo and statistical analysis flight gust loads analysis approach was proposed by Kim et al. [15] in the context of space and missile systems. This procedure was based on forcing functions derived by extracting measured wind profiles' turbulent components. The application of the method was limited to gust wavelengths greater than 500 ft (152 m). In 2009, a model-based method which included information from an observer for a non-linear aircraft model, onboard measured data, parameters available on commercial aircraft and a disturbance model for the estimation of gusts and structural loads was proposed. Non-linear parameter optimisation was used to estimate the gust profile [16]. Recently, Balatti et al. [17] proposed a method based on cubic B-spline functions for the problem of gust and turbulence identification using simulated in-flight data. Results showed the robustness of this method. Indeed, gust response data were created using a detailed model, and the identification was performed using a simplified model.

Different ways to identify gust loads have been investigated in recent decades, but primarily two approaches have been considered, namely the optimisation method and the direct method [11]. In the optimisation method, the gust response is used in an optimisation framework, and the gust is tuned until the model responses match the measured responses. In the direct method, the gust is calculated using the gust response as input of the inverted aircraft model. However, the creation of an inverted model is complicated and typically associated with ill-conditioned problems. To solve this, machine learning and DL techniques have been proposed to replace physical models with a model derived directly from the data without requiring a substantial understanding of the underlying physics. Because of this, various machine learning and DL techniques have been utilised to identify linear and nonlinear systems based on measurements.

For example, Zhou et al. [18] applied Long Short Term Memory (LSTM) to the problem of load identification in three different nonlinear cases and showed the ability of LSTM to identify the disturbance even when its location is unknown. Moreover, they showed the low sensitivity of the hyperparameters of the recurrent neural network in the identification accuracy.

Antonakis et al. [19] employed a Convolutional Neural Network (CNN) methodology to perform dynamic modelling and identifying gusts using recorded in-flight data. To achieve this, a neural network method was employed with in-flight data recordings [19]. As it is not feasible to compare the identified gust with a direct measurement, the von Karman turbulence model was used to confirm a similar frequency content. In recent years, numerous researchers have been working on wind estimation in atmospheric turbulence using small Unmanned Aerial Systems (sUAS) [20,21]. Indeed, turbulence is a crucial factor in sUAS, which operates at lower airspeed and altitude with respect to aircraft. Generally, indirect measurements are preferred over direct ones. Allison et al. [20] used LSTM to address the problem of turbulence identification using a small unmanned aerial system. Simulated in-flight data using the Dryden gust model and realistic large eddy simulations were used to train the model. The results revealed the ability of the LSTM model to predict turbulence fields. Zimmerman et al. [21] utilised in-flight data obtained from a multirotor drone flying close to an anemometer to collect data in order to validate and test models for wind estimation. The authors evaluated three models: an LSTM neural network, an artificial neural network, and a Gaussian process regression. While all three models showed similar results, the LSTM model had the highest performance.

Although the literature presents several strategies for gust identification, the impossibility of directly measuring gusts poses a challenge to validating these identification strategies. Moreover, the studies considering DL models are limited. This paper aims to extend their use by considering: (a) aeroelastic models with different levels of accuracy and (b) experimental measurements in a controlled and known environment to validate the DL model. To do this, two aeroelastic models of civil and commercial aircraft, with different levels of accuracy, are used to generate gust responses for training and testing the DL models. The developed DL models are tested considering five different cases of increasing complexity. Finally, the DL model demonstrating the highest performance is applied to experimental measurements of a wing in the wind tunnel subjected to known gusts. In Section 2, the theory and framework employed in this paper for aeroelastic gust load identification are described. Within this section, the tools required for this framework, which include aeroelastic modelling (2.1), experimental aeroelastic testing (2.2), deep learning (DL) methods (2.3), and hyper-parameter optimisation (2.4), are explained. In Section 3 in-flight gust response data are created for five cases, and the DL models are developed. The gust identification results are reported in Section 4 before presenting the conclusions.

2. Theory: aeroelastic gust load identification

The theory of aeroelastic gust load identification is introduced in this section. The theory explains how gust loads applied to aircraft structures can be identified using physical/numerical aeroelastic responses, physical/numerical models of the structure, and DL methods. The aeroelastic response includes various outputs such as Center of Gravity (CG) heave acceleration, vertical tip displacements etc. These outputs are directly measurable, whereas gust loads cannot be directly measured. Having an accurate estimation of gust loads from measurable flight data is important, as gust loads play a crucial role in the aircraft design process [22]. In many real-world applications, directly measuring input loads applied to a structure is not feasible. However, it is possible to estimate these loads by analyzing information from measurable quantities. Examples of these methods can be seen in [18,23].

In general, an inverse problem consists of either identifying the characteristics of a system under a known input and output or identifying the input of a known system by measuring its response. The system can be either a numerical model or a physical structure. This paper demonstrates that a DL-based model can replace the conventional model or physical structure, and be utilised for identification purposes. In science and engineering, many problems have the form of a Fredholm integral equation of the first kind, which after discretization, are typically ill-posed or strongly ill-conditioned [24]. Assuming zero initial conditions of displacement and velocity, i.e., $z(0) = 0$ and $\dot{z}(0) = 0$ and linearity of the models, the time-domain convolution integral between the impulse response function $h(t)$ and the exciting gust $w_g(t)$ is

$$z(t) = \int_0^t h(t-\tau) w_g(\tau) d\tau \tag{1}$$

where the kernel function $h(t-\tau)$ is a convolution-type kernel and τ is the time delayed operation satisfying $t \geq \tau$. The kernel function $h(t-\tau)$ can be calculated in the frequency domain as the transfer function between the input and the measurement of interest and converted to the time domain by using the inverse Fast Fourier transformation [17].

In real applications, Eq. (1) is discretized over the interval $[0, t]$. Indeed, considering n sampled points over that interval, Eq. (1) can be expressed in matrix form as

$$\mathbf{z} = \mathbf{H}\mathbf{w}_g \tag{2}$$

where $\mathbf{H} \in \mathbb{R}^{n \times n}$ is a transfer matrix, and $\mathbf{z} \in \mathbb{R}^{n \times 1}$ and $\mathbf{w}_g \in \mathbb{R}^{n \times 1}$ are vectors composed of discrete values of the response and gust, respectively. To determine $\mathbf{w}_g$ when $\mathbf{z}$ is available and $\mathbf{H}$ is known and non-singular, Eq. (2) can be inverted, to give

$$\tilde{\mathbf{w}}_g = \mathbf{H}^{-1}\mathbf{z} \tag{3}$$

where $\tilde{\mathbf{w}}_g$ is the identified gust. However, inverse problems are typically ill-posed and sensitive to the inversion of the transfer matrix. To deal with this problem, cubic B-spline functions were used to regularise $\mathbf{H}$ and improve the identification robustness [17]. Indeed, the gust can be expressed as a summation of cubic B-spline functions as

$$\tilde{\mathbf{w}}_g = \mathbf{\Psi}\mathbf{c} \tag{4}$$

where $\mathbf{\Psi} \in \mathbb{R}^{n \times m}$ is a known matrix consisting of m cubic B-spline basis functions and $\mathbf{c} \in \mathbb{R}^{m \times 1}$ is a vector containing the B-spline scaling factors. Equation (2) can be expressed as

$$\mathbf{z} = (\mathbf{H}\mathbf{\Psi})\mathbf{c} \tag{5}$$

where the only unknown is $\mathbf{c}$ and inverting $(\mathbf{H}\mathbf{\Psi})$ the gust can be identified.

When the assumptions of the linear system are not satisfied, such as in a non-linear aeroelastic system, the Duhamel integral cannot be applied. However, in such cases, there must exist a non-linear function $\mathcal{F}$ that relates the input gust w_g to the measurement z

$$z(t) = \mathcal{F}\left[w_g(t)\right] \tag{6}$$

Suppose there exists an inverse function $\mathcal{M}$ that relates the measurement z to the exciting gust w_g as

$$\mathcal{M}\left[z(t)\right] = \tilde{w}_g(t) \approx w_g(t) \tag{7}$$

with

$$\varepsilon = |\tilde{w}_g(t) - w_g(t)| \tag{8}$$

where ε is the error between the identified gust $\tilde{w}_g$ and the real gust w_g. Equation (7) indicates that solving the gust identification problem boils down to determining $\mathcal{M}$. The $\mathcal{M}$ can be derived either from a physical model and/or a physical structure. However, this paper demonstrates the use of DL methods, such as LSTM and CNNs, for the identification of $\mathcal{M}$. To this end, the DL methods should be employed in conjunction with an aeroelastic model or physical structure. Therefore, the following subsections describe the aeroelastic modelling process, experimental aeroelastic testing, and the theory behind the two DL methods used in this paper. The framework of the gust identification theory introduced in this paper can be outlined as follows:

- Generate possible input gust profiles (see Section 2.1.1 'Gust and Turbulence Models'). In the wind tunnel, these gust profiles can be generated using the gust generator (see Section 2.2).
- Conduct experiments (2.2) and/or run numerical aeroelastic models (2.1) to generate gust response data, such as the centre of gravity (CG) heave acceleration or vertical tip displacements.
- Using the generated data, train DL models for inverse gust identification (training process explained in Sections 2.3 and 2.4).
- Employ the trained DL models for gust identification.

2.1. Aeroelastic modelling

Aeroelasticity is the science that investigates the mutual interaction between aerodynamics, elastic, and inertia forces and their impact on the static and dynamic aircraft structural response. Depending on the analysis required and the design stage models with different levels of accuracy can be developed. The complexity of the model is determined by the type of aircraft configuration, the type of dynamic simulation required, the flight envelope of the aircraft, and finally, the stage in the design process. Aeroelastic models can be categorised into simplified and detailed models. During the preliminary design of an aircraft, when limited information is available, a simplified model can be used to obtain relevant information about aircraft's handling qualities, stability, and flight performance. Moreover, for an elastic aircraft, a numerical model of the aeroelastic system serves as the starting point for designing and analysing an active or passive GLA system. Indeed, model reduction techniques are necessary when working with a detailed model. Alternatively, a simplified model can be used directly. Regardless of the approach taken in defining the aeroelastic model, the equation of motion can be formulated in terms of the physical displacement $\mathbf{y} \in \mathbb{R}^{\mathbf{h\times 1}}$ or modal coordinates $\mathbf{q} \in \mathbb{R}^{\mathbf{p\times 1}}$ using the relation

$$\mathbf{y} = \mathbf{\Phi q} \tag{9}$$

where h represents the number of degrees of freedom in physical coordinates, p denotes the number of modal coordinates (dependent on the number of modes retained for the analysis), and $\mathbf{\Phi} \in \mathbb{R}^{h\times p}$ is the modal matrix. The reduced order aeroelastic equation of motion representative of an aircraft subjected to gusts can be expressed as

$$\mathbf{A}\ddot{\boldsymbol{\zeta}} + \left(\rho V\mathbf{B} + \mathbf{D}\right)\dot{\boldsymbol{\zeta}} + \left(\rho V^2\mathbf{C} + \mathbf{E}\right)\boldsymbol{\zeta} = \mathbf{f}w_g \tag{10}$$

where $\mathbf{A} \in \mathbb{R}^{\mathbf{q\times q}}, \mathbf{D} \in \mathbb{R}^{\mathbf{q\times q}}$ and $\mathbf{E} \in \mathbb{R}^{\mathbf{q\times q}}$ are respectively the structural inertia, damping and stiffness matrices, $\mathbf{B} \in \mathbb{R}^{\mathbf{q\times q}}$ and $\mathbf{C} \in \mathbb{R}^{\mathbf{q\times q}}$ are the aerodynamic damping and stiffness matrices which are functions of the Mach number and the reduced frequency, $\boldsymbol{\zeta} \in \mathbb{R}^{q\times 1}$ is the vector of the generalized coordinates and $\dot{\boldsymbol{\zeta}}$ and $\ddot{\boldsymbol{\zeta}}$ are its first and second derivatives, ρ is the air density and V is the aircraft velocity [2]. On the right-hand side of Eq. (10), the forcing term $\mathbf{f}$ is the force vectors associated with the gust, and w_g is the gust profile. Equation (10) can be solved in the time or frequency domain.

2.1.1. Gust and turbulence models

Atmospheric disturbance models are commonly classified into two idealized categories: discrete gusts and continuous turbulence. A discrete gust is typically modelled as a '1 - cosine'function due to the shape of the gust, which is described mathematically as

$$w_g(t) = \begin{cases} \frac{w_{g_0}}{2}\left[1 - \cos\left(2\pi\frac{V}{l_g}t\right)\right] & \text{for} \quad 0 \le t \le \frac{l_g}{V} \\ 0 & \text{for} \quad t > \frac{l_g}{V} \end{cases} \tag{11}$$

where w_{g_0} is the maximum gust velocity and l_g is the gust wavelength. According to the European Aviation Safety Agency (EASA) regulation [10] for the case of civil and commercial aircraft, gust wavelengths are varied between 18 m to 214 m, and the gust velocity is calculated as

$$w_{g0} = w_{ref}\left(\frac{H}{106.14}\right)^{1.6} \tag{12}$$

where the gust gradient H is half the gust wavelength l_g and the reference gust velocity w_{ref} reduces linearly from 17.07 m/s Equivalent Air Speed (EAS) at sea level to 13.41 m/s EAS at 4572 m, and then again to 6.36 m/s EAS at 18288 m. Fig. 1 shows gusts at different wavelengths at sea level for an airspeed of 200 m/s.

According to the EASA regulations, the power spectral density of atmospheric turbulence is described by the von Karman spectra as

$$\Phi_v(\Omega) = \frac{L_T}{\pi}\frac{1 + \frac{8}{3}(1.339\Omega L_T)^2}{[1 + (1.339\Omega L_T)^2]^{11/6}} \tag{13}$$

where Ω is the spatial frequency, L_T is the scale of turbulence (commonly assumed to be 2500 ft). According to Hoblit [3], the turbulence velocity time history is obtained as the output of a shape filter with the input given by a stationary Gaussian 'white-noise' time history. The transfer function that approximates the von Karman shape is

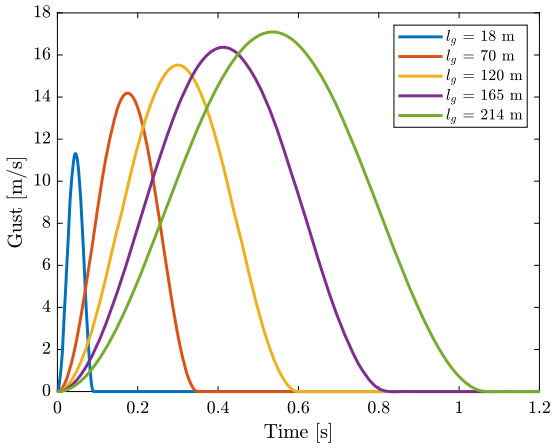

Fig. 1. Gusts with different gust wavelengths.

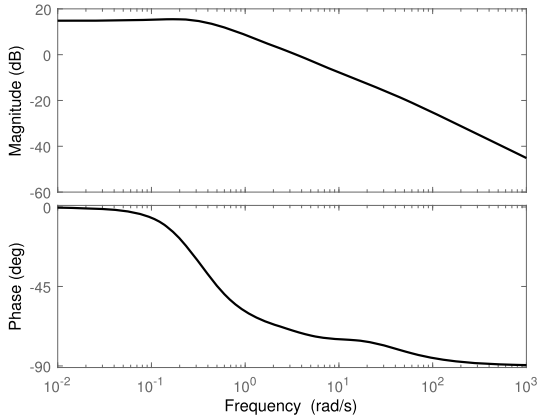

Fig. 2. Bode diagram of the von Karman approximation of atmospheric turbulence.

$$G(s) = \frac{\sigma_w}{\sqrt{\Phi_\eta}} \frac{\sqrt{\tau_T}}{\pi} \frac{(1 + 2.187\tau_T s)(1 + 0.1833\tau_T s)(1 + 0.021\tau_T s)}{(1 + 1.339\tau_T s)(1 + 1.118\tau_T s)(1 + 0.1277\tau_T s)(1 + 0.0146\tau_T s)} \tag{14}$$

where Φ_η is the power spectral density of the white noise, τ_T is the ratio between L_T and the horizontal velocity of the aircraft and σ_w is the component of the gust velocity. Fig. 2 shows the Bode diagram of the approximation of the von Karman turbulence model. The atmospheric turbulence is obtained in the time domain as the output of the state-space form of the transfer function Eq. (14) whose input is white noise.

2.2. Experimental aeroelastic testing on physical structures

Wind tunnels equipped with various sensors may be utilised for aeroelastic testing. In the context of gust load identification in this paper, the purpose of this aeroelastic testing is to gather data to train DL models. This particular aeroelastic testing requires a gust generator. A gust generator, capable of producing both discrete and continuous gusts, was commissioned and characterised at Swansea University's wind tunnel [25]. Different gust profiles, such as 1-cos with varying lengths and velocities and sinusoidal gusts, are considered. The wing response, such as vertical tip displacement, is measured using laser sensors, as detailed by the authors in [26]. Consequently, validated aeroelastic models, in conjunction with data from these sensors, can be utilized to train deep learning models for the identification of gust inputs, as described in subsequent sections. Fig. 3 shows the gust generator, the wing installed in the Swansea University wind tunnel, the balance and the laser displacement sensors [9]. For details on experimental aeroelastic testing, readers are referred to [25,26].

2.3. Deep learning models

CNNs are feed-forward neural networks commonly used for two-dimensional image recognition. CNNs' architecture allows for the use of training kernels, which are small matrices that are multiplied and averaged with all of the patches in the image. The key advantage of this model is that it accurately captures local features such as corners and basic shapes. If a one-dimensional time

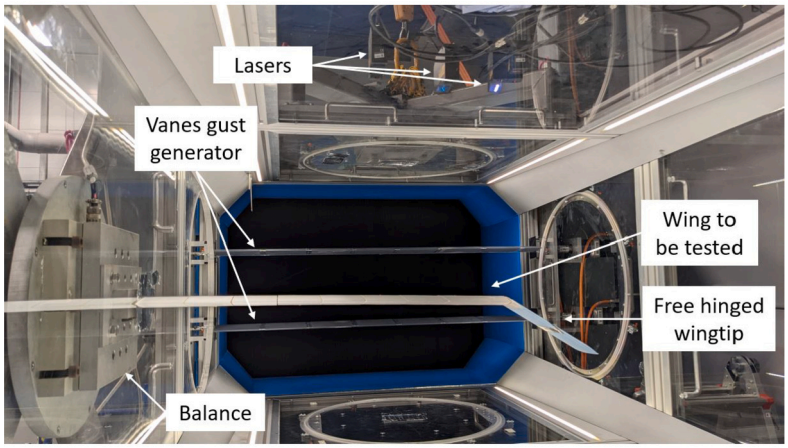

Fig. 3. Wing installed in the wind tunnel.

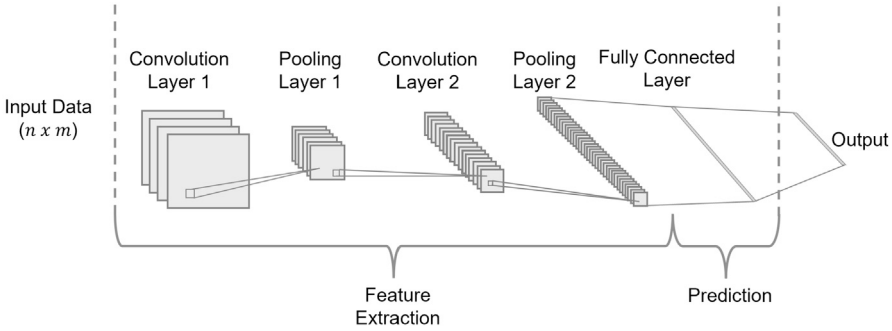

Fig. 4. Typical structure of CNN.

signal is transformed directly into a two-dimensional signal, the spatial correlation in the original correlation is eliminated, and some original information may be lost. So, in the context of this research, for the one-dimensional signals, one-dimensional CNNs are used to uncover patterns in the temporal dimension, such as local maxima, minima, rapid changes, and among others. A typical structure of CNN is shown in Fig. 4. The input layer takes the input vector and creates a feature map corresponding to the convolution kernel. The feature map is then passed on to the next layer using a set of weights. A receptive field, which is a square matrix of weights with sizes smaller than the input, establishes the connection between the convolution layer and the input. The convolution procedure is carried out as the receptive field strides (convolves) along the input area. For a given input $x(t) = \{x_1, x_2, \ldots, x_t\}$ and output $y(t) = \{y_1, y_2, \ldots, y_t\}$, the convolution operation is described as

$$y_{ij} = \sigma\left(\sum_{r=1}^{F}\sum_{c=1}^{F} w_{rc} x_{(r+i\times S)(c+j\times S)} + b\right), \quad 0 \le i \le \frac{H-F}{S}, \quad 0 \le j \le \frac{W-F}{S}, \tag{15}$$

where y_{ij} indicates a node's output on the feature map, and σ is a nonlinear activation function used to extract the features from the input. The height and width of the receptive field are denoted by F, while S represents the stride length. w_{rc}, $x_{(r+1\times S)(c+j\times S)}$ and b represent the weights, input data element, and bias, respectively. H and W stand for the input's height and width dimensions. In convolutional layers, the input size decreases depending on the number of filters. The pooling layer limits overfitting and minimises the input layer's spatial dimension by 75% [27].

LSTM models are a type of recurrent neural network that has lately shown promise in time series forecasting. The key characteristic of the LSTM architecture is a system that allows it to discard irrelevant information during learning while maintaining track of long-term dependencies.

A basic structure of the LSTM memory cell is presented in Fig. 5. In principle, an LSTM cell maintains two states: long-term (c_t) and short-term (h_t), which are updated as new data becomes available. A typical memory cell contains three control gates, input, output, and forget gates, performing the writing, reading and resetting functions in each cell. The multiplicative gates enable the model to hold information over long time periods, avoiding the vanishing gradient problem seen in Recurrent Neural Networks [28]. The following set of equations allows the LSTM to predict the output vector.

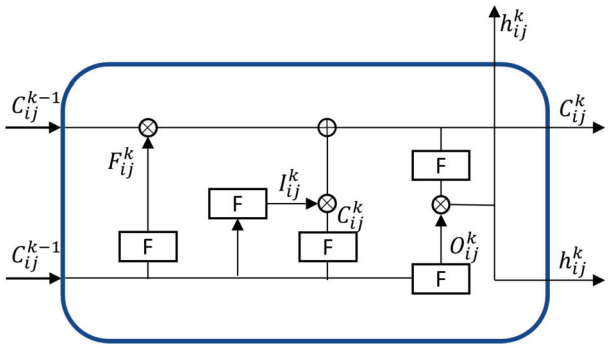

Fig. 5. Basic structure of the LSTM layer.

$$
\begin{aligned}
i_t &= \sigma\left(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i\right)\\
f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right)\\
c_t &= f_t c_{t-1} + i_t g\left(W_{xc}x_t + W_{hc}h_{t-1} + b_c\right)\\
o_t &= \sigma\left(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o\right)\\
h_t &= o_t h\left(c_t\right)
\end{aligned}
\tag{16}
$$

where i_t, f_t, c_t and o_t describe the input, forget gate, cell activation, and output, respectively. σ denotes the activation function, and W and b represent the weight matrix and the bias vector.

2.4. Hyper-parameter optimisation

Machine learning and DL methods often require the tuning of specific parameters known as hyper-parameters. These hyper-parameters are optimised during the training of DL algorithms since they have a critical impact on the algorithm's performance. Choosing appropriate hyperparameter values is crucial for building a strong and accurate model, as selecting inappropriate values can result in the model overfitting or underfitting the data. It is usual practice to display the model loss as a function of the number of epochs. This graphic can tell whether the model is overfitting, underfitting, or appropriately fitting the training dataset [29]. In addition, stopping criteria are implemented during the training to stop the training when the performance measure stops improving [30].

For the hyper-parameters of the CNN model, the number of filters and kernels, layers, learning rate, and epochs are considered. For the LSTM model, the number of neurons, layers, learning rate, and epochs are considered.

3. Data generation and deep learning modelling

This section outlines the process of in-flight data creation and DL modelling. In this work, two aeroelastic models representative of a civil jet aircraft with different levels of fidelity [17,22], and a wing model representative of one tested in the wind tunnel [26], are considered. As all the aeroelastic models discussed are available in the literature, only a concise summary is provided in this paper. The first model is a low-fidelity model and represents a flexible aircraft and consists of a rigid fuselage, a rectangular, uniform, untapered, unswept flexible wing with a hinged wingtip and a tailplane [7]. The analytical model was developed to have the least degrees of freedom and was used to perform multi-objective optimisation. In this case, the aeroelastic equation of motion is solved using the explicit Runge-Kutta method. The second model is a high-fidelity model, developed using a 'stick' model with lumped masses and double lattice panels. The second model was developed to investigate the effect of hinged wingtips [31]. This model is a modification of the FFAST (Future Fast Aeroelastic Simulation Technologies) aeroelastic model and includes several thousand elements, which are reduced using modal reduction to retain 55 modes for analysis [22]. Table 1 presents the key parameters for the two aircraft aeroelastic models under consideration [7,17,32]. Figs. 6 and 7 show the typical aircraft centre of mass heave vertical acceleration gust responses for the low-fidelity model and the high-fidelity model, respectively and used for training of DL models.

The wing physical structure consists of an aluminium spar with an "x-shaped" cross-section and 3D-printed aerofoils [6,9,26]. The wing considered in this work was subjected to discrete gusts with different gust lengths. The laser displacement sensor was used to measure the wing vertical displacement at 80% of the wing. Fig. 8 shows a case of an experimentally measured wing vertical displacement in response to a 1-cos gust.

The FFAST model and the aeroelastic model of the wing tested in the wind tunnel are solved in the frequency domain and the Fast Fourier transform is used to calculate the time domain response.

To assess the DL models across varying levels of complexity, five distinct cases of in-flight data were examined. These cases were selected to range from academic scenarios (involving discrete gusts at a single flight configuration) to more industrially relevant scenarios (involving turbulence and discrete gusts at any flight configuration). In all cases, ten thousand gust responses were utilized, employing Latin hypercube sampling to vary the gust length between 18 m and 214 m, and the maximum discrete gust amplitude

Table 1
Main aircraft parameters.

Half span	32.5 m
Total mass	187429 kg
Airspeed	200 m/s
Wing inertia	12083 kg m^2
Aircraft inertia	12425757 kg m^2
Mean Chord	4 m
Wingtip Mass	500 kg
Wingtip span	6.5 m
Engine weight	1680 kg

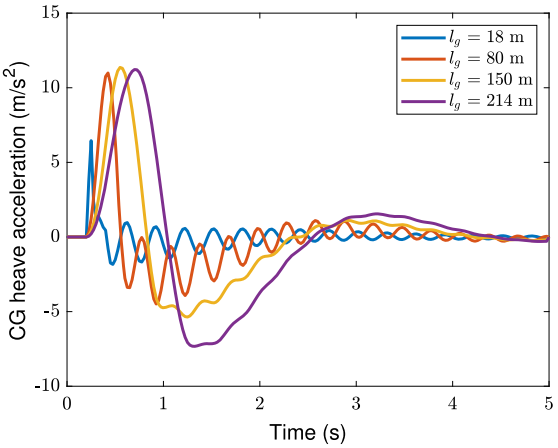

Fig. 6. Low-fidelity model, aircraft centre of mass vertical acceleration gust responses.

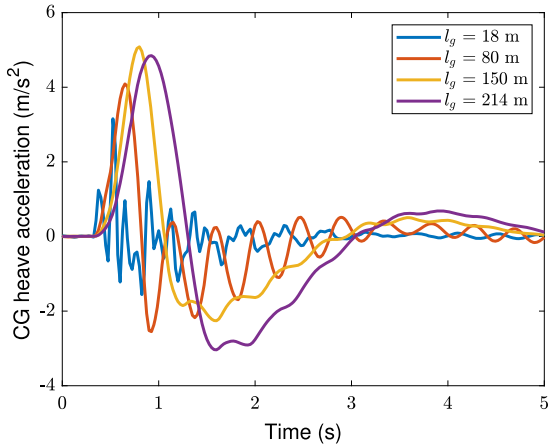

Fig. 7. High-fidelity model, aircraft centre of mass vertical acceleration gust responses.

between 2 m/s and 18 m/s. The gusts and aircraft centre of mass acceleration time histories were recorded with a time step of 25 ms.

3.1. In-flight data creation

The in-flight data were generated by solving Eq. (10) either in the time or the frequency domain, with the resulting response converted to the time domain using the inverse Fast Fourier transformation if the frequency domain approach was used. The first case considered the simplified model flying at 200 m/s and sea level, while the second case considered the detailed model flying at 200 m/s and sea level. In the third, fourth, and fifth cases, 104 flight configurations within the envelope of a civil, commercial aircraft were defined, as shown in Fig. 9. Latin hypercube sampling was used to select the flight configuration for each gust response in the third and fourth cases. Furthermore, the information regarding the flight configuration was not saved in the third case and was saved in the fourth. The fifth case was a more realistic case, where a discrete gust and turbulence event were considered together. For

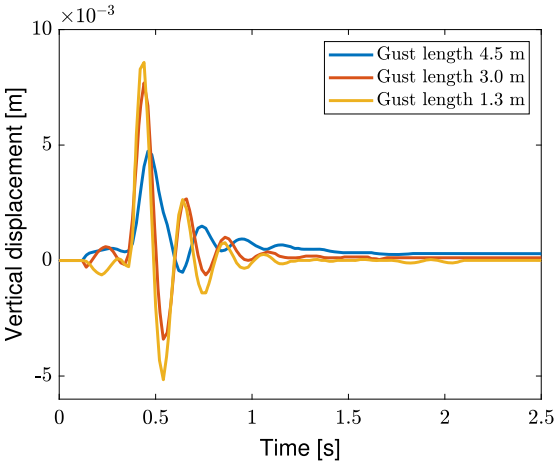

Fig. 8. Experimental measurements vertical displacement gust response.

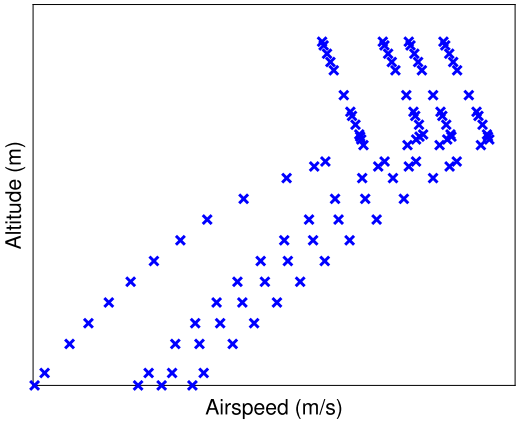

Fig. 9. Flight configurations considered.

each gust response, ad-hoc turbulence was created. In the first four cases, time histories of 5 seconds were considered. To consider the turbulence event before and after the discrete gust, in the fifth case, 15 seconds were considered.

3.2. Simulated wing gust response

Simulated experimental gust response measurements were generated by solving Eq. (10) in the frequency domain and converted to the time domain using the inverse Fast Fourier transformation. Ten thousand gust responses were considered at 18 m/s and sea level, using Latin hypercube sampling to vary the gust length between 1.3 m and 4.5 m and, based on the measured gusts, varying the amplitude between 0 m/s to 1 m/s. Gusts and vertical displacement at 80% wing span time histories were saved, considering a sampling time of 5 ms.

3.3. Deep learning modelling

In several applications, CNN and LSTM were individually used for time series modelling [33,34]. For example, LSTM was used in the context of nonlinear structural systems to predict impact loads and load location [18]. However, CNN-LSTM has been used to tackle many complex applications in manufacturing, finance, meteorology, etc., especially when dealing with time series data [27,35–37]. The CNN-LSTM model takes advantage of the convolutional layers' ability to extract meaningful knowledge and learn the internal representation of time-series data, as well as the LSTM layers' efficiency in distinguishing short-term and long-term time dependencies.

As a result, the CNN-LSTM model was primarily chosen for the gust identification task. However, considering that LSTM is predominantly helpful with temporal relationships, for the gust identification task, the time dependencies stay zeros until and after the gust emerges with sharp edges and shapes. Since the CNNs can accurately capture this behaviour, thus, a purely CNN model is also explored for the gust identification task.

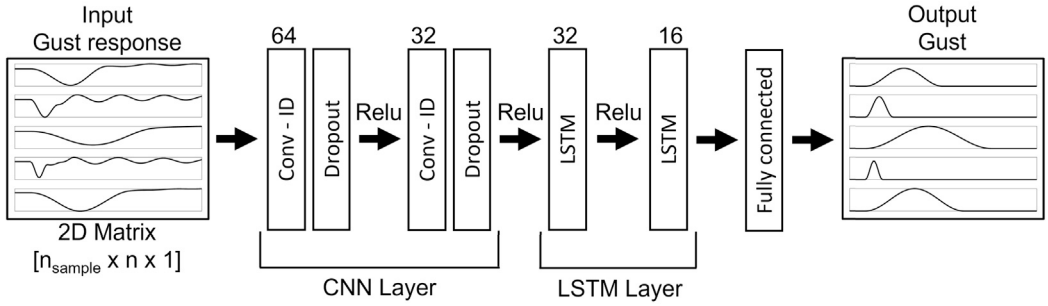

Fig. 10. Hybrid CNN-LSTM model.

Table 2
The hyperparameters of the models.

Model	Value
Dropout	0.1, 0.2, 0.3
Learning rate	10^{-3}, 10^{-3}, 10^{-5}
Batch size	16, 32, 64, 128
Epochs	stopping criteria with patience 10

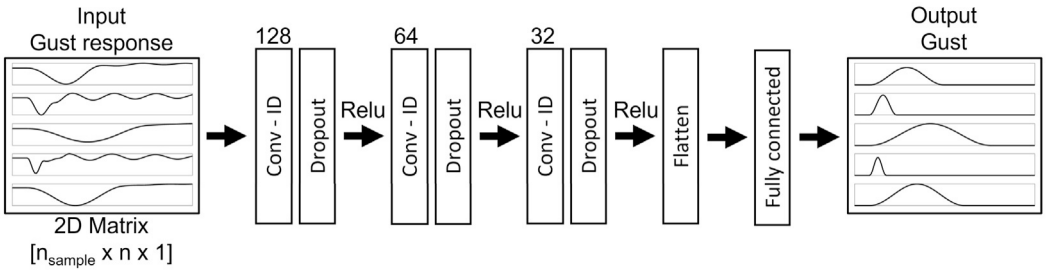

Fig. 11. CNN model.

Initially, the two sub-models (CNN-LSTM) are integrated by adding CNNs at the front end for feature extraction, followed by LSTM layers with a Dense layer on the output to interpret the features across time steps. Fig. 10 shows the architecture of the hybrid CNN-LSTM model utilised in this work. Two stacked 1-dimensional convolutional filters are applied in the front end; then, two layers of LSTM layers are integrated into the model, and a dropout layer is added in-between the CNN and LSTM layers to reduce the overfitting problem during model training. Finally, after adding another dropout layer, a fully connected layer is added to perform the gust identification task.

The performance of DL models is determined by predefined hyperparameters obtained through an optimization procedure. For the CNN-LSTM model used in this study, the number of layers, filters, neurons, and activation function are determined based on prior studies [35,38,39], and their values are shown in Fig. 10. Subsequently, for the CNN model used in this study, the architecture and values of hyperparameters, number of layers, filters, neurons and activation function are shown in Fig. 11.

Other parameters, such as the dropout rates, learning rate, batch size, and activation functions, are adjusted using a grid search. Table 2 shows the values used in the grid search to adjust the hyperparameter. Finally, the number of epochs is determined based on the stopping criteria when sufficient convergence of training and validation has been met with specified patience.

For the validation of the model, the standard holdout validation strategy is followed [35,40] by dividing the dataset into 80% and 20%, respectively for the training and test dataset. From the training dataset, 10% of the dataset is randomly selected for validation during model training, whereas 20% of the test dataset is set aside for final evaluation after model training to assess the model's performance on unseen data.

3.3.1. Model evaluation metrics

In this work, the Mean Squared Error (MSE) is used as the loss function during model training. The model parameters are updated by computing and backpropagating the MSE over the training data for each iteration (epoch). The MSE is defined as

$$\mathrm{MSE} = \frac{1}{n_t}\frac{1}{n}\sum_{i=1}^{n_t}\sum_{j=1}^{n}\left(\tilde{w}_{g_j}(t_i) - w_{g_j}(t_i)\right)^2, \tag{17}$$

Table 3
Summary of cases considered.

	Case 1	Case 2	Case 3	Case 4	Case 5
Modelling fidelity	Low	High	Low	Low	Low
Flight configuration	1	1	104	104	104
Airspeed and altitude	No	No	No	Yes	Yes
Disturbance	'1-cos'	'1-cos'	'1-cos'	'1-cos'	'1-cos'+turb.

Table 4
Gust identification results using cubic B-spline functions.

	Case 1	Case 2 (5 modes)	Case 2
R^2	0.9931	0.9974	0.9874
RMSE	0.0024	0.0015	0.0035
MAE	0.0150	0.0037	0.0342

where $w_{g_j}(t_i)$ represents the j-th reference gust at time step t_i, $\tilde{w}_{g_j}$ represents the j-th identified gust at time step t_i, n_t is a total number of time steps and n is the number of samples i.e., the number of gusts in the test dataset.

To measure the performance of the trained model, the following three metrics are utilised: the coefficient of determination R^2, the Root Mean Squared Error (RMSE), and the Mean Absolute Error (MAE). R^2 is the proportion of variance in the target variable, which is explained by the input variable. The R^2 ranges between 0 and 1, where 1 represents that the input variable perfectly explains the target variable and 0 is vice versa. R^2 of 0.9 signifies that the input variables can explain 90% of the variance in the target variable. Many deep learning papers have employed R^2 as a performance metric because it offers insights into how well the model fits the dataset [35,40–43].

$$R^2 = 1 - \frac{SS_{RES}}{SS_{TOT}} \tag{18}$$

where SS_{RES} is residual sum of squares, defined as

$$SS_{RES} = \frac{1}{n_t}\sum_{i=1}^{n_t}\sum_{j=1}^{n}\left(w_{g_j}(t_i) - \tilde{w}_{g_j}(t_i)\right)^2,$$

SS_{TOT} is total sum of squares, defined as

$$SS_{TOT} = \frac{1}{n_t}\sum_{i=1}^{n_t}\sum_{j=1}^{n}\left(w_{g_j}(t_i) - \bar{w}_{g_j}(t_i)\right)^2,$$

$\bar{w}_{g_j}(t_i) = \frac{1}{n}\sum_{i=1}^{n} w_{g_j}(t_i)$ is mean of observed data. The RMSE measures how far the model's predicted gusts are from the actual gusts and is defined as

$$\text{RMSE} = \frac{1}{n_t}\sum_{i=1}^{n_t}\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\tilde{w}_{g_j}(t_i) - w_{g_j}(t_i)\right)^2} \tag{19}$$

The MAE is the measure of the absolute difference between the predicted and the actual values and is defined as

$$\text{MAE} = \frac{1}{n_t}\frac{1}{n}\sum_{i=1}^{n_t}\sum_{j=1}^{n}|\tilde{w}_{g_j}(t_i) - w_{g_j}(t_i)| \tag{20}$$

4. Numerical gust identification results

The DL models presented in Section 3.3 were tested in different cases with increasing the level of complexity. In Cases 1 and 2, the results are compared to those obtained using the inverse method based on cubic B-spline functions [7]. The same model was used to generate the gust response and to perform the gust identification in both cases. Table 3 provides a summary of the aeroelastic model fidelity, flight configuration, airspeed and altitude knowledge, and disturbance considered for each case.

4.1. Results using cubic B-spline functions

To consider the effect of the model complexity, the identification using the cubic B-spline functions was performed considering the simplified and detailed models with 5, and 55 modes [7]. Table 4 displays the identification results using cubic B-spline functions. The simplified and detailed models with 5 modes displayed comparable performance in the identification process. However, as we increased the number of modes for the detailed model, the R^2 score decreases and the MAE increases.

Table 5
Gust identification results using the CNN-LSTM model.

	Case 1	Case 2	Case 3
R^2	0.9990	0.9465	0.9249
RMSE	0.0040	0.0303	0.0309
MAE	0.0019	0.0118	0.0118
Time	5 h 45 m 5 s	1 h 28 m 46 s	4 h 3 m 12 s

Table 6
Gust identification results using the CNN model.

	Case 1	Case 2	Case 3	Case 4	Case 5
R^2	0.9993	0.9987	0.9784	0.9958	0.8257
RMSE	0.0033	0.0046	0.0166	0.0148	0.1278
MAE	0.0020	0.0014	0.0122	0.0074	0.0988
Time	1 m 36 s	47.4 s	38 s	4 m 45 s	26 m 40 s

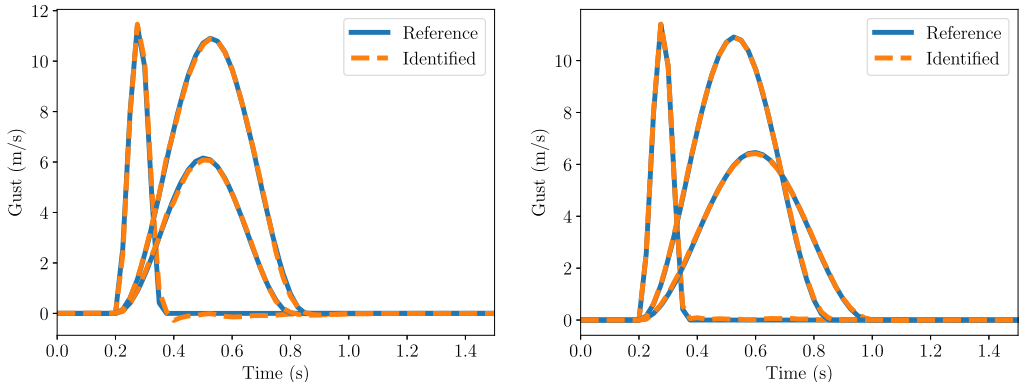

Fig. 12. Identification for Case 1 using the CNN-LSTM model (left) and CNN model (right); In each case, the gusts are randomly chosen from the test data pool.

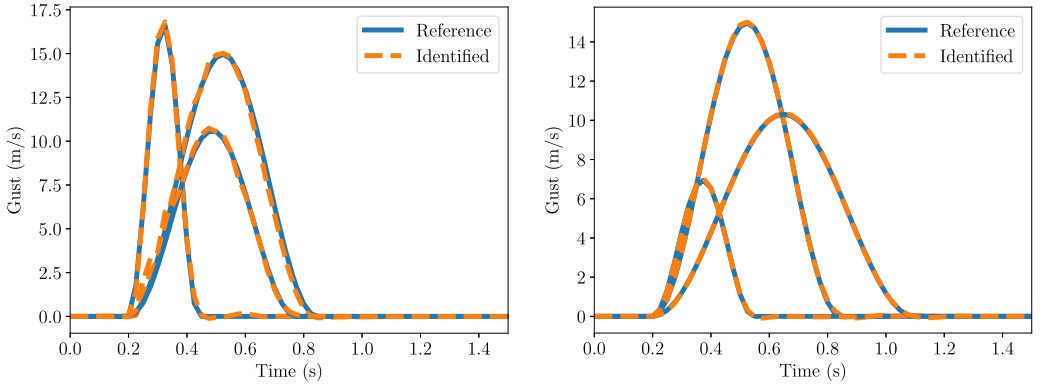

Fig. 13. Identification for Case 2 using the CNN-LSTM model (left) and CNN model (right); In each case, the gusts are randomly chosen from the test data pool.

4.2. Comparison of CNN-LSTM and CNN models for Cases 1, 2 and 3

Table 5 presents the identification results obtained using the CNN-LSTM model along with the corresponding training time.

The identification of Case 1 using cubic B-spline functions and the CNN-LSTM model gives similar results, but the CNN-LSTM model is computationally more expensive. Table 6 displays the identification results obtained using the CNN model, along with the corresponding training times. Notably, the training time for the CNN model is significantly reduced compared to that of the CNN-LSTM model.

Figs. 12, 13 and 14 depict the reference and the identified gusts for Cases 1, 2, and 3, respectively using both the CNN-LSTM and CNN model. For Cases 1 and 2, the identified gust closely resemble the reference gusts with only a small difference in Case 3 due to the complexity of considering 104 flight configurations. Overall, the performance of the identification task decreases as the

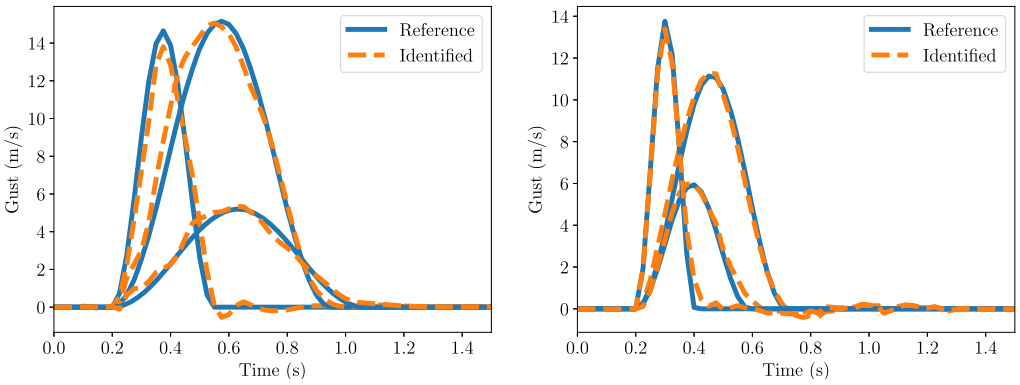

Fig. 14. Identification for Case 3 using the CNN-LSTM model (left) and CNN model (right); In each case, the gusts are randomly chosen from the test data pool.

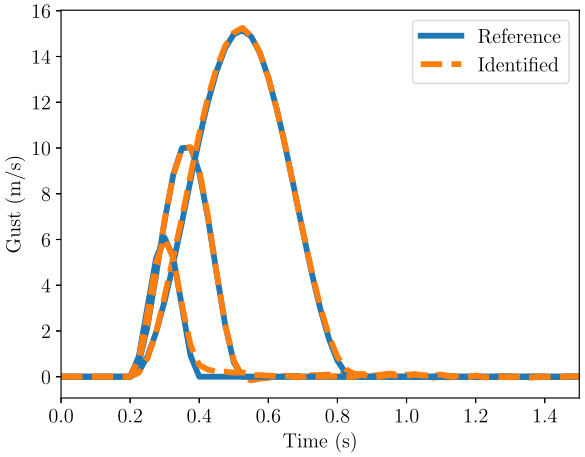

Fig. 15. Identification for Case 4 using the CNN model; In each case, the gusts are randomly chosen from the test data pool.

modelling difficulty increases from Case 1 to Case 3, which involves the shift from considering only one flight configuration to all flight configurations and, as well as from using a simplified model to a detailed model.

The CNN model outperforms the cubic B-spline functions method in identifying gusts at a single flight configuration, as observed in Cases 1 and 2. While the DL models and the cubic B-spline functions method yield comparable results in Cases 1 and 2, the DL models offer the advantage of using a single unified model for gust identification, irrespective of the aircraft's altitude and speed. In general, when comparing CNN and CNN-LSTM, CNN exhibited superior performance compared to the CNN-LSTM model. Indeed, upon scrutinizing the input and output data of the DL models, it becomes evident that the data are characterized by sharp edges, rather than relying solely on long-term time dependencies. This characteristic is effectively addressed by the CNN model.

4.3. CNN model for Case 4 and 5

Due to the lengthy training time required for CNN-LSTM model, experiments on cases 4 and 5 were run using only the CNN model. Fig. 15 shows the reference and the identified gusts for Case 4 using the CNN model. Introducing the information about flight configuration (Case 4) yields identification performances similar to that of a single flight configuration (Cases 1 and 2). With the B-spline method, each flight configuration requires a separate model, resulting in a collection of models for the entire flight configuration. In contrast, the advantage of using the DL model is to have a single, unified model for all fight configurations.

Fig. 16 shows the identified gust and turbulence model for Case 5. The maximum identification error is observed following the discrete gust, where the aircraft response is influenced by both discrete gusts and turbulence.

4.4. Experimental gust identification results

The above results indicate that, for the specific cases considered, the CNN model outperforms the CNN-LSTM model. Hence, in this subsection, the gust identification using experimental gust response was performed using the CNN model. The simulated wing gust responses from Section 3.2 served as training data for the CNN model. Table 7 presents the identification results and the associated training time. Wing gust responses from reference [9] were used to test the model and the identified gusts were compared with the previously measured gusts [25]. Fig. 17 depicts the reference and the identified gust for gust lengths of 2.25 m and 1.5 m.

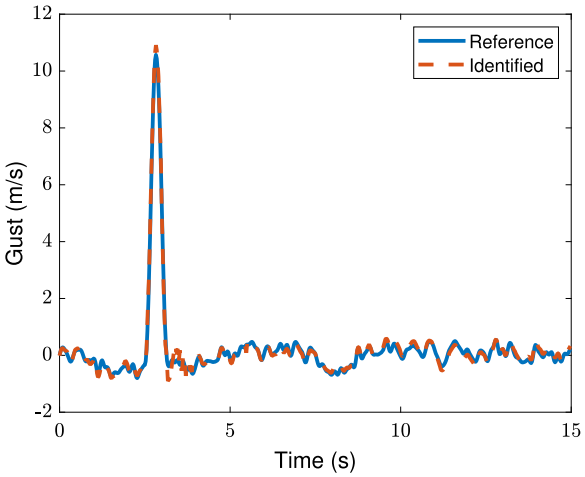

Fig. 16. Identification for Case 5 using the CNN model.

Table 7
Wing gust identification results using the CNN model.

R^2	0.9962
RMSE	0.00019
MAE	0.00012
Time	1 m 36 s

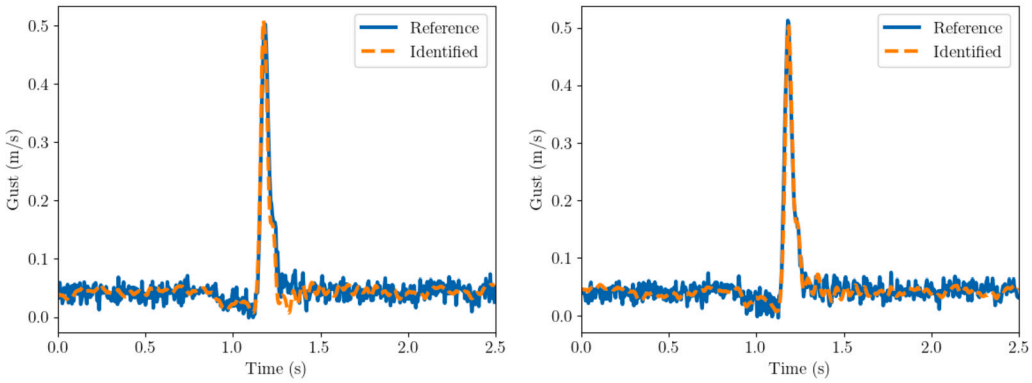

Fig. 17. Identification using the CNN model from wind tunnel experimental measurements: Discrete gust with a length of 2.25 m (left) and 1.5 m (right).

5. Conclusions

This study proposes the use of two DL models, a CNN-LSTM model and a CNN model, to identify gust and turbulence. The recently proposed gust identification system based on cubic B-spline functions, introduced by the authors of this paper [17] was employed as a reference. Two aeroelastic models with different levels of fidelity were employed to generate in-flight gust response measurements considering five different cases of increasing complexity. In the first two cases, discrete gusts at a constant flight configuration were analysed using the simplified and detailed models. The results demonstrate that the CNN model can identify the gust more accurately than the cubic B-spline functions. Similar performance was observed when using the CNN-LSTM model for the in-flight measurements generated by the simplified model. However, the CNN-LSTM model's performance was reduced when the detailed model was used. Overall, when comparing CNN and CNN-LSTM, CNN demonstrated better performance over the CNN-LSTM model. Despite the established success of LSTM models in similar tasks, their effectiveness is highly dependent on the specific case (data). Additionally, upon examining the input data (Figs. 6 and 7) and output (Fig. 1) of deep learning models, it becomes apparent that the emphasis is on capturing sharp edges rather than relying on long-term time dependencies, a characteristic effectively addressed by the CNN model.

The CNN model was evaluated using the entire flight envelope of civil and commercial aircraft. Introducing altitude and airspeed as input improved the discrete gusts identification accuracy. Additionally, the CNN model accurately predicted a disturbance caused by a real-life scenario with gust and turbulence acting together.

To the authors' knowledge, this study demonstrated the potential of DL models for the problem of gust and turbulence identification in civil aircraft for the first time. The results highlight the advantage of using a single model to accurately identify the disturbance in any flight configuration. Future works will investigate the robustness of the proposed DL method and the evaluation of the required training data. Furthermore, this work will be extended using real gust response measurements from wind tunnel experiments. Moreover, the measured data will be tested on the DL models created from the simulated data using the transfer learning method.

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.

Data availability

Data will be made available on request.

Acknowledgements

The research leading to these results has received funding from the Engineering Physical Science Research Council (EPSRC) through a program grant EP/R006768/1. The authors also acknowledge the EPSRC Impact acceleration fund.

References

[1] IATA, Our commitment to fly net zero by 2050, https://www.iata.org/en/programs/environment/flynetzero/, 2021. (Accessed 30 March 2023).
[2] Jan Robert Wright, Jonathan Edward Cooper, Introduction to Aircraft Aeroelasticity and Loads, vol. 20, John Wiley & Sons, 2008.
[3] Frederic M. Hoblit, Gust Loads on Aircraft: Concepts and Applications, American Institute of Aeronautics and Astronautics, 1988.
[4] Christopher D. Regan, Christine V. Jutte, Survey of applications of active control technology for gust alleviation and new challenges for lighter-weight aircraft, NASA TM-2012-216008, 04 2012.
[5] Dongqiang Zhao, Zhichun Yang, Xianang Zeng, Jinge Yu, Yining Gao, Guoning Huang, Wind tunnel test of gust load alleviation for a large-scale full aircraft model, Chin. J. Aeronaut. (ISSN 1000-9361) (2022).
[6] Davide Balatti, Hamed Haddad Khodaparast, Michael I. Friswell, Marinos Manolesos, Aeroelastic model validation through wind tunnel testing of a wing with hinged wingtip, in: International Forum on Aeroelasticity and Structural Dynamics, Madrid, Spain, 2022.
[7] Davide Balatti, Hamed Haddad Khodaparast, Michael I. Friswell, Marinos Manolesos, Mohammadreza Amoozgar, The effect of folding wingtips on the worst-case gust loads of a simplified aircraft model, Proc. Inst. Mech. Eng., Part G, J. Aerosp. Eng. 236 (2) (2022) 219–237.
[8] E. Rynaski, D. Andrisani II, B. Eulrich, Gust alleviation using direct turbulence measurements, in: 5th Atmospheric Flight Mechanics Conference for Future Space Systems, 1979, p. 1674.
[9] Davide Balatti, James D. Ellis, Shakir JiffrI, Hamed Haddad Khodaparast, Micheal I. Friswell, Active hinged wingtip for gust load alleviation and manoeuvres, in: AIAA SCITECH 2023 Forum, 2023, p. 2567.
[10] European Aviation Safety Agency, Certification specifications and acceptable means of compliance for large aeroplanes cs25, https://www.caa.co.uk/media/vkdl44xb/caa-cs-25-amendment-26-initial-airworthiness.pdf, 2020. (Accessed 30 March 2023).
[11] Simone Simeone, Andrea Da Ronch, T. Rendall, A gust reconstruction framework applied to a nonlinear reduced order model of a wing typical section, in: 58th AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2017, p. 0634.
[12] Eric J. Tuegel, Anthony R. Ingraffea, Thomas G. Eason, S. Michael Spottswood, Reengineering aircraft structural life prediction using a digital twin, Int. J. Aerosp. Eng. 2011 (2011) 1687–5966.
[13] John C. Houbolt, Atmospheric turbulence, AIAA J. 11 (4) (1973) 421–437.
[14] John C. Houbolt, Design manual for vertical gusts based on power spectral techniques, Technical report, Aeronautical Research Associates of Princeton Inc., NJ, 1970.
[15] M.C. Kim, A.M. Kabe, S.S. Lee, Atmospheric flight gust loads analysis, J. Spacecr. Rockets 37 (4) (2000) 446–452.
[16] H. Henrichfreise, L. Bensch, J. Jusseit, L. Merz, M. Gojny, Estimation of gusts and structural loads for commercial aircraft, in: International Forum on Aeroelasticity and Structural Dynamics (IFASD), Seattle, 2009.
[17] Davide Balatti, Hamed Haddad Khodaparast, Michael I. Friswell, Marinos Manolesos, Andrea Castrichini, Aircraft turbulence and gust identification using simulated in-flight data, Aerosp. Sci. Technol. 115 (2021) 106805.
[18] J.M. Zhou, Longlei Dong, Wei Guan, Jian Yan, Impact load identification of nonlinear structures using deep recurrent neural network, Mech. Syst. Signal Process. 133 (2019) 106292.
[19] Aristeidis Antonakis, Mudassir Lone, Alastair Cooke, Neural network based dynamic model and gust identification system for the jetstream g-nfla, Proc. Inst. Mech. Eng., Part G, J. Aerosp. Eng. 231 (6) (2017) 1138–1153.
[20] Sam Allison, He Bai, Balaji Jayaraman, Wind estimation using quadcopter motion: a machine learning approach, Aerosp. Sci. Technol. 98 (2020) 105699.
[21] Steven Zimmerman, Miayan Yeremi, Ryozo Nagamune, Steven Rogak, Wind estimation by multirotor dynamic state measurement and machine learning models, Measurement 198 (2022) 111331.
[22] H. Haddad Khodaparast, Jonathan E. Cooper, Rapid prediction of worst-case gust loads following structural modification, AIAA J. 52 (2) (2014) 242–254.
[23] A. Kawano, A. Morassi, Load identification in a plate-beam lattice from interior dynamic data, Appl. Math. Model. 125 (2024) 347–366.
[24] Janusz Mroczka, Damian Szczuczyński, Inverse problems formulated in terms of first-kind Fredholm integral equations in indirect measurements, Metrol. Meas. Syst. 16 (3) (2009) 333–357.
[25] Davide Balatti, Hamed Haddad Khodaparast, Michael I. Friswell, Marinos Manolesos, Improving wind tunnel "1-cos" gust profiles, J. Aircr. 59 (6) (2022) 1514–1528, https://doi.org/10.2514/1.C036772.
[26] Davide Balatti, Hamed Haddad Khodaparast, Michael I. Friswell, Marinos Manolesos, Andrea Castrichini, Experimental and numerical investigation of an aircraft wing with hinged wingtip for gust load alleviation, J. Fluids Struct. 119 (2023) 103892.
[27] Aniekan Essien, Cinzia Giannetti, A deep learning model for smart manufacturing using convolutional lstm neural network autoencoders, IEEE Trans. Ind. Inform. 16 (9) (2020) 6069–6078.

[28] John F. Kolen, Stefan C. Kremer, Gradient Flow in Recurrent Nets: The Difficulty of Learning LongTerm Dependencies, 2001.
[29] Koushal Kumar, B. Abhishek, Artificial Neural Networks for Diagnosis of Kidney Stones Disease, GRIN Verlag, 2012.
[30] François Chollet, et al., Keras, https://keras.io, 2015. (Accessed 30 March 2023).
[31] A. Castrichini, V.H. Siddaramaiah, D.E. Calderon, J.E. Cooper, T. Wilson, Y. Lemmens, Preliminary investigation of use of flexible folding wing tips for static and dynamic load alleviation, Aeronaut. J. 121 (1235) (2017) 73–94.
[32] Davide Balatti, Numerical and experimental studies of aeroelastic hinged wingtips, PhD thesis, Swansea University, 2023.
[33] Bendong Zhao, Huanzhang Lu, Shangfeng Chen, Junliang Liu, Dongya Wu, Convolutional neural networks for time series classification, J. Syst. Eng. Electron. 28 (1) (2017) 162–169.
[34] Sima Siami-Namini, Neda Tavakoli, Akbar Siami Namin, The performance of lstm and bilstm in forecasting time series, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 3285–3292.
[35] Kayal Lakshmanan, Eugenio Borghini, Arnold Beckmann, Cameron Pleydell-Pearce, Cinzia Giannetti, Data modelling and remaining useful life estimation of rolls in a steel making cold rolling process, Proc. Comput. Sci. 207 (2022) 1057–1066.
[36] Ioannis E. Livieris, Emmanuel Pintelas, Panagiotis Pintelas, A cnn–lstm model for gold price time-series forecasting, Neural Comput. Appl. 32 (2020) 17351–17360.
[37] Xuebo Jin, Xinghong Yu, Xiaoyi Wang, Yuting Bai, Tingli Su, Jianlei Kong, Prediction for time series with cnn and lstm, in: Proceedings of the 11th International Conference on Modelling, Identification and Control (ICMIC2019), Springer, 2020, pp. 631–641.
[38] Kayalvizhi Lakshmanan, Antonio J. Gil, Ferdinando Auricchio, Fabrizio Tessicini, A Fault Diagnosis Methodology for an External Gear Pump with the Use of Machine Learning Classification Algorithms: Support Vector Machine and Multilayer Perceptron, Loughborough University, 2020.
[39] Kayal Lakshmanan, Fabrizio Tessicini, Antonio J. Gil, Ferdinando Auricchio, A fault prognosis strategy for an external gear pump using machine learning algorithms and synthetic data generation methods, Appl. Math. Model. 123 (2023) 348–372.
[40] Matheus F. Torquato, Kayalvizhi Lakshmanan, Natalia Narożańska, Ryan Potter, Alexander Williams, Fawzi Belblidia, Ashraf A. Fahmy, Johann Sienz, Cascade optimisation of battery electric vehicle powertrains, Proc. Comput. Sci. 192 (2021) 592–601.
[41] Davide Chicco, Matthijs J. Warrens, Giuseppe Jurman, The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation, PeerJ Comput. Sci. 7 (2021) e623.
[42] Sadiq Hussain, Silvia Gaftandzhieva, Md Maniruzzaman, Rositsa Doneva, Zahraa Fadhil Muhsin, Regression analysis of student academic performance using deep learning, Educ. Inf. Technol. 26 (1) (2021) 783–798.
[43] Kayal Lakshmanan, Aurash Karimi, Alex Carr, Philippe Wauters, Michael Auinger, Cameron Pleydell-Pearce, Cinzia Giannetti, A hybrid modelling approach based on deep learning for the prediction of the silicon content in the blast furnace, Proc. Comput. Sci. 225 (2023) 2204–2213.