



# Prediction of wing buffet pressure loads using a convolutional and recurrent neural network framework

R. Zahn<sup>1</sup> · A. Weiner<sup>2</sup> · C. Breitsamter<sup>1</sup>

Received: 23 September 2022 / Revised: 21 December 2022 / Accepted: 13 January 2023 / Published online: 14 March 2023  
© The Author(s) 2023

## Abstract

In the present study, a hybrid deep learning reduced-order model (ROM) is applied for the prediction of wing buffet pressure distributions on a civil aircraft configuration. The hybrid model is compound of a convolutional variational neural network autoencoder (CNN-VAR-AE) and a long short-term memory (LSTM) neural network. The CNN-VAR-AE is used for the reduction of the high-dimensional flow field data, whereas the LSTM is applied to predict the temporal evolution of the pressure distributions. For training the neural network, experimental buffet data obtained by unsteady pressure sensitive paint measurement (iPSP), is applied. As a test case, the Airbus XRF-1 configuration is selected, considering two different experimental setups. The first setup is defined by a wind tunnel model with a clean wing, whereas the second setup includes an ultra high bypass ratio engine nacelle on each wing. Both configurations have been tested in the European Transonic Windtunnel, considering several transonic buffet conditions. Finalizing the training of the hybrid neural networks, the trained models are applied for the prediction of buffet flow conditions which are not included in the training data set. A comparison of the experimental results and the pressure distributions predicted by the hybrid ROMs indicate a precise prediction performance. Considering both aircraft configurations, the main buffet flow features are captured by the hybrid ROMs.

**Keywords** Deep learning · Convolutional autoencoder · Long short-term memory neural network · Wing buffet aerodynamics · Airbus XRF-1 configuration

## List of symbols

$\mathbf{b}$	Bias vector	$m$	Timesteps predicted ahead
$C$	Channel dimension	$Ma_\infty$	Freestream Mach number
$C_{in}$	Input channel	$N_{in}$	Number of input time steps
$C_{out}$	Output channel	$n$	Previous timesteps
$c_p$	Pressure coefficient	$\mathbf{o}$	Output gate vector of LSTM cell
$\mathbf{c}$	Cell state vector of LSTM cell	Re	Reynolds number
$\mathbf{f}$	Forget gate vector of LSTM cell	$s$	Stride parameter
$H$	Height	$W$	Width
$\mathbf{h}$	Hidden state vector of LSTM cell	$\mathbf{W}$	Weight matrix
$\mathbf{i}$	Input gate vector of LSTM cell	$\mathbf{x}$	Model input vector
$k$	Time step	$\mathbf{y}$	Model output vector
		$\alpha$	Angle of attack
		$\sigma$	Sigmoid activation

✉ R. Zahn  
rebecca.zahn@tum.de

A. Weiner  
a.weiner@tu-braunschweig.de

C. Breitsamter  
christian.breitsamter@tum.de

<sup>1</sup> Technical University of Munich, Munich, Germany

<sup>2</sup> Technical University of Braunschweig, Brunswick, Germany

## 1 Introduction and motivation

Transonic wing buffet, also referred to as wing shock buffet, represents an unsteady aerodynamic phenomenon which occurs during certain flow conditions in the transonic regime. The self-sustained instability of the buffet phenomenon is characterized by shock-wave oscillations

and intermittent boundary layer separation, which leads to undesired vibrations of the aircraft structure. Therefore, the determination of buffet loads is of paramount importance for the safety and efficiency of a civil aircraft.

Nowadays, a variety of computational methods for the prediction of unsteady aerodynamic loads is available. For the investigation of transonic buffet flows, which are dominated by non-linear flow features, high-fidelity Computational Fluid Dynamics (CFD) methods such as Unsteady Reynolds-Averaged Navier–Stokes (URANS) or Detached-Eddy Simulation (DES) are commonly applied [1, 2]. Further, experimental investigations have been used to some extent for the determination of unsteady buffet loads [3, 4]. However, although high-performance computing and experimental facilities are available, the accurate computation of unsteady buffet loads is challenging due to the high computational costs and time, especially if realistic aircraft configurations are considered.

An alternative to the application of high-fidelity solutions is given by system identification methods. These methods are characterized by the representation of an unknown aerodynamic system, defined by a certain number of inputs and outputs. In unsteady aerodynamic modeling, the inputs are usually defined by the motion of the body, whereas the output denotes the corresponding, integrated and local aerodynamic loads [5]. The unknown aerodynamic system can be treated as a black-box with a reduced number of degrees of freedom compared to the full-order method. Therefore, these methods are commonly referred to as Reduced-Order Models (ROMs).

With focus on unsteady aerodynamic modeling, various ROM techniques such as Wiener models [6], Volterra series [7] and Kriging models [8] have been successfully applied. Further, recurrent neural networks (RNN) have been used by Mannarino and Mantegazza [9] for the prediction of limit cycle oscillations. Further, radial basis function neural networks (RBF-NN) have been employed by Zhang et al. [10] and Winter and Breitsamter [11] for flutter analysis and airfoil load prediction, respectively. In addition, Winter and Breitsamter [12] proposed a non-linear ROM based on a series connection of a recurrent neuro-fuzzy model (NFM) and a multilayer perceptron (MLP) neural network for non-linear time series prediction.

However, considering the non-linear ROM methods mentioned above, the prediction of spatio-temporal characteristics, such as unsteady pressure distributions, is not feasible due to the large number of output variables [13]. To solve this issue, several studies dealt with the application of dimensionality reduction techniques, such as Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD). Park et al. [14] applied POD in combination

with a neural network for wing design optimization tasks. Considering buffet aerodynamics, Timme [15] and Ohmichi et al. [16] used POD and DMD in order to identify instability mechanisms and dominant flow modes of three-dimensional wing buffet. Further, a study by Candon et al. [17] investigated DMD for dynamic bending and torsion load spectra prediction resulting from transonic buffet on a high-agility aircraft configuration.

Besides the application of dimensionality reduction techniques, deep learning methods such as convolutional neural networks (CNN) have been applied for the prediction of spatio-temporal characteristics of high-dimensional flow field data. Afshar et al. [18] applied a CNN for the prediction of the velocity and pressure field around an airfoil, depending on the shape of the airfoil and flow parameters. Sekar et al. [19] used a CNN for feature extraction of an airfoil and further processed them as an input for a MLP neural network to predict steady flow fields. To enable a better and faster prediction of temporal characteristics of flow field data, several studies proposed hybrid deep learning models based on CNN and RNN. Li et al. [20] applied a convolutional long short-term memory (ConvLSTM) neural network for the prediction of supersonic cascaded channel flow. Hasegawa et al. [21] proposed a series connected ROM based on a convolutional autoencoder (CNN-AE) and a LSTM for the prediction of unsteady-flow characteristics around bluff bodies with different shapes. Further, Nakamura et al. [22] applied the proposed CNN-AE/LSTM model for the prediction of turbulent channel flow.

Following the studies mentioned above, the present study focuses on the application of the hybrid CNN-AE/LSTM neural network as proposed by [21] for the prediction of wing buffet pressure loads. In addition, instead of using a standard CNN-AE, a convolutional variational autoencoder (CNN-VAR-AE) is applied. Compared to previous studies, which used numerical data sets, experimental buffet data are applied for the training of the hybrid neural network. In particular, surface pressure data originated from unsteady pressure sensitive paint (iPSP) measurements, which are conducted by the German Aerospace Center (DLR), are used. The experimental data are obtained during a wind tunnel test campaign in the European Transonic Wind tunnel (ETW), as part of the research project FOR2895. As a test case, the Airbus XRF-1 configuration is selected. During the test campaigns, a clean wing as well as a configuration including ultra high bypass ratio (UHBR) nacelles, are considered. Each configuration has been tested at different buffet conditions. Based on the available experimental data, two independent deep learning models are trained. For performance evaluation, the trained ROMs are applied for the prediction of flow conditions which are not included in the

data set used for training. A comparison of the experimental data and the results predicted by the hybrid ROMs, a precise agreement is indicated. For both configurations the trained ROMs are able to capture the main buffet characteristics.

## 2 Deep learning approaches

In the following section, the applied hybrid deep learning model is introduced. The first subsection covers a brief introduction of convolutional neural networks (CNN) as well as the working principle of autoencoders (AE). In the second part of the section, the LSTM neural network applied for time-series prediction, is discussed in detail.

### 2.1 Convolutional neural network (CNN)

The convolutional neural network (CNN) [23] is a type of neural network mainly developed for processing data represented by a grid-like topology [24]. Although CNNs are applicable to time-series data, they are commonly used for image data, containing spatial information. In general, a CNN is composed of three main layers: a convolutional layer, a pooling layer and a fully connected (FC) layer. The convolutional layer represents the main building block of each CNN architecture. Assuming a three-dimensional matrix of data, the position of each data point can be defined by an index  $(i, j, o)$ . Following the nomenclature of Goodfellow et al. [24] and Rosov and Breitsamter [13], the convolution can be defined as follows:

$$y_{i,j,o} = \sum_{l=0}^{C_{in}-1} \sum_{m=0}^{H_k-1} \sum_{n=0}^{W_k-1} x_{l,i+xs+m,j+xs+n} W_{o,l,m,n} + b_o \tag{1}$$

with  $x$  denoting the input at each index and  $W$  defining the weights of the filter, also referred to as kernel. The size of the input data is defined by its height  $H$  and width  $W$  as well as the number of input channels  $C_{in}$  ( $C_{in} \times H \times W$ ). The corresponding kernel size is defined by  $C_{in} \times H_k \times W_k \times C_{out}$  and includes an additional dimension, referred to as output channel  $C_{out}$  [13]. The kernel slides stepwise over the input data, performing element-wise multiplications with the elements of the input matrix. At each step, the results are summed up and written as a new output.

Further, a parameter referred to as stride  $s$  is introduced in Eq. 1, which defines the number of elements the filter selects at each step. By applying the striding technique, the output size can be reduced compared to the input size. Choosing a stride of one, slides are picked an entry apart, resulting in an output size equal to the input size. If the stride is defined as two, the size of the input data is reduced by a factor of two and so on.

Within the present work, the CNN architecture is used within an autoencoder structure, which is commonly referred to as a convolutional autoencoder (CNN-AE). The application of a CNN-AE aims for encoding a given input and reconstructing the input to a given output [24]. The network structure is composed of three parts, the encoder, decoder and latent space. The encoder maps the high-dimensional data into a low-dimensional latent space and the decoder extends the data back into its original resolution. To learn the representation of the input data, the network is trained in an unsupervised way.

To enable a better training process, a convolutional variational AE (CNN-VAR-AE) is applied. Compared to a normal CNN-AE, the input is encoded as a normal distribution over the latent space, rather than a single point. From this distribution, a random point is selected, which is fed into the decoder. Based on the sampled point, the reconstruction error is computed and backpropagated through the CNN-VAR-AE.

### 2.2 Long short-term memory (LSTM)

The LSTM architecture has been proposed by Hochreiter and Schmidhuber [25] and represents a type of recurrent neural network (RNN), which is capable of predicting both long- and short-term dependencies in time-series data. In contrast to a basic RNN, the application of the LSTM solves for a problem known as vanishing gradient problem [26], where the training performance of the network saturates after a certain number of training iterations. Further, the representation of time-delayed effects, which are important for unsteady aerodynamic modeling, are incorporated in the LSTM architecture [5]. The LSTM cell processes the incoming information through three gates, namely the forget gate  $f$ , the input gate  $i$  and the output gate  $o$ . In Fig. 1, the architecture of a LSTM cell including the characteristic gates, is illustrated.

The forget gate  $f$  processes the input of the current time step  $x_t$  as well as the vector representing the output from the previous time step  $h_{t-1}$ , which is also referred to as the hidden state of the LSTM cell:

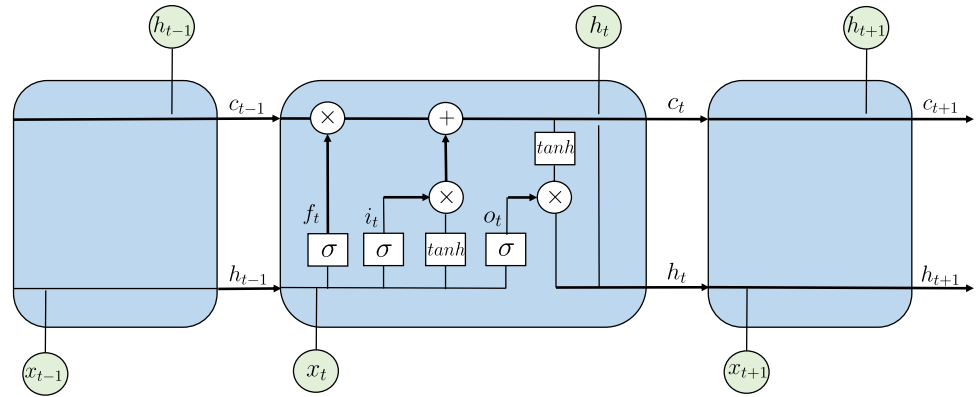
$$f_t = \sigma(W_f x_t + W_f h_{t-1} + b_f). \tag{2}$$

Both inputs of the forget gate are multiplied with a set of weights  $W_f$  and a bias  $b_f$  is added. By applying a sigmoid ( $\sigma$ ) activation, parts of the incoming information is discarded from the cell.

Equal to the forget gate, the input gate processes the current time step  $x_t$  as well as the hidden state from the previous time step  $h_{t-1}$  by means of a sigmoid activation function:

$$i_t = \sigma(W_i x_t + W_i h_{t-1} + b_i) \tag{3}$$

**Fig. 1** Architecture of the LSTM cell including the characteristic gate structure



with  $\mathbf{W}_i$  and  $\mathbf{b}_i$  defining the weights and bias of the input gate, respectively. In the second step, both inputs are processed by an hyperbolic tangent ( $\tanh$ ) activation, which creates a new cell state vector  $\tilde{c}_t$ :

$$\tilde{c}_t = \tanh(\mathbf{W}_h \mathbf{x}_t + \mathbf{W}_h \mathbf{h}_{t-1} + \mathbf{b}_h). \quad (4)$$

Based on the new cell state  $\tilde{c}_t$ , the old information in the cell is updated. Therefore, the cell state from the previous time step  $c_{t-1}$  is multiplied with the forget gate vector  $f_t$  and the current cell state is updated with the input gate vector  $i_t$ :

$$c_t = f_t c_{t-1} + i_t \tilde{c}_t. \quad (5)$$

After passing the input gate, the data of the current input  $x_t$ , the previous hidden state  $h_{t-1}$  as well as the current cell state  $c_t$  are processed by a sigmoid and tanh activation:

$$\begin{aligned} o_t &= \sigma(\mathbf{W}_o \mathbf{x}_t + \mathbf{W}_o \mathbf{h}_{t-1} + \mathbf{b}_o) \\ \mathbf{h}_t &= o_t \cdot \tanh(c_t). \end{aligned} \quad (6)$$

The prediction, which are defined by the updated hidden state  $h_t$ , are transferred to the next hidden layer or the output layer.

### 2.3 Hybrid model architecture

To enable an accurate prediction of wing buffet pressure loads, the CNN-VAR-AE and the LSTM as introduced in Sects. 2.1 and 2.2 are combined to form a hybrid deep learning model. In Fig. 2, the architecture of the hybrid model is visualized.

The hybrid ROM is applied for the prediction of buffet pressure distributions at time step  $k + m$  based on several previous  $c_p$ -snapshots of the pressure distribution at time steps  $k - n + 1$  to  $k$ . Here,  $n$  denotes the number of pressure samples applied for the prediction, whereas  $m$  defines the number of time steps ahead which should be predicted by the trained model.

As already outlined in Sect. 2.1, the CNN-VAR-AE is composed of an encoder, decoder and latent space vector. Both the encoder and decoder of the CNN-VAR-AE are divided into several levels. At each level, a number of operations is performed on the input data, which is fed into the encoder. Defining the input of the CNN-VAR-AE, snapshots of pressure distribution  $c_p(\tau, x, y)$  at each measured time step  $\tau$  are combined into an array along the channel dimension  $C_{in}$ . Therefore, the size of the input is defined as  $N_{in} \times C_{in} \times H \times W$ , with  $N_{in}$  defining the number of input time steps.  $H$  and  $W$  denote the height and width of each  $c_p$ -snapshot, which are defined by powers of two.

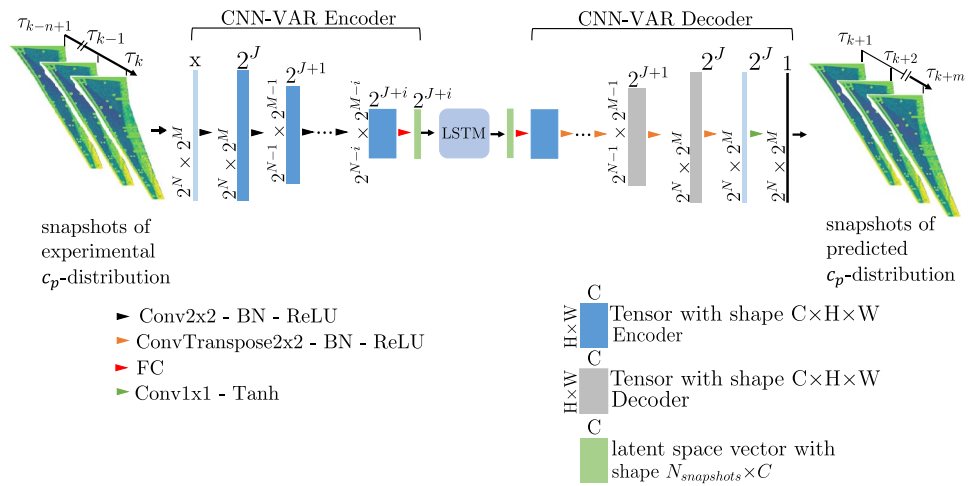
As a first step, a convolution with a kernel size of  $2 \times 2$  and stride  $s = 2$  is performed. Following the convolution, the input is normalized based on the mean and variance of each training batch, which is referred to as batch normalization (BN) [27]. In the last step, an activation function is applied to each element of the incoming array. Within the scope of this work, a rectified linear unit (ReLU) is used.

The sequence of operations described above is applied at each level of the encoder. Therefore, the output has twice as many channels as the input. However, the first convolution applied to the input array increases the number of channels from  $x$  to  $C$ . In addition, the size of the spatial dimensions ( $H, W$ ) is reduced by a factor of two at each level of the encoder. As the last encoder processing step, the data are fed into a FC layer to reduce the channel dimension. In addition, a tanh activation is applied. The resulting reduced latent features are then sampled from a normal distribution.

After the input data have been processed by the encoder, the data are reshaped and fed into the LSTM neural network. The LSTM is used for the prediction of buffet pressure distributions at time step  $k + m$  based on several previous snapshots of the pressure distribution at time steps  $k - n + 1$  to  $k$ . In Fig. 3, the schematic of the multiple time steps ahead prediction obtained by the LSTM model, is depicted.

After finalizing the prediction of the LSTM, the output is fed into the decoder. Therefore, the low-dimensional data are reconstructed to the original high-dimensional flow field.

**Fig. 2** Architecture of the hybrid deep learning model



At each level of the decoder, a transposed convolution is applied to upscale the spatial dimension. The corresponding kernel size and stride values are chosen equally to those of the encoder. In addition, BN is applied at each level of the decoder. To obtain the final prediction, an activation function is applied to the output of the last decoder level. Therefore, a hyperbolic tangent tanh is chosen, which reshapes all predicted elements to  $[-1, 1]$ .

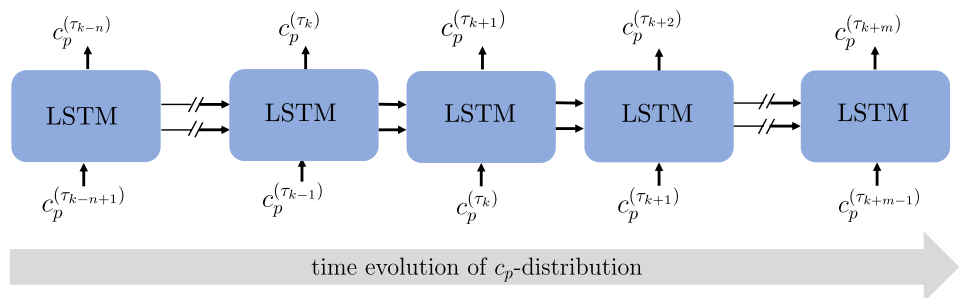
### 3 Test case: Airbus XRF-1

The experimental buffet data used for training and validation of the hybrid ROM have been obtained during two consecutive wind tunnel test campaigns in the European Transonic Wind Tunnel (ETW). To provide a modern and realistic test case, the Airbus XRF-1 research configuration has been selected. During the test campaigns, two different configurations were considered for buffet investigation. Both models include a fixed vertical (VTP) and horizontal tailplane (HTP) as well as adjustable ailerons. In addition to the stabilizers

and the ailerons, one model is additionally equipped with Ultra High Bypass Ratio (UHBR) engine nacelles. The nacelles represent through flow nacelles (TFNs), which are connected to the wing with engine pylons. In Fig. 4a, b, both XRF-1 wind tunnel models with clean wing and UHBR engine nacelles, are visualized, respectively.

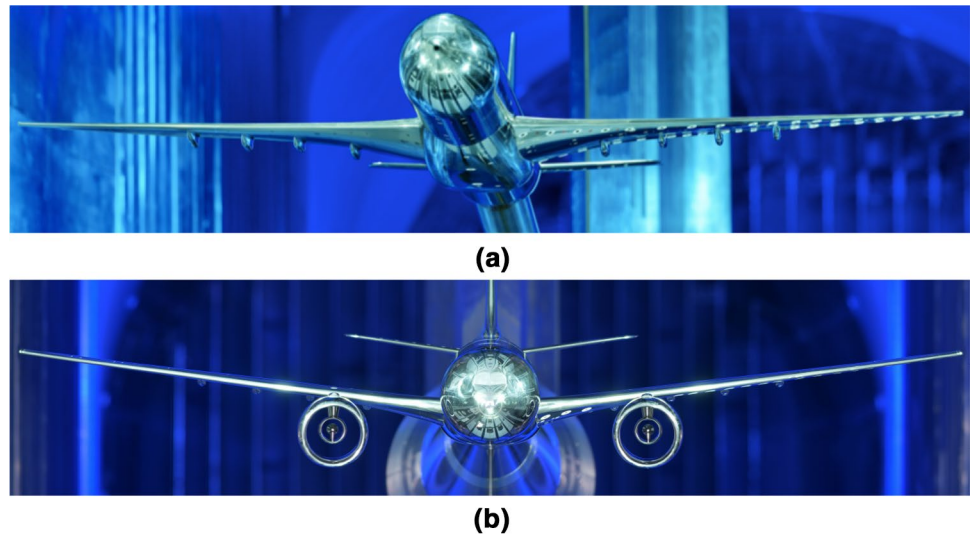
The experimental investigations, which have been conducted by the ETW and the German Aerospace Center (DLR), included unsteady pressure sensitive paint [28, 29] measurements, which are referred to as iPSP in the following. With the iPSP setup, upper wing surface buffet data acquisition was possible with a sampling frequency of 1000 Hz. All iPSP measurements were conducted in a pitch/pause mode, enabling a fixed incidence angle at each run. The measured flow conditions were defined to enable the analysis of isolated effects due to changes in Mach and Reynolds number as well as the angle of attack. The Mach number was varied between  $Ma_\infty = [0.84, 0.87, 0.9]$ , whereas the Reynolds number was defined by  $Re = [3.3 \text{ Mio.}, 12.9 \text{ Mio.}, 25 \text{ Mio.}]$ . For each flow condition, several angles of attack were considered.

**Fig. 3** Prediction of time evolution of the  $c_p$ -distribution as obtained by the LSTM





**Fig. 4** **a** Front view of the Airbus XRF-1 wind tunnel model with clean wings. **b** Front view of the Airbus XRF-1 wind tunnel model with UHBR engine nacelles (©Airbus/ETW)



#### 4 Flow topology at buffet condition

In the following, a characterization of the buffet flow topology on the Airbus XRF-1 wing based on the measured iPSP data, is given. In particular, distinct differences between the buffet flow on the clean wing and the wing with UHBR are identified. Due to visualization restrictions, axis notations are suppressed in the following figures. Further, the buffet pressure distributions are visualized using the mean of the pressure coefficient ( $\bar{c}_p$ ).

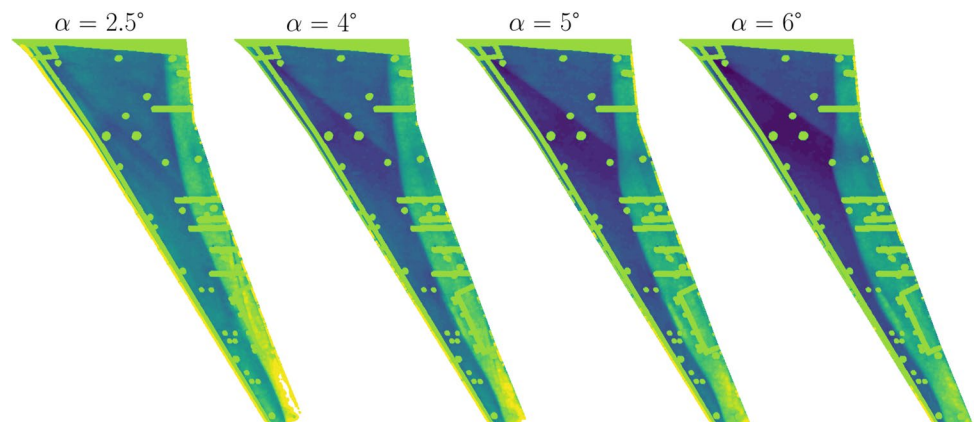
Starting with the clean wing configuration, the pressure distribution of the upper wing surface at a flow condition of  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio and varying angles of attack  $\alpha = [2.5^\circ, 4^\circ, 5^\circ, 6^\circ]$  is depicted in Fig. 5.

At each flow condition, a distinct  $\lambda$ -shaped two-shock pattern across the entire wing span is clearly visible. The main shock spreads over the rear part of the chord from wing root

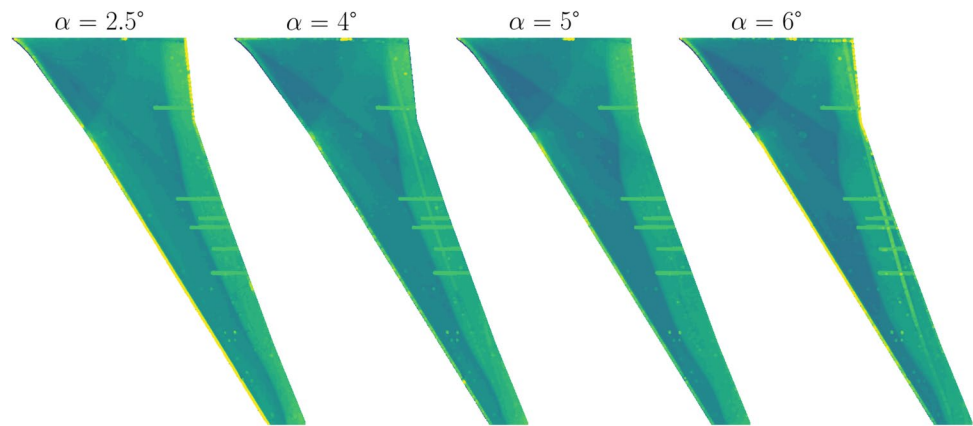
to wing tip, whereas a weaker shock is visible at the inboard part of the wing. This shock originates near the leading edge (LE) at the root of the wing towards the mid section of the wing. With increasing incidence, the inboard shock position slightly shifts aft, while the mid-span and outboard shock moves further upstream. The sweep angle of the inboard shock increases and the intersection of both shock patterns moves inboard. Further, the intensity of the shock increases with increasing incidence.

Shifting the focus to the surface pressure distribution of the wing with the UHBR nacelle installed, some differences in the surface pressure are indicated. In Fig. 6, pressure distributions at  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio and varying angles of attack  $\alpha = [2.5^\circ, 4^\circ, 5^\circ, 6^\circ]$  are visualized. Similar to the pressure distribution on the clean wing configuration, a spanwise two-shock pattern is visible. Further, with increasing incidence, the shock in the mid-span region and close to the tip shifts towards the LE. Compared to the

**Fig. 5** Comparison of surface  $c_p$  at varying angles of attack (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.)



**Fig. 6** Comparison of surface  $c_p$  at varying angles of attack (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.)



shock pattern on the clean wing, a more complex shock variation across the span, including lateral separation, is observed. In addition, the movement of the inboard shock position is less pronounced than on the clean wing; however, at around 30% span, the shock position starts to move to the TE. In addition to the two-shock pattern, a third inboard positioned shock is indicated, originating from the intersection of the LE with the engine pylon inboard towards the wing root. Further, an intersection between both inboard shocks is indicated.

As shown in the previous figures, characteristic buffet flow features are captured by the optical iPSP measurements. Further, clear differences between both wind tunnel models are indicated. Therefore, the hybrid ROM is applied for the prediction of a highly non-dimensional flow field, representing characteristic buffet features.

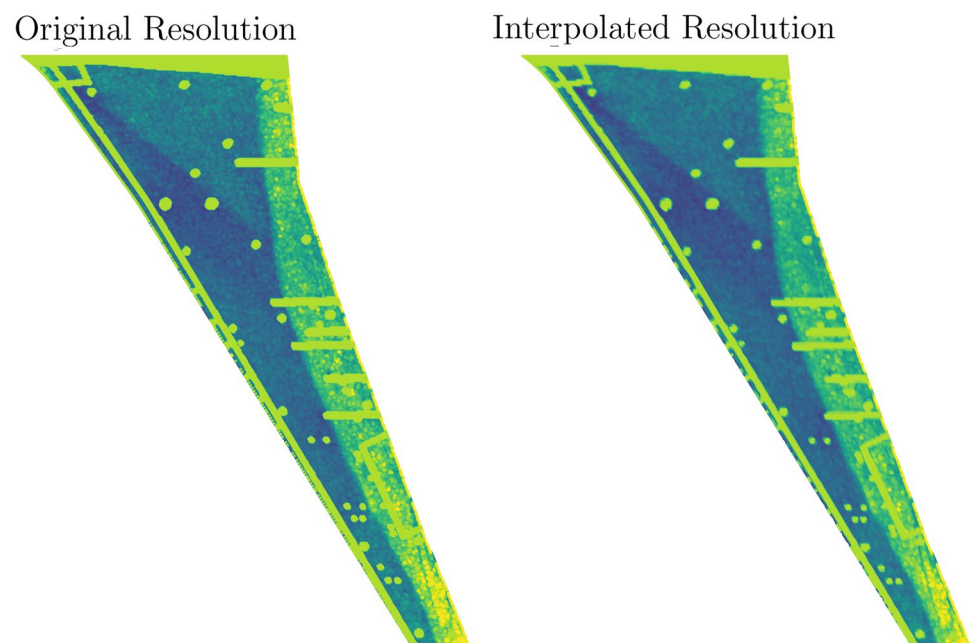
## 5 Application of the hybrid deep learning model

In the following section, the application process and the results of the hybrid ROM are discussed. In the first part of the section, the preprocessing steps of the experimental data are briefly described. In the second subsection, the selection of hyperparameters and the training procedure is outlined. In the last subsection, the result of the trained ROMs are presented and discussed in detail.

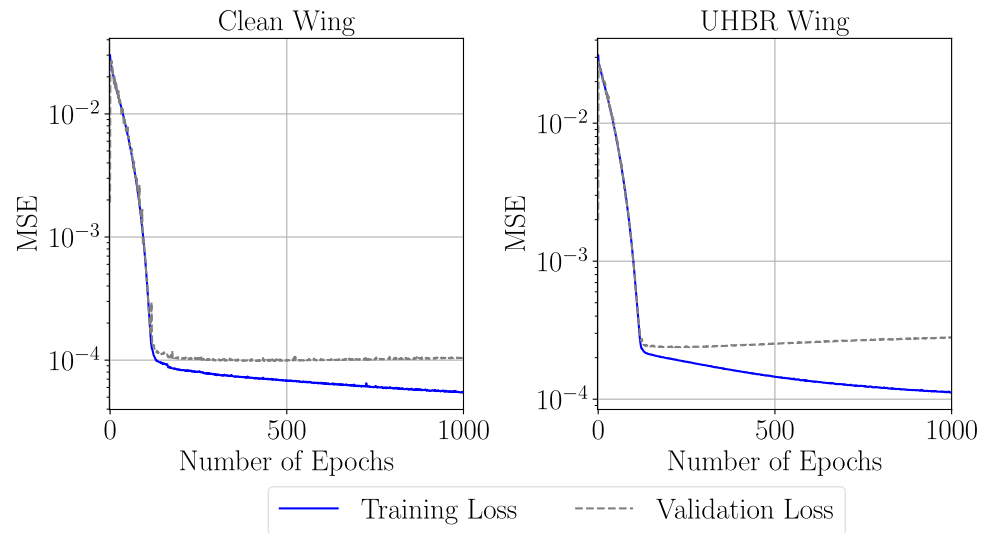
### 5.1 Data preprocessing

To feed the experimental data as obtained by iPSP measurements into the deep learning model, the data need to be preprocessed accordingly. All preprocessing steps are

**Fig. 7** Original (left) and interpolated (right) number of data points representing the buffet pressure distribution on the upper wing surface of the XRF-1 configuration (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 4^\circ$ )



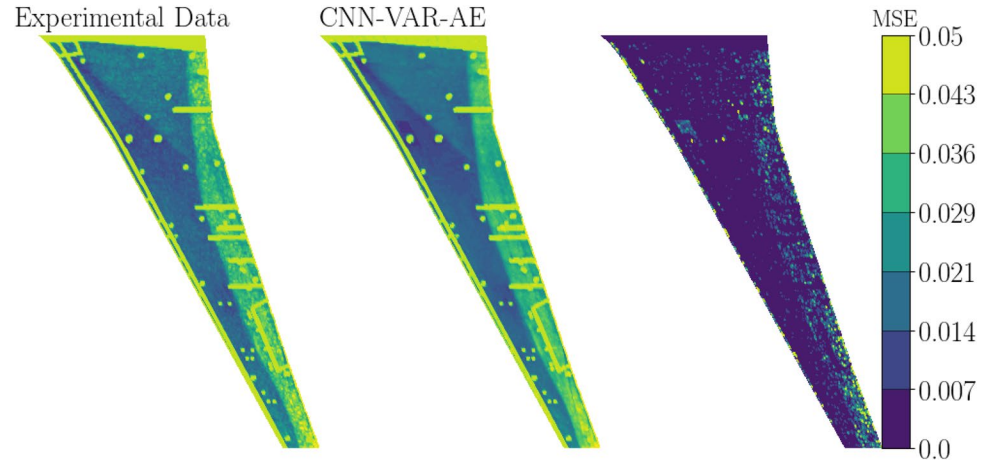
**Fig. 8** Convergence trends of training and validation losses of the individually trained CNN-VAR-AEs (left: clean wing, right: UHBR wing)



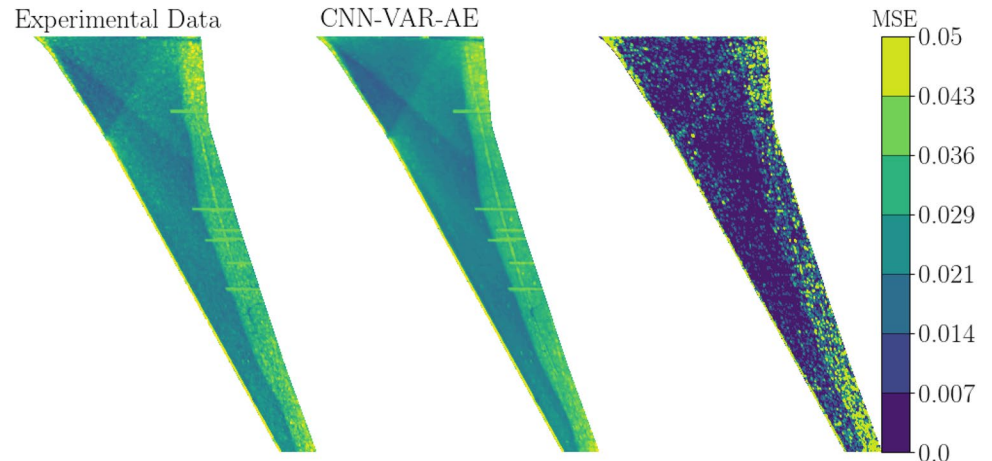
accomplished using the Python library flowTorch [30]. Based on the geometry of the XRF-1 wind tunnel model

and the iPSP measurement technique, the pressure

**Fig. 9** Comparison of an original validation  $c_p$ -snapshot and a  $c_p$ -snapshot obtained by the trained CNN-VAR-AE (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 4^\circ$ )

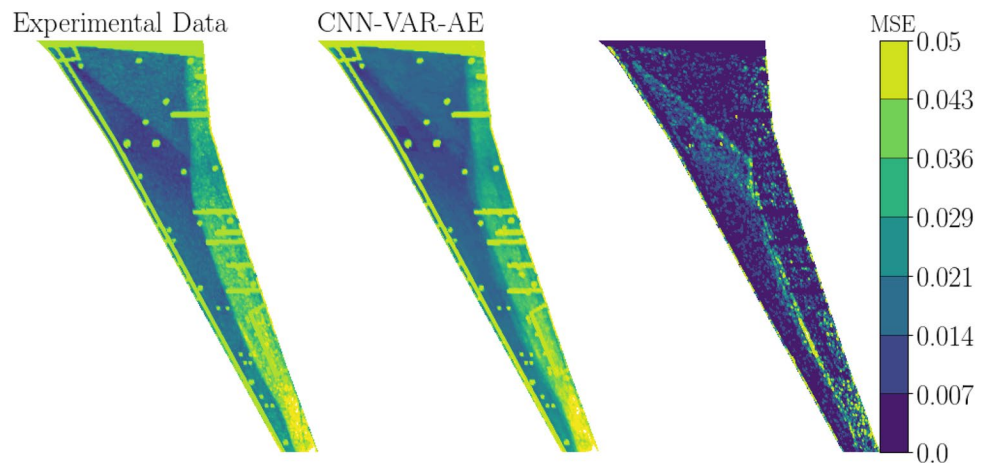


**Fig. 10** Comparison of an original validation  $c_p$ -snapshot and a  $c_p$ -snapshot obtained by the trained CNN-VAR-AE (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 4^\circ$ )





**Fig. 11** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained CNN-VAR-AE (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )



distribution on the upper wing surface is discretized by  $465 \times 159$  data points.

Due to measurement errors during the wind tunnel measurement campaign, the data set includes a small amount of non physical  $c_p$  values. Therefore, in the first preprocessing step, the experimental data are cleaned by applying a weight mask, which defines values of  $c_p \geq 1.5$  as 1 and values of  $c_p \leq -1.5$  as 0.

In the second step, the number of data points representing the pressure distribution is downscaled by linear interpolation from  $465 \times 159$  to  $256 \times 128$  ( $2^8 \times 2^7$ ). In Fig. 7, a comparison of the wing pressure distribution represented by the original amount of data points (left) and the interpolated data points (right), is visualized. As shown in Fig. 7, the reduced resolution still maintains a high level of detail of the spatial resolution. Further, rescaling the surface resolution to an input array size of powers of two enables faster processing of the neural network.

In the final step, the data set is normalized based on the minimum and maximum pressure values in the data set ( $c_{p,min}, c_{p,max}$ ). Therefore, the resulting value range is rescaled to  $[-1, 1]$ .

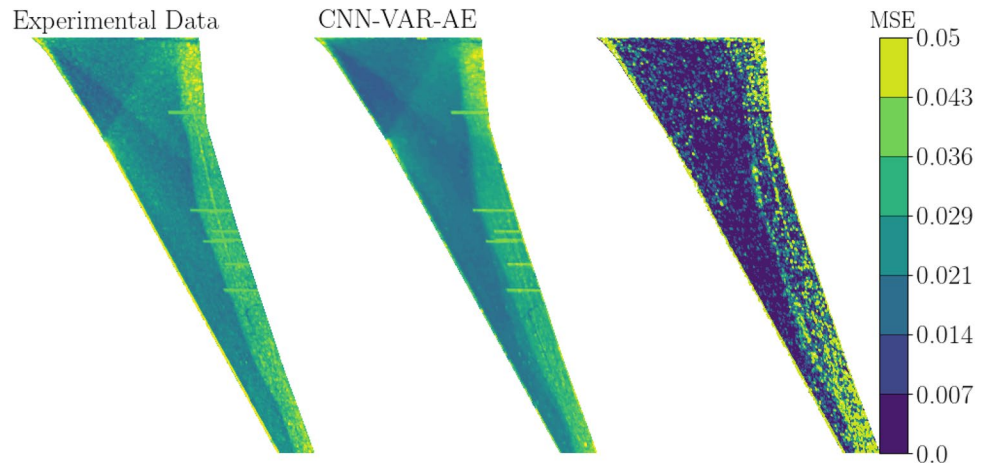
### 5.2 Training of the hybrid model

The training procedure of the hybrid deep learning model includes two consecutive steps. In the first step, the CNN-VAR-AE is trained independently, using  $c_p$ -snapshots representing on or more flow conditions. In the second step, the LSTM is trained using the reduced  $c_p$ -snapshots as obtained by the trained encoder of the CNN-VAR-AE.

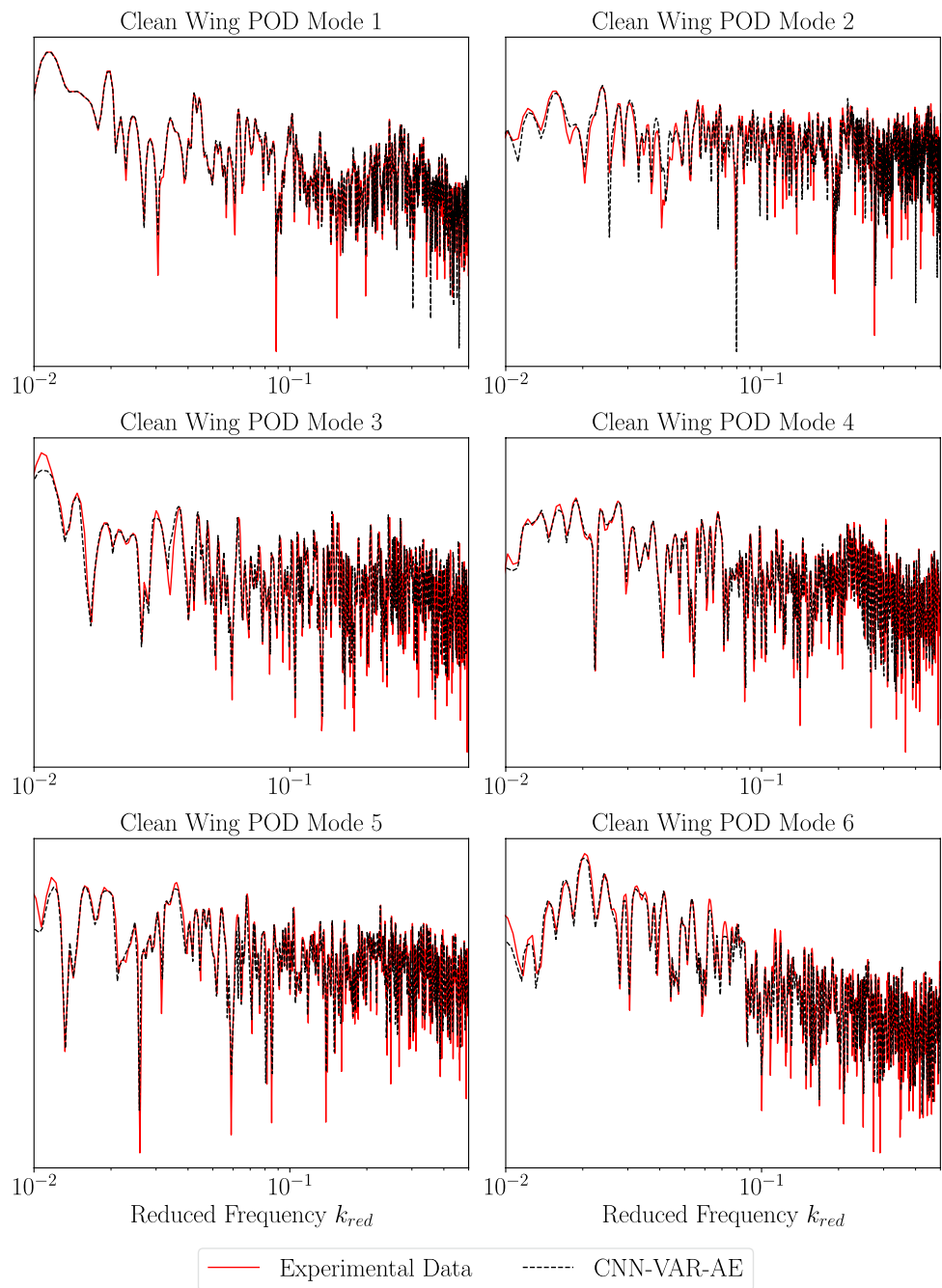
Since the application of the trained ROM aims for the prediction of surface pressure distributions at unknown buffet conditions,  $c_p$ -snapshots representing several buffet conditions are applied for ROM training. Therefore, flow conditions must be selected in order to allow an interpolation for the flow condition of interest. In the following, the focus will be on a prediction of buffet flow considering a variation of the angle of attack. Further, two hybrid models are individually trained using the data originated from the clean wing and the UHBR wing configuration, respectively.

As a first step, the CNN-VAR-AE is trained using snapshots representing two different flow conditions, defined by  $Ma_\infty = 0.9, Re = 25$  Mio. and  $\alpha = [4^\circ, 6^\circ]$ . The CNN-VAR-AE is trained using 2000  $c_p$ -snapshots, with 1000 snapshots

**Fig. 12** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained CNN-VAR-AE (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )



**Fig. 13** Power Spectra of the first six POD modes of the buffet cycle (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ ). The experimental results are compared to the results predicted by the CNN-VAR-AE

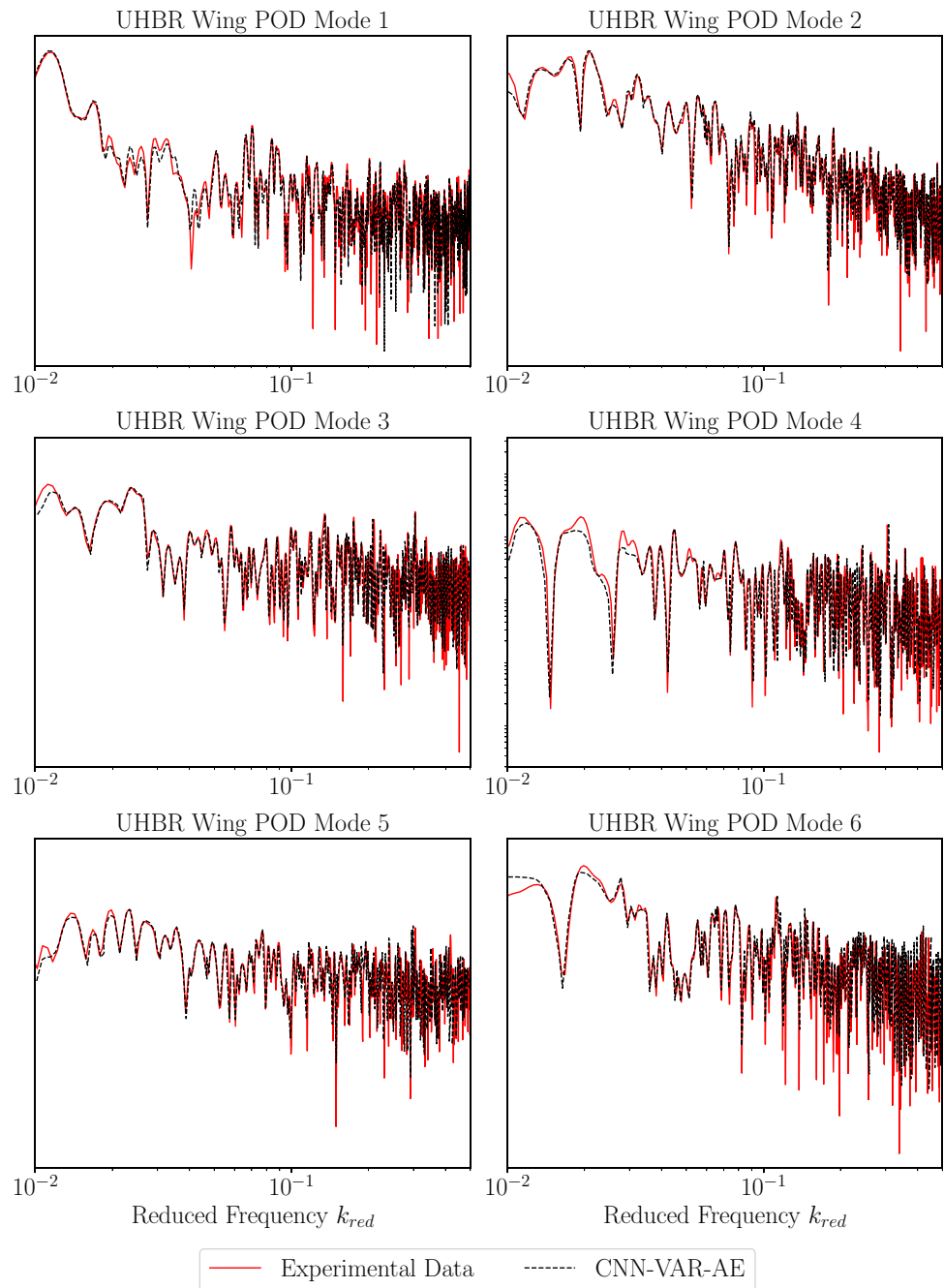


representing each flow condition. For hyperparameter tuning and validation, in total 400 snapshots are used. During the training, the snapshots are fed into the encoder in batches, including 128 time steps each. For both the encoder and decoder five convolution levels are applied, which reduces the input array size of  $64 \times 256 \times 128$  to  $512 \times 32 \times 16$ . By passing the FC layer, the latent channel size is downscaled from 512 to 256 features. The initial learning rate is set to  $10^{-4}$  and the training of the CNN-VAR-AE is terminated after 1000 epochs. To achieve an appropriate model performance, the mean squared error (MSE) between the reference

experimental data and the predictions, is minimized. In Fig. 8, training and validation losses of the individually trained CNN-VAR-AEs, are visualized.

Besides the evaluation of the corresponding convergence trends, the training performance of each CNN-VAR-AE is assessed based on a visual comparison of an experimental  $c_p$ -snapshot of the validation data set and the corresponding  $c_p$ -snapshot as obtained by the CNN-VAR-AE. Therefore, one original  $c_p$ -snapshot (left) and the corresponding  $c_p$ -snapshot predicted by the CNN-VAR-AE (middle) are exemplary visualized in Figs. 9 and 10 for the

**Fig. 14** Power Spectra of the first six POD modes of the buffet cycle (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ ). The experimental results are compared to the results predicted by the CNN-VAR-AE

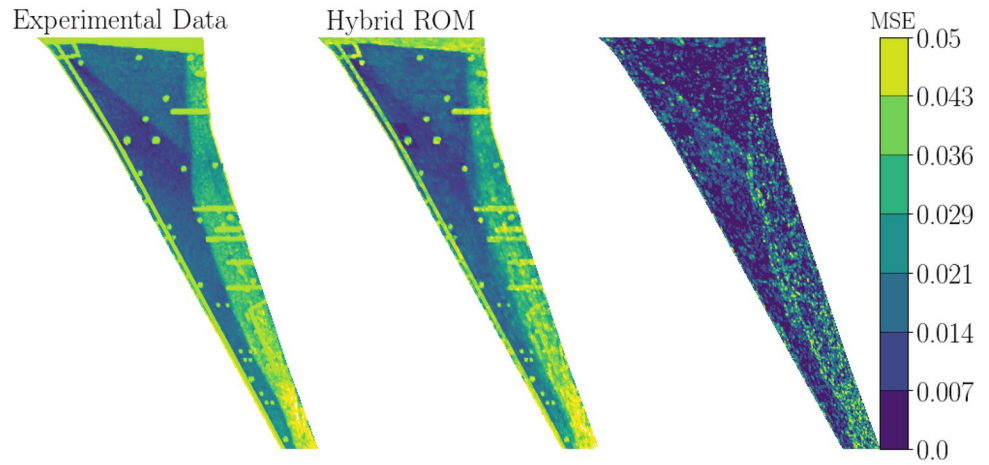


clean wing and UHBR wing configuration, respectively. The deviation between both snapshots is represented by the MSE on the wing surface, as depicted on the wing surface on the right. It has to be noted that the legend only refers to the MSE distribution on the right wing surface. For both test cases, an angle of attack of  $\alpha = 4^\circ$  is considered.

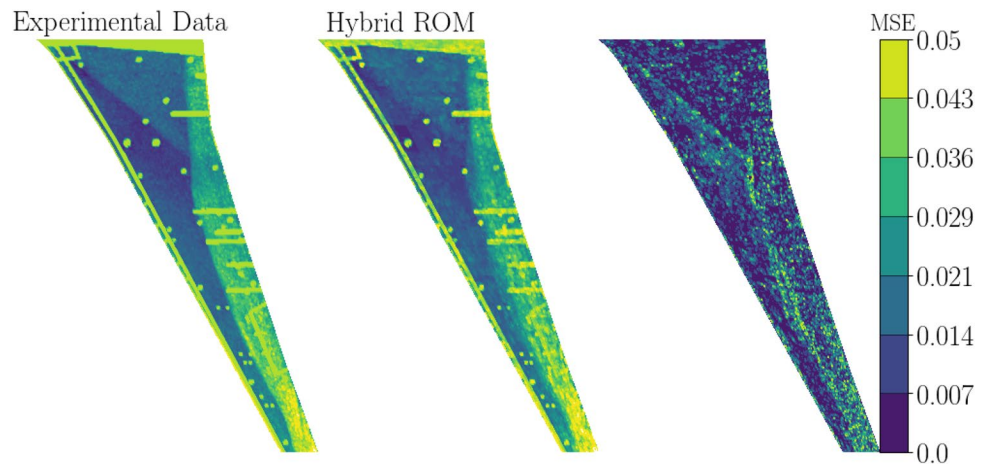
Equal to the training of the CNN-VAR-AE, 2000 snapshots are used for training the LSTM. The overall amount

of training snapshots is divided in sequences of 128 snapshots, while the batch size is defined as one. For the training, a stacked LSTM with two layers is applied, with each layer including 256 neurons. Analog to the training of the CNN-VAR-AE, the initial learning rate is defined as  $10^{-4}$ . The LSTM is trained for 5000 epochs, until a sufficient convergence is reached.

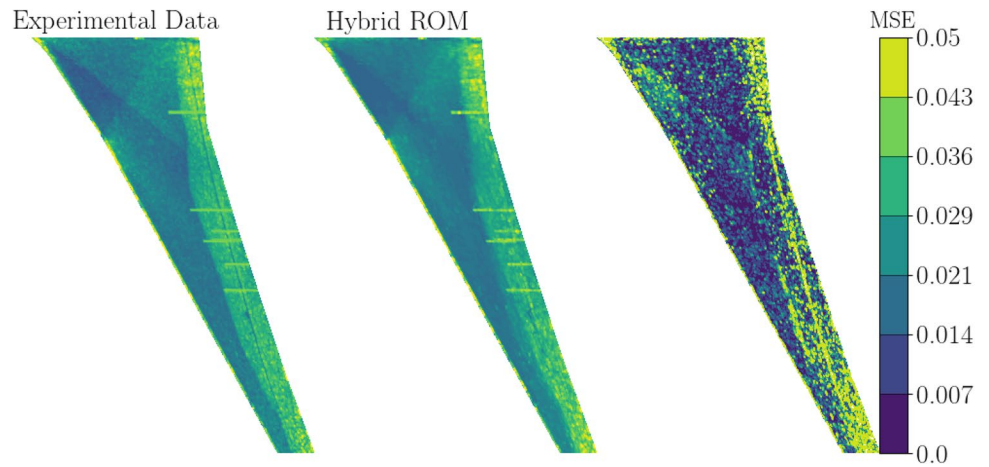
**Fig. 15** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained hybrid ROM at timestep  $t = 150$  (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )



**Fig. 16** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained hybrid ROM at timestep  $t = 200$  (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )

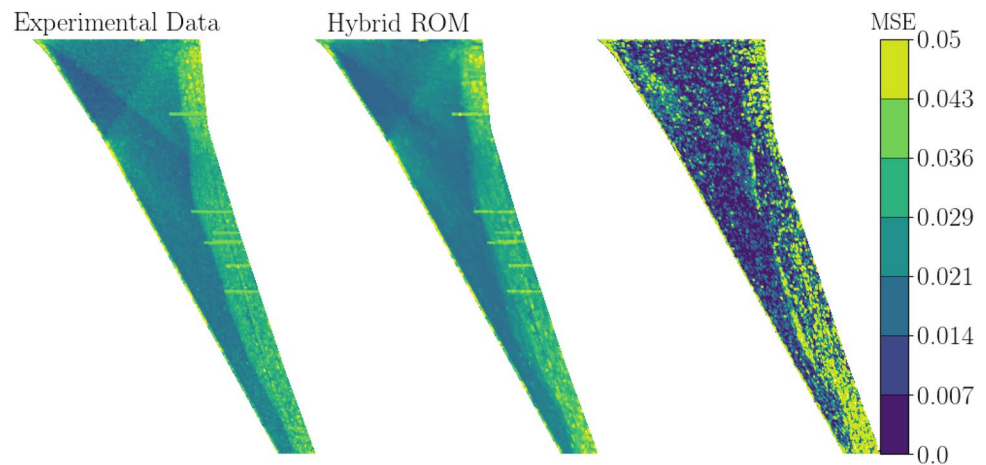


**Fig. 17** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained hybrid ROM at timestep  $t = 150$  (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )





**Fig. 18** Comparison of an original  $c_p$ -snapshot and a  $c_p$ -snapshot predicted by the trained hybrid ROM at timestep  $t = 200$  (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ )



### 5.3 Performance evaluation

To evaluate the performance of the individually trained ROMs, both ROMs are applied for the prediction of buffet pressure distributions at flow conditions which are not included in the training data set. Since for both test cases, angles of attack of  $\alpha = [4^\circ, 6^\circ]$  are used, the test flow condition is defined by  $\alpha = 5^\circ$  (with  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.).

As a first step, the trained CNN-VAR-AEs are applied for the prediction of the unknown buffet condition. In Figs. 11 and 12, a comparison between an original  $c_p$ -snapshot and interpolated  $c_p$ -snapshot of the clean wing and UHBR wing configuration, is visualized, respectively.

Examining the  $c_p$ -distribution as predicted by the trained CNN-VAR-AEs, an overall good agreement is indicated. The characteristic shock patterns as discussed in Sect. 4 are correctly captured by the CNN-VAR-AE for both configurations. Although the chord- and spanwise position of the shock is in good agreement with the experimental data, larger MSE values yield slight deviations along the position of the shock, in particular in spanwise direction, for both test cases. However, it has to be emphasized that the application of the ROM focuses on an accurate representation of the characteristic buffet flow physics, such as the shock movement in chord- and spanwise direction, rather than a correct visual representation of the shock position.

To identify if the trained CNN-VAR-AEs are able to correctly reproduce the buffet flow physics, both experimental and predicted  $c_p$ -snapshots are compared by applying modal analysis, in particular proper orthogonal decomposition (POD). By applying POD, important modes of the buffet flow are extracted, with the order of the modes determining the contribution to the buffet instability.

In Figs. 13 and 14, power spectra of the first six POD modes of the experimental and corresponding predicted data of the clean wing and UHBR wing, are visualized, respectively. Higher modes are neglected, since they do

not indicate a good agreement. At first sight, the coverage of the first six modes does not seem to be very high; however, it has to be considered that the ROM is applied to experimental data, including a high amount of noise and representing non-linear, transonic flow data.

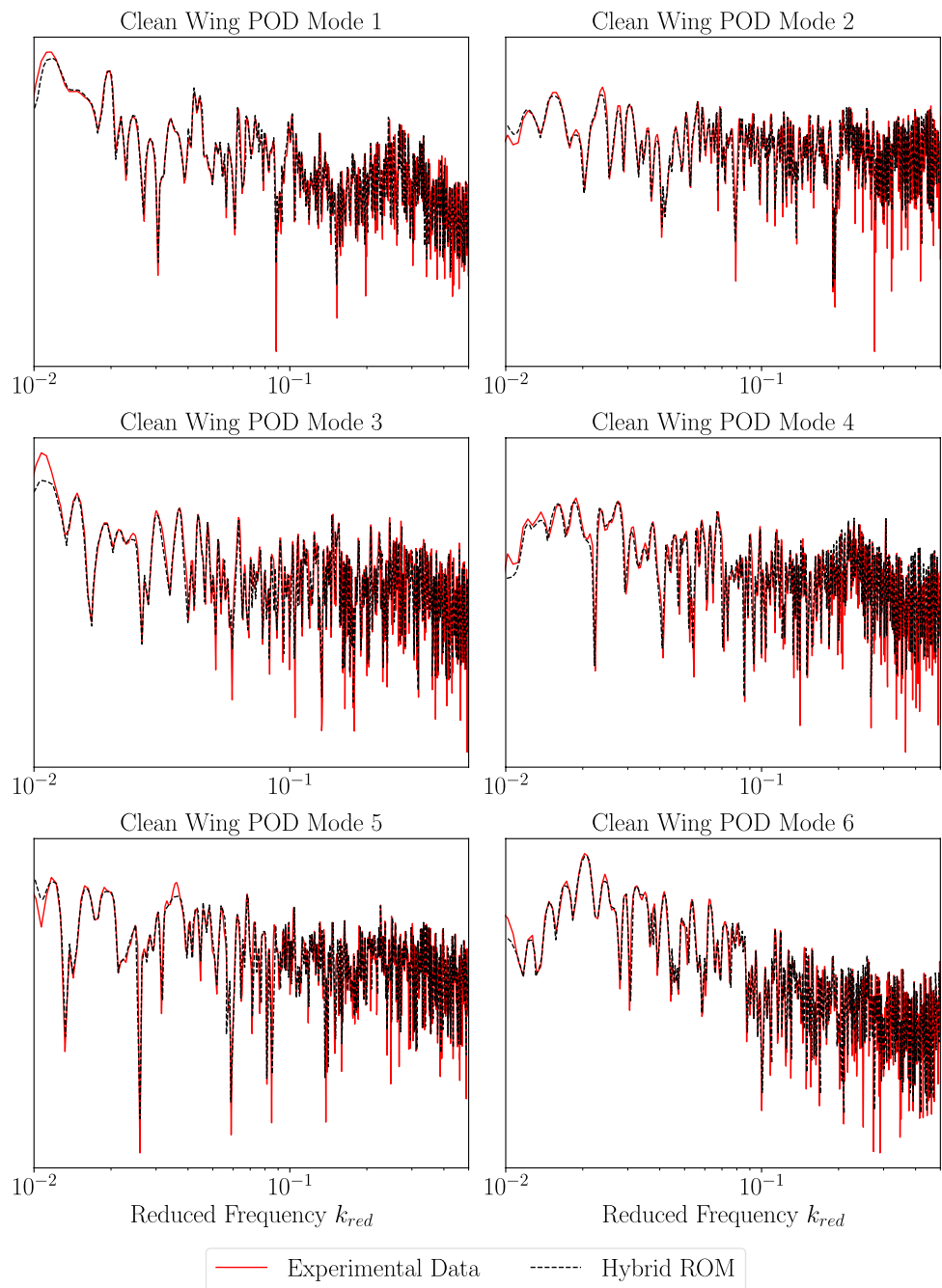
As shown in Figs. 13 and 14, the CNN-VAR-AEs are able to accurately capture both low- and high-frequency content of each mode with a high degree of accuracy. At higher modes, there are some deviations in the frequency amplitude visible; however, the overall trend is represented by the ROMs.

Following the performance evaluation of the trained CNN-VAR-AEs, the trained hybrid model is applied for surface pressure prediction. Therefore, the test data sets are encoded and fed into the LSTM. The LSTM is applied in a recurrent multi-step prediction mode, with the first 32 encoded timesteps applied for initialization. As the prediction advances, the experimental  $c_p$ -snapshots are successively replaced by  $c_p$ -snapshots predicted by the LSTM. In Figs. 15 and 16, a comparison of an original and predicted  $c_p$ -snapshot of the clean wing configuration, is depicted. Therefore, two timesteps  $t = [150, 200]$  are considered. In Figs. 17 and 18, original and predicted  $c_p$ -snapshots of the UHBR configuration are compared considering the same timesteps. Equal to the performance evaluation of the trained CNN-VAR-AEs, the results obtained by the hybrid ROMs are compared to the experimental data sets using POD.

In Figs. 17 and 18, original and predicted  $c_p$ -snapshots of the UHBR configuration are compared considering the same timesteps. Equal to the performance evaluation of the trained CNN-VAR-AEs, the results obtained by the hybrid ROMs are compared to the experimental data sets using POD. Therefore, the first six POD modes of the clean wing and the UHBR wing configuration are compared in Figs. 19 and 20, respectively. Examining the resulting spectra, a good agreement between the reference experimental data and the



**Fig. 19** Power Spectra of the first six POD modes of the buffet cycle (clean wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ ). The experimental results are compared to the results predicted by the hybrid ROM



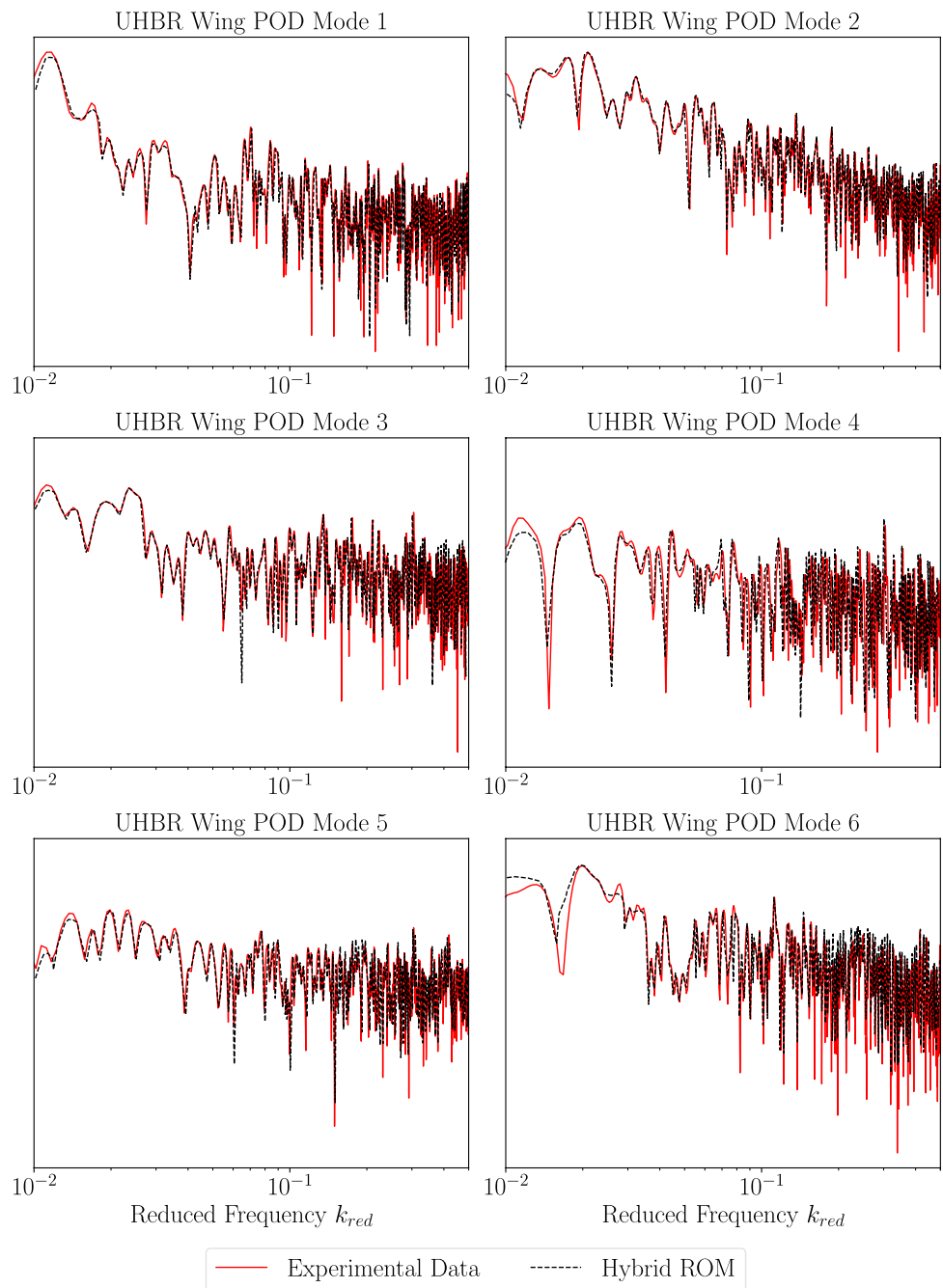
predicted data is shown. Both the low- and high-frequency content of each mode are captured by the hybrid ROM.

## 6 Conclusion

In the present study, a hybrid deep learning model based on a convolutional variational autoencoder (CNN-VAR-AE) and a long short-term memory (LSTM) neural network has been applied for the prediction of wing pressure distributions originated from transonic buffet. For

training the neural network, experimental data as obtained by unsteady pressure sensitive paint (iPSP) measurements in the European Transonic Windtunnel (ETW), has been selected. As a test case, the Airbus XRF-1 configuration has been applied. Two different experimental setups have been considered, defining an aircraft model with clean wings and wings with ultra high bypass ratio (UHBR) nacelle installed. For both test cases, two hybrid models are individually trained. After finalizing the training, the trained reduced-order models (ROMs) are applied for the prediction of buffet pressure loads at a flow condition

**Fig. 20** Power Spectra of the first six POD modes of the buffet cycle (UHBR wing,  $Ma_\infty = 0.9$ ,  $Re = 25$  Mio.,  $\alpha = 5^\circ$ ). The experimental results are compared to the results predicted by the hybrid ROM



which is not included in the training data set. The results obtained by the trained model are compared to experimental data by means of proper orthogonal decomposition. Based on the resulting representation of the POD modes, an accurate prediction is indicated. For both the clean wing and the UHBR wing configuration, the first six POD modes are correctly reproduced by the hybrid ROM. Therefore, a good performance capability of the proposed ROM method for non-linear flow field prediction based on experimental data is shown.

For future studies, it is intended to apply the proposed ROM method for the reconstruction of buffet pressure distributions at different flow conditions, including Mach and Reynolds number variations. Further, to improve the prediction quality of the ROM, the preprocessing routine could be adapted by applying POD or DMD to the data to reduce the noise in the data. In addition, using numerical data or a data set including both numerical and experimental data could also result in an improved performance of the hybrid ROM.

**Acknowledgements** The authors gratefully acknowledge the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) for funding this work in the framework of the research unit FOR2895 (Unsteady flow and interaction phenomena at high speed stall conditions), subproject TP7, Grant number BR1511/14-1. Further, the authors would like to thank the Helmholtz Gemeinschaft HGF (Helmholtz Association), Deutsches Zentrum für Luft - und Raumfahrt DLR (German Aerospace Center) and Airbus for providing the wind tunnel model and financing the wind tunnel measurements.

**Funding** Open Access funding enabled and organized by Projekt DEAL.

**Data availability statement** The data shown in this study is not available, since it contains Airbus restricted data.

## Declarations

**Conflict of interest** The authors declare that they have no conflict of interest.

**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

## References

- Sator, F., Timme, S.: Delayed detached-eddy simulation of shock-buffet on half wing-body configuration. *AIAA J.* **55**(4), 1230–1240 (2016)
- Iovnovich, M., Raveh, D.E.: Numerical study of shock buffet on three-dimensional wings. *AIAA J.* **53**(2), 449–463 (2014)
- Paladini, E., Dandois, J., Sipp, D., Robinet, J.C.: Analysis and comparison of transonic buffet phenomenon over several three-dimensional wings. *AIAA J.* **57**(1), 1–18 (2016)
- Koike, S., Ueno, M., Nakakita, K., Hashimoto, A.: Unsteady pressure measurements of transonic buffet on the NASA common research model. In: 4th AIAA Applied Aerodynamic Conference, AIAA-2016-4044, Washington, D.C. (2016)
- Kou, J., Zhang, W.: Data-driven modeling for unsteady aerodynamics and aeroelasticity. *Prog. Aerosp. Sci.* **125**, 110725 (2021)
- Kou, J., Zhang, W., Yin, M.: Novel Wiener models with a time-delayed nonlinear block and their identification. *Nonlinear Dyn.* **85**(4), 2389–2404 (2016)
- Silva, W.A.: Application of nonlinear systems theory to transonic unsteady aerodynamic responses. *J. Aircr.* **30**(5), 660–668 (1993)
- Glaz, B., Liu, L., Friedmann, P.P.: Reduced-order nonlinear unsteady aerodynamic modeling using a surrogate-based recurrence framework. *AIAA J.* **48**(10), 2418–2429 (2010)
- Mannarino, A., Mantegazza, P.: Nonlinear aeroelastic reduced order modeling by recurrent neural networks. *J. Fluids Struct.* **48**, 103–121 (2014)
- Zhang, W., Kou, J., Wang, Z.: Nonlinear aerodynamic reduced-order model for limit-cycle oscillation and flutter. *AIAA J.* **54**(10), 3304–3311 (2016)
- Winter, M., Breitsamter, C.: Reduced-order modeling of unsteady aerodynamic loads using radial basis function neural networks. In: Deutscher Luft- und Raumfahrtkongress, Bonn (2014)
- Winter, M., Breitsamter, C.: Nonlinear identification via connected neural networks for unsteady aerodynamic analysis. *Aerosp. Sci. Technol.* **77**, 802–818 (2018)
- Rozov, V., Breitsamter, C.: Data-driven prediction of unsteady pressure distributions based on deep learning. *J. Fluids Struct.* **104** (2021)
- Park, K.H., Jun, S.O., Baek, S.M., Cho, M.H., Yee, K.J., Lee, D.H.: Reduced-order model with an artificial neural network for aerostructural design optimization. *J. Aircr.* **50**(4), 1106–1116 (2013)
- Timme, S.: Global instability of wing shock-buffet onset. *J. Fluid Mech.* **885** (2020)
- Ohmichi, Y., Ishida, T., Hashimoto, A.: Modal decomposition analysis of three-dimensional transonic buffet phenomenon on a swept wing. *AIAA J.* **56**(10), 3938–3950 (2018)
- Candon, M., Levinski, O., Altaf, A., Carrese, R., Marzocca, P.: Aircraft transonic buffet load prediction using artificial neural networks. In: 59rd AIAA Structures, Structural Dynamics and Materials Conference, AIAA-2019-0763, San Diego, CA (2019)
- Afshar, Y., Bhatnagar, S., Pan, S., Duraisamy, K., Kaushik, S.: Prediction of aerodynamic flow fields using convolutional neural networks. *Comput. Mech.* **64**, 525–545 (2019)
- Sekar, V., Jiang, Q., Shu, C., Khoo, B.C.: Fast flow field prediction over airfoils using deep learning approach. *Phys. Fluids* **31**, 057103 (2019)
- Li, Y., Chang, J., Wang, Z., Kong, C.: An efficient deep learning framework to reconstruct the flow field sequences of the supersonic cascade channel. *Phys. Fluids* **33**, 056106 (2021)
- Hasegawa, K., Fukami, K., Murata, T., Fukagata, K.: Machine-learning based reduced-order modeling of flows around two-dimensional bluff bodies of various shapes. *Theor. Comput. Fluid Dyn.* **34**, 367–388 (2020)
- Nakamura, T., Fukami, K., Hasegawa, K., Nabae, Y., Fukagata, K.: Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow. *Phys. Fluids* **33**, 025116 (2021)
- LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
- Goodfellow, I., Bengio, Y., Courville, A.: *Deep Learning*, 1st edn. MIT Press, Cambridge. ISBN:978-0-2620-3561-3 (2016)
- Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural Comput.* **9**(8), 1735–1780 (1997)
- Hochreiter, S., Bengio, Y., Frasconi, P., Schmidhuber, J.: *Gradient flow in recurrent nets: the difficulty of learning long-term dependencies*, A Field Guide to Dynamical Recurrent Neural Networks, IEEE Press (2001)
- Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *Proceedings of the 32nd International Conference on Machine Learning*, vol. 37, pp. 448–456 (2015)
- Yorita, D., Klein, C., Henne, U., Ondrus, V., Beifuss, U., Hensch, A.-K., Longo, R., Guntermann, P., Quest, J.: Successful application of cryogenic pressure sensitive paint technique at ETW. In: 55th AIAA Aerospace Sciences Meeting. AIAA SciTech, Kissimmee (2018)
- Liu, T., Sullivan, J.P., Asai, K., Klein, C., Egami, Y.: *Pressure and Temperature Sensitive Paints*, Second Edition (Experimental Fluid Mechanics), Chapter 9. Springer, Berlin (2021)

30. Weiner, A., Semaan, R.: flowTorch—a Python library for analysis and reduced-order modeling of fluid flows. *J. Open Source Softw.* **6**(68), 3860 (2021)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.