NON-LINEAR ORTHOGONAL MODAL DECOMPOSITIONS IN TURBULENT FLOWS VIA AUTOENCODERS

Hamidreza Eivazi
FLOW, Engineering Mechanics
KTH Royal Institute of Technology
Stockholm, Sweden
hamidre@kth.se

Soledad Le Clainche
School of Aerospace Engineering
Universidad Politécnica de Madrid
Madrid, Spain
soledad.leclainche@upm.es

Sergio Hoyas
Inst. Univ. Matemática Pura y Aplicada
Universitat Politècnica de València
Valencia, Spain
sergio.hoyas@mot.upv.es

Ricardo Vinuesa
FLOW, Engineering Mechanics
KTH Royal Institute of Technology
Stockholm, Sweden
rvinuesa@mech.kth.se

ABSTRACT

We propose a deep probabilistic-neural-network architecture for learning a minimal and near-orthogonal set of non-linear modes from high-fidelity turbulent-flow data. Our approach is based on β-variational autoencoders (β-VAEs) and convolutional neural networks (CNNs), which enable extracting non-linear modes from multi-scale turbulent flows while encouraging the learning of independent latent variables and penalizing the size of the latent vector. Moreover, we introduce an algorithm for ordering VAE-based modes with respect to their contribution to the reconstruction. We apply this method for non-linear mode decomposition of the turbulent flow through a simplified urban environment. We demonstrate that by constraining the shape of the latent space, it is possible to motivate the orthogonality and extract a set of parsimonious modes sufficient for high-quality reconstruction. Our results show the excellent performance of the method in the reconstruction against linear-theory-based decompositions. We show the ability of our approach in the extraction of near-orthogonal modes with the determinant of the correlation matrix equal to 0.99, which may lead to interpretability.

INTRODUCTION

Modal-decomposition techniques offer methods to identify a low-dimensional coordinate system for capturing dominant flow features (Taira *et al.*, 2017, 2020) useful for developing reduced-order models, analyzing non-linear and chaotic dynamics, and designing efficient flow-control schemes. Proper-orthogonal decomposition (POD) (Lumley, 1967) and dynamic-mode decomposition (DMD) (Rowley *et al.*, 2009; Schmid, 2010) are two mode-decomposition methods based on linear algebra that have been widely used to extract the dominant spatio-temporal features in fluid flows. Balanced POD (BPOD) (Rowley, 2005), spectral POD (SPOD) (Towne *et al.*, 2018), higher-order DMD (HODMD) (Le Clainche & Vega, 2017) and spatio-temporal Koopman decomposition (STKD) (Clainche & Vega, 2018) are several successful variants of POD and DMD for analysis of turbulent flows.

Besides the aforementioned linear methods for modal decomposition of flow-field data, deep neural networks (DNNs) have shown promising performance in learning a compact latent representation of high-dimensional data by accounting for the non-linearity in the low-dimensional mapping using non-linear activation functions (Hinton & Salakhutdinov, 2006). In particular, unsupervised learning based on autoencoders (AEs) has been shown suitable for efficient mode decomposition and reduced-order modeling with superior performance in flow reconstruction over the linear POD (Milano & Koumoutsakos, 2002; Eivazi *et al.*, 2020).

Moreover, convolutional neural networks (CNNs) (Lecun *et al.*, 1998) and their ability in pattern recognition have received increasing attention by the fluid-mechanics community (Lee & You, 2019; Fukami *et al.*, 2019; Kim & Lee, 2020; Kim *et al.*, 2021; Guastoni *et al.*, 2021). Murata *et al.* (2020) proposed a CNN-based autoencoder architecture for decomposition of flow-fields into non-linear low-dimensional modes and to visualize each mode. They applied this so-called mode-decomposing convolutional-neural-network autoencoder (MD-CNN-AE) to a relatively simple laminar flow around a circular cylinder at $Re_D = 100$ (where Re_D is the Reynolds number based on freestream velocity and cylinder diameter). Their results showed the superior performance of the CNN-based autoencoder over POD where the reconstruction of the flow from only two MD-CNN-AE modes contains also the higher-order POD modes. The architecture of AE-based methods allows a non-linear low-dimensional mapping leading to a superior performance against linear-theory-based methods. However, the AE-based methods do not benefit from the useful properties of the eigenvalue or singular-value-decomposition techniques, e.g., optimality and orthogonality. In contrast to the POD modes, which are an orthogonal set of basis vectors arranged in the order of their energy content, the AE-based modes are neither orthogonal nor ranked. This may lead to the lack of interpretability and robustness of the AE-based modes (Vinuesa & Sirmacek, 2021). In order to obtain ranked modes, Fukami *et al.* (2020) proposed a hierarchical CNN-AE architecture inspired by the concept of hierarchical autoencoder (AE) (Saegusa *et al.*, 2004). The proposed method was first applied to a laminar cylinder wake and its transient process and further to an in-plane cross-sectional velocity field of turbulent channel flow at $Re_\tau = 180$ (note that Re_τ is the friction Reynolds number, based on channel half height and friction velocity). They showed that the hierarchical autoencoder (AE) can rank the AE modes following their contributions to the reconstructed field while achieving efficient order reduction. However, issues related to interpretability and non-uniqueness remained unanswered.

In this paper, we propose a probabilistic method based on β-variational autoencoders (β-VAEs) (Higgins *et al.*, 2017) and CNNs in order to extract a minimal (parsimonious) set of near-orthogonal non-linear modes from turbulent flows. We applied the proposed machine-learning method for the modal decomposition of high-fidelity turbulent-flow simulation data of a simplified urban environment. The flow simulation is carried through well-resolved large-eddy simulation (LES) by means of the spectral-element method. The results from the proposed method are compared with the results from a conventional CNN-AE model, a hierarchical CNN-AE model, and POD. Through the training process of the available AE-based modal-decomposition methods, the objective is to only minimize the reconstruction loss. Therefore, the focus of the available AE-based methods for non-linear modal decomposition is only on the performance of the model in the reconstruction and not on the properties of the learned modes, such as orthogonality and minimality. In contrast, here we minimize the correlation between the latent variables and penalize the size of the latent vector in addition to the minimization of the reconstruction loss. By solving this multi-objective optimization problem using our CNN-βVAE approach, we seek a minimal set of uncorrelated non-linear modes that are able to accurately describe the turbulent flow-field data. The obtained modes are extremely useful for the development of compact reduced-order surrogate models or designing flow-control strategies. Moreover, understanding the physics of the turbulent flow through the extraction of uncorrelated non-linear mechanisms is another incentive for applying CNN-βVAEs for modal decomposition. In particular, the development of accurate predictive models and understanding flow structures in urban environments are of significant importance due to their impact on urban planning, air quality management, and pollutant dispersion (Vinuesa *et al.*, 2015).

We applied POD using the singular-value decomposition (SVD) method on the urban-flow database discussed above. Figure 1 shows the eigenvalues λ_i (left) and the cumulative eigenvalue spectrum $\sum_{j=1}^{j=i}\lambda_j$ (right) normalized with the cumulative sum of the eigenvalues $\sum_{j=1}^{j=m}\lambda_j$, where i indicates the number of modes. We observed that 247 modes are required to capture 99% of the energy as it is depicted by the vertical red line in figure 1 (right). This result implies that it is impractical to represent turbulent flows as a linear superposition of a few modal functions, and thus, more sophisticated algorithms enabling a non-linear modal decomposition are required.

AUTOENCODERS FOR MODAL DECOMPOSITION

An autoencoder is a deep neural network (DNN) with an architecture suitable for unsupervised feature extraction. The network comprises two parts: an encoder that maps the input data to a low-dimensional latent space $\boldsymbol{x} \mapsto \boldsymbol{r}$, and a decoder that projects the latent vector $\boldsymbol{r}$ back to the reference space $\boldsymbol{r} \mapsto \tilde{\boldsymbol{x}}$. We refer to the encoder and decoder parts as $\mathcal{E}$ and $\mathcal{D}$, respectively. Through the model training, the autoencoder learns to extract the most important features in the data that are

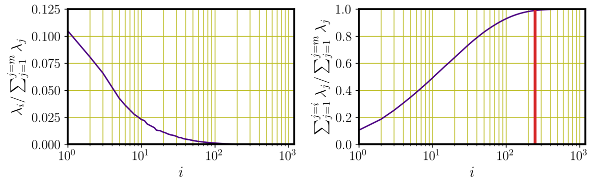

Figure 1. Eigenvalues λ_i (left) and the cumulative eigenvalue spectrum $\sum_{j=1}^{j=i}\lambda_j$ (right) normalized with the cumulative sum of the eigenvalues $\sum_{j=1}^{j=m}\lambda_j$, where i indicates the number of modes. The solid red line shows the number of modes required to capture 99% of the energy.

required for reconstruction by optimizing the model parameters $\boldsymbol{w}$ to minimize the reconstruction loss $\mathcal{L}_{\text{rec}}$. The autoencoder architecture is attractive for modal decomposition as it provides a framework that can incorporate non-linearity in the mappings through the use of non-linear activation functions.

Another challenge in the modal decomposition of turbulent flows is the process of information from input fields that contain multiscale coherent features. The presence of coherent features motivates the use of convolutional layers in autoencoder models to process the input information.

We consider three different types of AEs: CNN-based (CNN-AE), CNN-based hierarchical (CNN-HAE) and CNN-based β-variational (CNN-βVAE). For simplicity, we consider the fluctuating component of the streamwise velocity u as the input/output of the model, but it is also possible to consider all the velocity components (u, v, w) as the input/output. The first convolution layer contains 16 filters with a size of (3×3), and it is followed by a max-pooling layer with $P = 2$. At each convolution step, we double the number of feature maps to extract more information from the turbulent-flow data while at each downsampling step we reduce the dimension. This allows the next layer to combine the features individually identified in each feature map, enabling the extraction of larger and more complex features for progressively deeper convolutional networks from simple non-linear combinations of the previous ones. Therefore, convolutional layers can learn to recognize turbulent-flow patterns of various complexity and scales (Guastoni *et al.*, 2021). After five steps of convolution and max pooling, the extracted features are flattened and fed to fully connected layers to reduce the dimension to the latent vector $\boldsymbol{r}$ with a size of d. The latent vector $\boldsymbol{r}$ is mapped back to the reference space through the consecutive upsampling and convolution operations using nearest-neighbor interpolation. Throughout the model, we use a filter size of (3×3) with the stride of one for convolution layers and (2×2) max pooling and upsampling operations with the same stride. We employ the hyperbolic-tangent (tanh) function $\varphi(z) = (e^z - e^{-z})/(e^z + e^{-z})$ as the non-linear activation function, since it led to the best performance in our study. For all the models, we use mean-squared error as the loss function for reconstruction $\mathcal{L}_{\text{rec}}$ and the Adam algorithm (Kingma & Ba, 2017) to optimize the model parameters $\boldsymbol{w}$. We employ the early-stopping criterion and obtain the best model based on the validation loss to avoid overfitting, where 20% of the data is randomly selected for validation.

The CNN-HAE architecture was proposed by Fukami *et al.* (2020), it is based on a hierarchy of CNN-AEs, and it is aimed at extracting the modes ranked in terms of their contribution to the reconstruction while achieving more efficient data compression. To this end, the first subnetwork $\mathcal{F}_1$ is trained to map the high-dimensional data to a latent vector with size $d = 1$. The latent vector can be obtained using the encoder part of the first subnetwork as $\boldsymbol{r}_1 = \mathcal{E}_1(\boldsymbol{x})$. The second subnetwork $\mathcal{F}_2$ is then trained to reconstruct the input data at the output from a two-dimensional latent vector comprising the first latent vector $\boldsymbol{r}_1$, which has already been obtained, and the second latent vector $\boldsymbol{r}_2$, being updated through the training of $\mathcal{F}_2$, as $[\boldsymbol{r}_1\ \boldsymbol{r}_2]$. The subsequent networks are trained in a similar way. In the present study, we employ the same hyperparameters as those of the CNN-AE model for the CNN-HAE.

Finally, we also use a modified version of the so-called variational autoencoder (VAE) (Kingma & Welling, 2014), which is a probabilistic generative neural architecture emerging from the combination of statistics and information theory. The goal is to map the data into a latent distribution, from which new meaningful samples can be generated. VAEs have gained increasing attention in the scientific community (Maulik *et al.*, 2020), both due to their strong probabilistic foundation and their valuable application in the field of representation learning. The CNN-βVAE employed here, which is represented in Figure 2, has the goal of minimizing the correlation between the latent variables, motivating the network to extract a set of orthogonal modes, also penalizing the size of the latent vector d. This leads to an efficient representation of the high-dimensional data useful for flow analysis, reduced-order modeling, and flow control. Note that the decoder part is the same as that of the CNN-AE model.

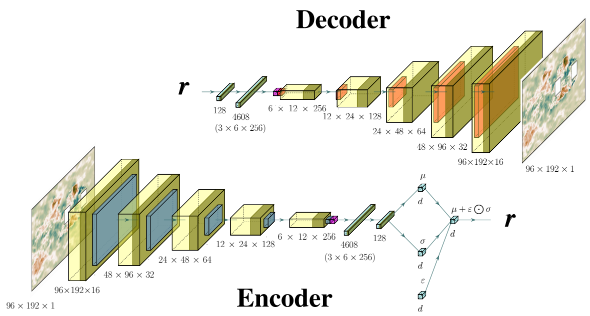

Figure 2. Schematic view of the CNN-βVAE. The color coding for each layer is: 2D-convolution ( ), tanh activation ( ), max pooling ( ), reshape ( ), fully-connected layer ( ), upsampling ( ).

Let us consider a data sample $\boldsymbol{x}$ in some high-dimensional space $\mathcal{X}$ with the distribution $p(\boldsymbol{x})$. VAEs define two probability distributions: $q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})$, a so-called probabilistic encoder (or recognition model) and in a similar vein $p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{r})$, which is referred to as a probabilistic decoder (or generative model). $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ denote the unknown parameters. The objective is optimize the unknown parameters to maximize the so-called marginal likelihood, which can be defined for each $\boldsymbol{x}$ as:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{r}|\boldsymbol{x})) + \mathcal{C}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{x}), \quad (1)$$

where the first right-hand-side (RHS) term is the Kullback–Leibler (KL) divergence D_{KL} between $q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})$ and $p_{\boldsymbol{\theta}}(\boldsymbol{r}|\boldsymbol{x})$. The KL-divergence is non-negative, which indicates that the second RHS term $\mathcal{C}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{x})$ is a lower bound on the marginal likelihood, and can be written as:

$$\begin{aligned}\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) \geq \mathcal{C}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{x}) = \\ -D_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})||p_{\boldsymbol{\theta}}(\boldsymbol{r})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})}\left[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{r})\right].\end{aligned} \quad (2)$$

This term is usually called evidence lower bound (ELBO). Since the operation that samples a latent vector from $q_{\boldsymbol{\phi}(\boldsymbol{r}|\boldsymbol{x})}$ is not differentiable, we need to perform a change of variable, the so-called reparameterization trick (Kingma & Welling, 2014), to differentiate ELBO with respect to both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$. We assume $q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})$ to be a Gaussian distribution,

$$\log q_{\boldsymbol{\phi}(\boldsymbol{r}|\boldsymbol{x})} = \log \mathcal{N}(\boldsymbol{r};\boldsymbol{\mu},\boldsymbol{\sigma}^2\mathbf{I}), \quad (3)$$

where the mean $\boldsymbol{\mu}$ and the standard deviation $\boldsymbol{\sigma}$ are outputs of the encoder, and $\mathbf{I}$ is the identity matrix. We sample from $q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})$ using $\boldsymbol{r} = \boldsymbol{\mu} + \boldsymbol{\sigma} \odot \boldsymbol{\varepsilon}$ where $\boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0},\mathbf{I})$ is an auxiliary normally-distributed random number, and $\odot$ indicates an element-wise product. Note that for the testing steps the mean part of the encoder output $\boldsymbol{\mu}$ of the CNN-βVAE is taken as the vector of latent variables $\boldsymbol{r}$. Moreover, the term $\mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{r}|\boldsymbol{x})}[\log p_{\boldsymbol{\theta}}(\boldsymbol{x}|\boldsymbol{r})]$, which is the so-called log-likelihood, encourages accurate reconstruction of the data and can be estimated as a negative reconstruction error in an autoencoder setting (Kingma & Welling, 2014). This leads to the VAE cost function $\mathcal{C}$, and we can take the negative of it as a loss function $\mathcal{L}$ for training the NNs:

$$\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{x}) = \mathcal{L}_{\mathrm{rec}} - \frac{1}{2}\sum_{i=1}^{d}(1+\log(\sigma_i^2) - \mu_i^2 - \sigma_i^2). \quad (4)$$

In the field of representation learning (Bengio *et al.*, 2013), it is of interest to find a latent representation of the high-dimensional data as an uncorrelated representation with a minimal number of parameters (factors), the so-called disentangled representation, which can be useful for a large variety of tasks and domains. Higgins *et al.* (2017) proposed to augment the original VAE loss function with a single hyperparameter $\beta \geq 0$ that controls the extent of the learning constraints. The goal is to encourage learning of statistically-independent latent variables $\boldsymbol{r}_i$ and penalize the size of the latent vector $\boldsymbol{r}$. This can be obtained by minimizing the distance $D_{\mathrm{KL}}\left[p(\boldsymbol{r})||\prod_i p(\boldsymbol{r}_i)\right]$ between $p(\boldsymbol{r})$ and the product of its marginals. In practice, this is performed by upweighting the KL term in the ELBO, see equation (2), with a penalization factor β leading to the following loss function:

$$\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\phi};\boldsymbol{x}) = \mathcal{L}_{\mathrm{rec}} - \frac{\beta}{2}\sum_{i=1}^{d}(1+\log(\sigma_i^2) - \mu_i^2 - \sigma_i^2) \quad (5)$$

for β-VAEs. A detailed discussion on disentangling in β-VAEs can be found in the works by Higgins *et al.* (2017); Burgess *et al.* (2018); Achille & Soatto (2018); Locatello *et al.* (2019).

RESULTS AND DISCUSSION

The key insight of the present study is to encourage independence of the latent variables $\boldsymbol{r}_1,\ldots,\boldsymbol{r}_d$ to extract near-orthogonal modes from turbulent flows using CNN-βVAEs. This is to motivate disentangled representations in the language of representation learning. We impose a limit on the capacity of the latent information and motivate learning statistically independent latent variables using the penalization factor β. The objective is to motivate orthogonality in the latent space to obtain modes that are useful for flow analysis, reduced-order modeling, and flow control.

We define two evaluation metrics to measure the quality of the reconstructions and the orthogonality (disentanglement) of the latent variables. For reconstruction quality, we evaluate the energy percentage E_u that is captured by the model reconstructions as:

$$E_u = \left(1 - \left\langle \frac{\sum_{i=1}^{n} (u-\tilde{u})^2}{\sum_{i=1}^{n} u^2} \right\rangle \right) \times 100, \tag{6}$$

where $\langle\cdot\rangle$ indicates ensemble averaging in time, u and $\tilde{u}$ denote the reference value of the fluctuating component of streamwise velocity and its reconstruction, respectively, and n is the number of grid points. To measure the independency of the latent variables, we compute the determinant of the correlation matrix multiplied by 100 and refer to it as $\det_{\mathbf{R}}$, where $\mathbf{R} = (R_{ij})_{d\times d}$ is the correlation matrix defined by:

$$R_{ii} = 1 \text{ and } R_{ij} = \frac{C_{ij}}{\sqrt{C_{ii}C_{jj}}}, \tag{7}$$

for all $1 \leq i \neq j \leq d$, and C_{ij} denotes the components i, j of the covariance matrix $\mathbf{C}$. Note that $\det_{\mathbf{R}}$ is 100 when all the variables are completely uncorrelated ($R_{ij} = 0$) and zero when they are completely correlated ($R_{ij} = 1$). We report the value of $\det_{\mathbf{R}}$ as a metric for independency of the latent variables.

We compare the performance of CNN-βVAEs with that of CNN-HAEs, CNN-AEs and POD in terms of reconstruction accuracy and orthogonality of the latent variables. We select $d = 5$ as the size of the latent vector for all models. For the CNN-βVAE we use $\beta = 10^{-3}$. Figure 3 shows the reconstruction of the first time step in the dataset obtained from different methods in comparison with the reference data. It can be observed that a more accurate reconstruction can be obtained using NN-based methods in comparison with 5 POD modes due to the introduction of non-linearity in the algorithm. The CNN-AE model leads to the best reconstructions with $E_u = 94.22\%$ while 32.41% of the energy is captured by the 5 POD modes. Both CNN-HAE and CNN-βVAE models also lead to excellent reconstructions with E_u of 91.84% and 87.36%, respectively, which are slightly lower than that of the CNN-AE. For the CNN-βVAE, it is due to the fact that the regularization with the KL term in the β-VAE loss function, equation (5), induces a trade-off between the reconstruction quality and learning independent representations. However, our results show that excellent reconstructions can be obtained using all the three CNN-AE-based models leading to E_u of about 90% using only 5 modes.

The reconstruction results are also reported in a detailed area between the two obstacles in figure 4 to provide a clear insight into the fidelity of the reconstructions. It can be seen that although some small-scale features are lost, all three CNN-AE models are able to preserve the dominant structures of the turbulent flow. However, POD can not reconstruct the turbulent flow properly from 5 modes.

Next, we compare the independence of the latent variables obtained from different methods. Orthogonality of the modes is a useful property for flow analysis, reduced-order modeling and flow control. Moreover, motivating the orthogonality of the modes may lead to interpretability. Results are depicted in figure 5 as the absolute value of the correlation matrix $\mathbf{R}$ corresponding to the latent variables from the CNN-HAE, CNN-AE, CNN-βVAE with $d = 5$, and also POD with 5 modes. It can be seen that although the CNN-HAE model extract modes in the order of their contribution in the reconstruction, the latent variables are correlated leading to the lowest value for $\det_{\mathbf{R}}$ among all methods. As mentioned above, it is possible to motivate the disentanglement or independence of the latent variables using CNN-βVAEs and obtain near-orthogonal modes. It can be observed that the correlation between the latent variables is reduced for the CNN-βVAE in comparison to that of the CNN-AE, where $\det_{\mathbf{R}}$ is equal to 99.20 for the CNN-βVAE method and 87.59 for the CNN-AE technique.

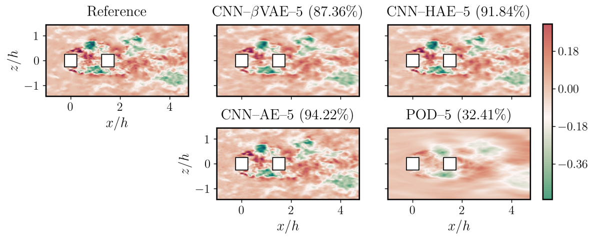

Figure 3. Reconstruction of the fluctuating component of the streamwise velocity obtained from different methods, as indicated in each panel, in comparison with the reference data. The value in brackets on each panel indicates the obtained E_u.

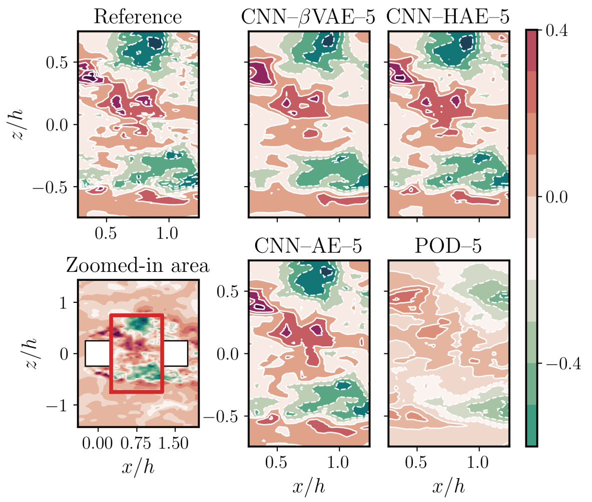

Figure 4. Reconstruction of the fluctuating component of the streamwise velocity obtained from different methods, as indicated on each panel, in comparison with the reference data for the zoomed-in area between the two obstacles marked by the red rectangle.

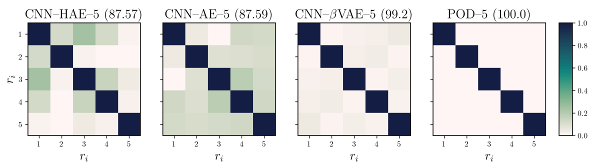

Figure 5. Correlation matrix $\mathbf{R}$ for the latent variables obtained from different models as indicated on the panels. The value in brackets indicates the corresponding $\det_{\mathbf{R}}$ for each case.

RANKING THE CNN-βVAE MODES

POD modes are sorted in terms of their energy content. This property is extremely useful for understanding and analyzing the dominant patterns in complex flows. Fukami *et al.* (2020) implemented hierarchical autoencoders to extract AE-based modes in the order of their contribution in the reconstruction, which requires training multiple NNs and might be cumbersome especially for the extraction of higher-order modes. Here, we propose a strategy for ranking the CNN-βVAE modes. We showed that CNN-βVAEs are able to extract near-orthogonal and parsimonious modes from turbulent flows. These properties allow us to rank these modes after the training process and based on their contribution to the reconstruction. In particular, we rank CNN-βVAE modes based on the maximum E_u that can be obtained from q modes, where q represents the rank. To this end, after the training process, we first use the encoder to map high-dimensional data to the latent vector $\mathcal{E} : \boldsymbol{x} \mapsto \boldsymbol{r}$. We zero out all the latent variables except the i^{th} variable, which leads to a latent vector $\hat{\boldsymbol{r}}_i$. Then, we employ the decoder part of the CNN-βVAE to send this latent vector to the original space $\mathcal{D} : \hat{\boldsymbol{r}}_i \mapsto \tilde{\boldsymbol{x}}_i$. This procedure is performed for all the time steps. The energy percentage that is captured by only considering the i^{th} mode is evaluated as E_k^i. The first mode is selected as the mode leading to the maximum value of E_k^i. For the second mode, we perform the same procedure while we preserve the first mode and look for the mode, which in combination with the first mode, leads to the maximum value of E_u^i. In a similar way, the third mode is selected as the mode which gives the maximum E_u^i in combination with the first and second modes. We continue this procedure to rank all the modes. Figure 6 illustrates the ranked modes obtained from a CNN-βVAE model with $d = 5$ and $\beta = 10^{-3}$, together with the modes obtained from the POD, CNN-AE, and CNN-HAE methods with $d = 5$. The POD and the CNN-HAE modes are already ranked and we also perform the ordering procedure for the CNN-AE modes. A clear resemblance can be observed between the first two modes of the CNN-βVAE, see figure 6(a), and those from POD, as shown in figure 6(b), indicating the ability of CNN-βVAEs in the extraction of interpretable modes from turbulent flows. These modes correspond to the large-scale vortex shedding from around the obstacles into the wake region. Moreover, these results show that using the ranking procedure it is possible to sort the CNN-βVAE modes based on their importance for reconstruction. It also can be seen in figure 6(c) that it is extremely difficult to relate CNN-AE modes to physical processes, a fact that is referred to as the lack of interpretability. We observe that although the CNN-HAE model is able to extract large-scale features first, the obtained modes may not be physically interpretable, as shown in figure 6(d).

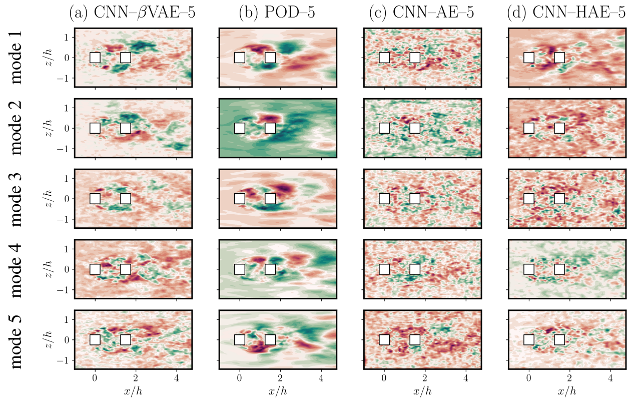

Figure 6. The ranked spatial modes obtained from the CNN-βVAE (a), POD (b), CNN-AE (c), and the CNN-HAE (d). The size of the latent vector d is equal to 5.

SUMMARY AND CONCLUSIONS

In this study, we propose a probabilistic deep-neural-network architecture based on β-VAEs and CNNs for non-linear mode decomposition of turbulent flows. The objective is to learn a compact, near-orthogonal and parsimonious latent representation of high-dimensional data by introducing non-linearity in the process of dimension reduction and also minimizing the correlation between the latent variables, as well as penalizing the size of the latent vector. This may lead to a set of interpretable modes useful for flow analysis, reduced-order modeling and flow control. Since the correlations among the learned latent variables are minimized, we proposed an algorithm to rank the VAE-based modes based on their contribution to the reconstruction. We applied the proposed CNN-βVAE architecture for modal decomposition of the turbulent flow through a simplified urban environment. Furthermore, we compared the performance of the CNN-βVEs in terms of the quality of the reconstructions and orthogonality of the extracted modes with that of the CNN-AEs and CNN-HAEs. Our results from modal decomposition using POD indicates that 247 modes are required to obtain 99% of the energy from the reconstruction. This indicates that it is challenging to represent turbulent flows as a linear superposition of a few POD modes. Our proposed CNN-βVAE model with a latent vector size $d = 5$ and a penalization factor of $\beta = 10^{-3}$ leads to E_u of 87.36% against 32.41% obtained from POD, which shows the excellent performance of the CNN-βVAE in the reconstruction of the turbulent flow from only five modes. This model also leads to near-orthogonal modes, where $\det_{\mathbf{R}}$ is equal to 99.20. We showed that by constraining the shape of the latent space and motivating orthogonality of the modes, we can extract meaningful non-linear features where the first mode of this CNN-βVAE model represents the large-scale vortex shedding from around the obstacles into the wake. Our comparison between the CNN-βVAE, CNN-HAE, and CNN-AE models indicates that although motivating orthogonality of the modes decreases the reconstruction accuracy, very good reconstructions can be obtained from five modes using the CNN-βVAE leading to E_u of 87.36% against 93.93% and 91.84% of the CNN-AE and the CNN-HAE, respectively. The CNN-βVAE model leads to a set of near-orthogonal modes with the highest $\det_{\mathbf{R}}$ among the AE-based models.

The proposed CNN-βVAE architecture can be extended in future works for the development of reduced-order surrogate models or advanced flow-control methods, among others. In particular, the proposed method can be employed for non-linear modal decomposition of the data obtained from thermal-imaging cameras in complex urban environments, with extensive application in the context of urban air-quality forecast and control.

ACKNOWLEDGMENTS

We acknowledge Álvaro Martínez for his contributions to this work. RV acknowledges the Göran Gustafsson foundation for the financial support of this research. SH has been supported by project RTI2018-102256-B-I00 of Mineco/FEDER.

SLC acknowledges the support of the Spanish Ministry of Science and Innovation under the grant PID2020-114173RB-100.

REFERENCES

Achille, Alessandro & Soatto, Stefano 2018 Information dropout: Learning optimal representations through noisy computation. *IEEE Trans. Pattern Anal. Mach. Intell.* **40** (12), 2897–2905.

Bengio, Yoshua, Courville, Aaron & Vincent, Pascal 2013 Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.* **35** (8), 1798–1828.

Burgess, Christopher P., Higgins, Irina, Pal, Arka, Matthey, Loic, Watters, Nick, Desjardins, Guillaume & Lerchner, Alexander 2018 Understanding disentangling in β-VAE.

Clainche, Soledad Le & Vega, José M. 2018 Spatio-temporal Koopman decomposition. *J. Nonlinear Sci.* **28** (5), 1793–1842.

Eivazi, Hamidreza, Veisi, Hadi, Naderi, Mohammad Hossein & Esfahanian, Vahid 2020 Deep neural networks for nonlinear model order reduction of unsteady flows. *Phys. Fluids* **32** (10), 105104.

Fukami, Kai, Fukagata, Koji & Taira, Kunihiko 2019 Super-resolution reconstruction of turbulent flows with machine learning. *J. Fluid Mech.* **870**, 106–120.

Fukami, Kai, Nakamura, Taichi & Fukagata, Koji 2020 Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Phys. Fluids* **32** (9), 095110.

Guastoni, Luca, Güemes, Alejandro, Ianiro, Andrea, Discetti, Stefano, Schlatter, Philipp, Azizpour, Hossein & Vinuesa, Ricardo 2021 Convolutional-network models to predict wall-bounded turbulence from wall quantities. *J. Fluid Mech.* **928**, A27.

Higgins, I., Matthey, Loïc, Pal, A., Burgess, Christopher P., Glorot, Xavier, Botvinick, M., Mohamed, S. & Lerchner, Alexander 2017 beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*.

Hinton, G. E. & Salakhutdinov, R. R. 2006 Reducing the dimensionality of data with neural networks. *Science* **313** (5786), 504–507.

Kim, Hyojin, Kim, Junhyuk, Won, Sungjin & Lee, Changhoon 2021 Unsupervised deep learning for super-resolution reconstruction of turbulence. *J. Fluid Mech.* **910**, A29.

Kim, Junhyuk & Lee, Changhoon 2020 Prediction of turbulent heat transfer using convolutional neural networks. *J. Fluid Mech.* **882**, A18.

Kingma, Diederik P. & Ba, Jimmy 2017 Adam: A method for stochastic optimization.

Kingma, Diederik P & Welling, Max 2014 Auto-encoding variational Bayes.

Le Clainche, Soledad & Vega, José M. 2017 Higher order dynamic mode decomposition. *SIAM J. Appl. Dyn. Syst.* **16** (2), 882–925.

Lecun, Y., Bottou, L., Bengio, Y. & Haffner, P. 1998 Gradient-based learning applied to document recognition. *Proc. IEEE* **86** (11), 2278–2324.

Lee, Sangseung & You, Donghyun 2019 Data-driven prediction of unsteady flow over a circular cylinder using deep learning. *J. Fluid Mech.* **879**, 217–254.

Locatello, Francesco, Bauer, Stefan, Lucic, Mario, Rätsch, Gunnar, Gelly, Sylvain, Schölkopf, Bernhard & Bachem, Olivier 2019 Challenging common assumptions in the unsupervised learning of disentangled representations.

Lumley, J L 1967 The structure of inhomogeneous turbulent flows. *Atmospheric Turbulence and Radio Wave Propogation. A. M. Yaglom and V. I. Tatarski (eds). Nauka, Moscow* pp. 166–78.

Maulik, Romit, Fukami, Kai, Ramachandra, Nesar, Fukagata, Koji & Taira, Kunihiko 2020 Probabilistic neural networks for fluid flow surrogate modeling and data recovery. *Phys. Rev. Fluids* **5**, 104401.

Milano, Michele & Koumoutsakos, Petros 2002 Neural network modeling for near wall turbulent flow. *J. Comput. Phys.* **182** (1), 1–26.

Murata, Takaaki, Fukami, Kai & Fukagata, Koji 2020 Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *J. Fluid Mech.* **882**, A13.

Rowley, C. W. 2005 Model reduction for fluids using balanced proper orthogonal decomposition. *Int. J. Bifurcation Chaos* **15** (03), 997–1013.

Rowley, Clarence W., Mezić, Igor, Bagheri, Shervin, Schlatter, Philipp & Henningson, Dan S. 2009 Spectral analysis of nonlinear flows. *J. Fluid Mech.* **641**, 115–127.

Saegusa, Ryo, Sakano, Hitoshi & Hashimoto, Shuji 2004 Nonlinear principal component analysis to preserve the order of principal components. *Neurocomputing* **61**, 57–70.

Schmid, Peter J. 2010 Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **656** (November), 5–28.

Taira, Kunihiko, Brunton, Steven L., Dawson, Scott T. M., Rowley, Clarence W., Colonius, Tim, McKeon, Beverley J., Schmidt, Oliver T., Gordeyev, Stanislav, Theofilis, Vassilios & Ukeiley, Lawrence S. 2017 Modal analysis of fluid flows: An overview. *AIAA J.* **55** (12), 4013–4041.

Taira, Kunihiko, Hemati, Maziar S., Brunton, Steven L., Sun, Yiyang, Duraisamy, Karthik, Bagheri, Shervin, Dawson, Scott T. M. & Yeh, Chi-An 2020 Modal analysis of fluid flows: Applications and outlook. *AIAA J.* **58** (3), 998–1022.

Towne, Aaron, Schmidt, Oliver T. & Colonius, Tim 2018 Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis. *J. Fluid Mech.* **847**, 821–867.

Vinuesa, Ricardo, Schlatter, Philipp, Malm, Johan, Mavriplis, Catherine & Henningson, Dan S. 2015 Direct numerical simulation of the flow around a wall-mounted square cylinder under various inflow conditions. *J. Turbul.* **16** (6), 555–587.

Vinuesa, Ricardo & Sirmacek, Beril 2021 Interpretable deep-learning models to help achieve the sustainable development goals. *Nat. Mach. Intell.* **3** (11), 926–926.