Visualization of nonlinear modal structures for three-dimensional unsteady fluid flows with customized decoder design

Kazuto Hasegawa[1], **Kai Fukami**[2], **Koji Fukagata**[1]
1. Mechanical Engineering, Keio University, Yokohama, 223-8522, Japan
2. Mechanical and Aerospace Engineering, University of California, Los Angeles, CA 90095, USA
fukagata@mech.keio.ac.jp

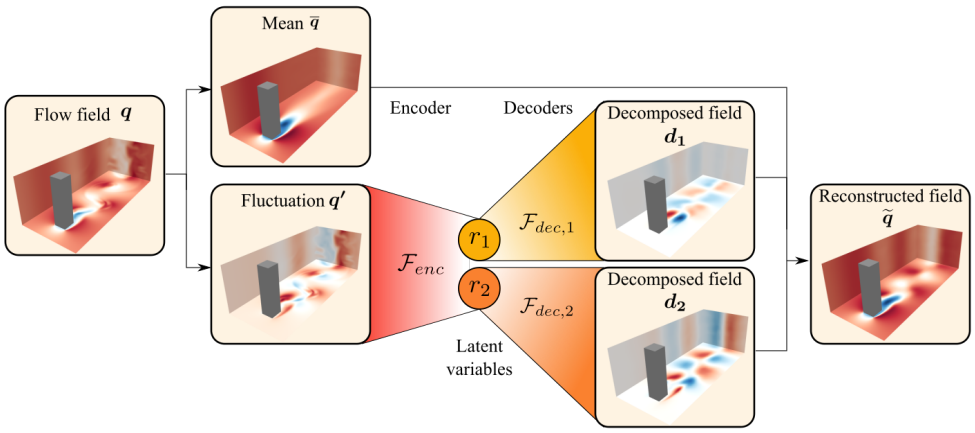

Figure 1: Schematic of mode-decomposing convolutional neural network-based autoencoder.

Abstract

Understanding nonlinear manifolds of scientific data extracted via autoencoder is important to propel practical uses of non-intrusive reduced-order modeling in the community. We here tackle this matter by visualizing nonlinear autoencoder modes with the aid of mode-decomposing convolutional neural network autoencoder (MD-CNN-AE). The MD-CNN-AE has a customization in the decoder part, which enable us to visualize individual modes extracted through the encoder part. The present demonstration is performed with a three-dimensional flow around a square cylinder at $Re_D = 300$, which possesses complex nonlinear vortical phenomena associated with strong nonlinearities. The results are compared with a conventional linear model order reduction method, i.e, principal component analysis (PCA). The reconstructed fields with MD-CNN-AE hold more energetic information than that with PCA, despite the same number of latent variables. The present results indicate the strong capability of MD-CNN-AE for efficient low-dimensionalization and data compression of three-dimensional flow fields in an interpretable manner.

1 Introduction

Thus far, intrusive reduced-order modeling (ROM) has gained attentions in science and engineering community to handle systems with a spatio-temporal high-degree of freedom. It is not exception in fluid mechanics, and the field has received enormous benefits of intrusive ROM due to complexities

Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021).

of fluid flow phenomena. One of the most typical methods is principal component analysis (PCA) [1] that expresses data with several principal and orthogonal modes and corresponding eigenvalues. Although PCA provides physically interpretable insights for us, it is also fact that applying to fluid flow phenomena is still challenging due to the linear operation of PCA. For example, Alfonsi et al. [2] reported that 7260 PCA modes are needed to reconstruct a turbulent channel flow at friction Reynolds number $Re_\tau = 180$ with 95% energy, which cannot be regarded as interpretable and handleable numbers. To tackle this issue, non-intrusive ROM supported by data-driven analysis rather than theoretical approaches has emerged as a powerful tool to incorporate nonlinearities into its modeling [3]. This type of nonlinear ROM is suitable for complex fluid flows and outperform the linear PCA [4]. However, one of the issues here is interpretability of results provided by non-intrusive ROM, which is caused by “black-box" treatment of machine learning. To address this interpretability in fluid mechanics, Murata et al. [5] has recently proposed mode-decomposing convolutional neural network-based autoencoder (MD-CNN-AE) to visualize nonlinear modes corresponding to each latent variable and applied to a two-dimensional cylinder flow. But since the investigations were limited to two-dimensional unsteady laminar flows, we here examine the applicability of MD-CNN-AE considering a three-dimensional flow around a square cylinder while comparing to PCA.

2 Modal order reduction techniques

2.1 Mode-decomposing convolutional neural network-based autoencoder (MD-CNN-AE)

Let us first introduce the schematic of the present MD-CNN-AE in figure 1. A flow field $\boldsymbol{q}$ is decomposed into a mean $\overline{\boldsymbol{q}}$ and fluctuation field $\boldsymbol{q}'$. In this study, the fluctuation field is then low-dimensionalized into latent variables $\boldsymbol{r}$ using a encoder part $\mathcal{F}_{enc}$. Each decoder $\mathcal{F}_{dec,i}$ corresponding to each latent variable r_i then recovers a decomposed field $\boldsymbol{d}_i$. The reconstructed field $\widetilde{\boldsymbol{q}}$ is finally obtained by adding the decomposed fields and the mean field. These operations can be summarized as

$$\boldsymbol{r} = \mathcal{F}_{enc}(\boldsymbol{q}'), \quad \widetilde{\boldsymbol{q}} = \overline{\boldsymbol{q}} + \sum_{i=1}^{n_r} \boldsymbol{d}_i = \overline{\boldsymbol{q}} + \sum_{i=1}^{n_r} \mathcal{F}_{dec,i}(r_i), \tag{1}$$

where the number of latent variables n_r can be changed. As illustrated in figure 1, we set $n_r = 2$ following the previous study by Murata et al. [5]. Note that the computational cost of MD-CNN-AE is proportional to the number of latent variables since each latent variable requires an individual decoder. The weight optimization of the present MD-CNN-AE is performed minimizing a loss function that is defined between a solution of fluctuation $\boldsymbol{q}'$ and a decoded field $\sum_{i=1}^{n_r} \boldsymbol{d}_i$ such that

$$\boldsymbol{w} = \mathrm{argmin}_{\boldsymbol{w}} ||\boldsymbol{q}' - \sum_{i=1}^{n_r} \boldsymbol{d}_i||_2, \tag{2}$$

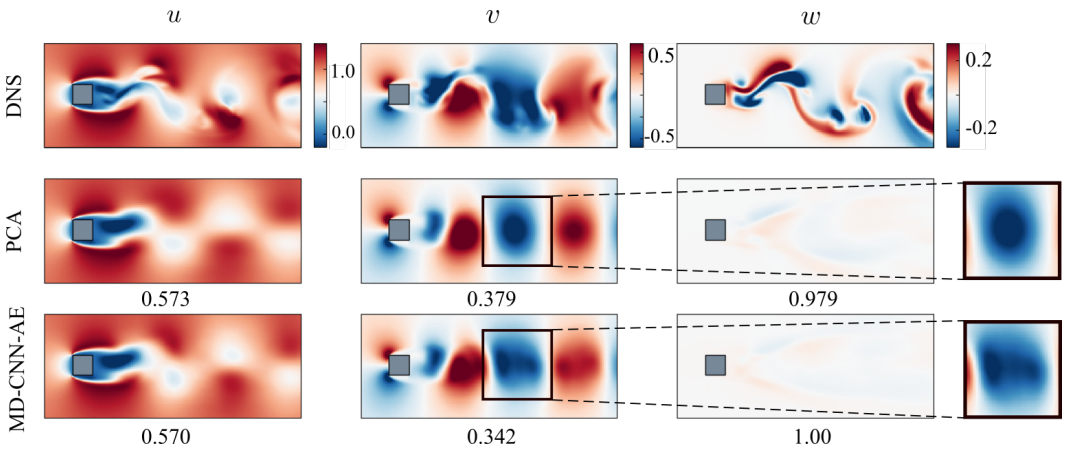

Figure 2: Comparison of the reference DNS flow fields and the reconstructed fields by MD-CNN-AE and two PCA modes. Values underneath the velocity contours represent the L_2 error norm.

where $\boldsymbol{w}$ denotes weights of MD-CNN-AE. We use the L_2 error norm to determine the cost function and Adam optimizer [6] for the present training. The hyperbolic tangent function is used as an activation function, which is also the same setting as Murata et al. [5].

2.2 Principal component analysis (PCA)

In this study, the comparison between MD-CNN-AE and PCA is also performed. With PCA, the reconstructed flow field $\widetilde{\boldsymbol{q}}_{\mathrm{PCA}}$ is expressed as

$$\widetilde{\boldsymbol{q}}_{\mathrm{PCA}} = \overline{\boldsymbol{q}} + \sum_{i=1}^{n_r} a_i \boldsymbol{\phi}_i, \tag{3}$$

where a_i denotes temporal PCA coefficients and $\boldsymbol{\phi}_i$ represents Fourier modes that are orthogonal with each other among modes. Note that MD-CNN-AE with linear activation function and the L_2 error-based optimization is equivalent to PCA [5].

3 Application to three-dimensional flow around a square cylinder

The present MD-CNN-AE is applied to a three-dimensional square cylinder wake simulated by direct numerical simulation (DNS). The Reynolds number Re_D is set to 300 and the velocity fields $\boldsymbol{q} = \{u, v, w\}$ are used as quantities of interest. The size of the domain and the number of grid points are $(L_x, L_y, L_z) = (12.8D, 4D, 4D)$ and $(N_x, N_y, N_z) = (256, 128, 160)$, where D represents the side length of the square cylinder. We use 3000 snapshots as training data and another 3000 snapshots for evaluation. The present training and evaluation are performed under the NVIDIA Tesla V100 graphical processing units.

We first examine the capability to reconstruct the flow fields of MD-CNN-AE. The reconstructed fields at $z/D = 2.0$ are presented in figure 2 comparing with the reference DNS fields and the

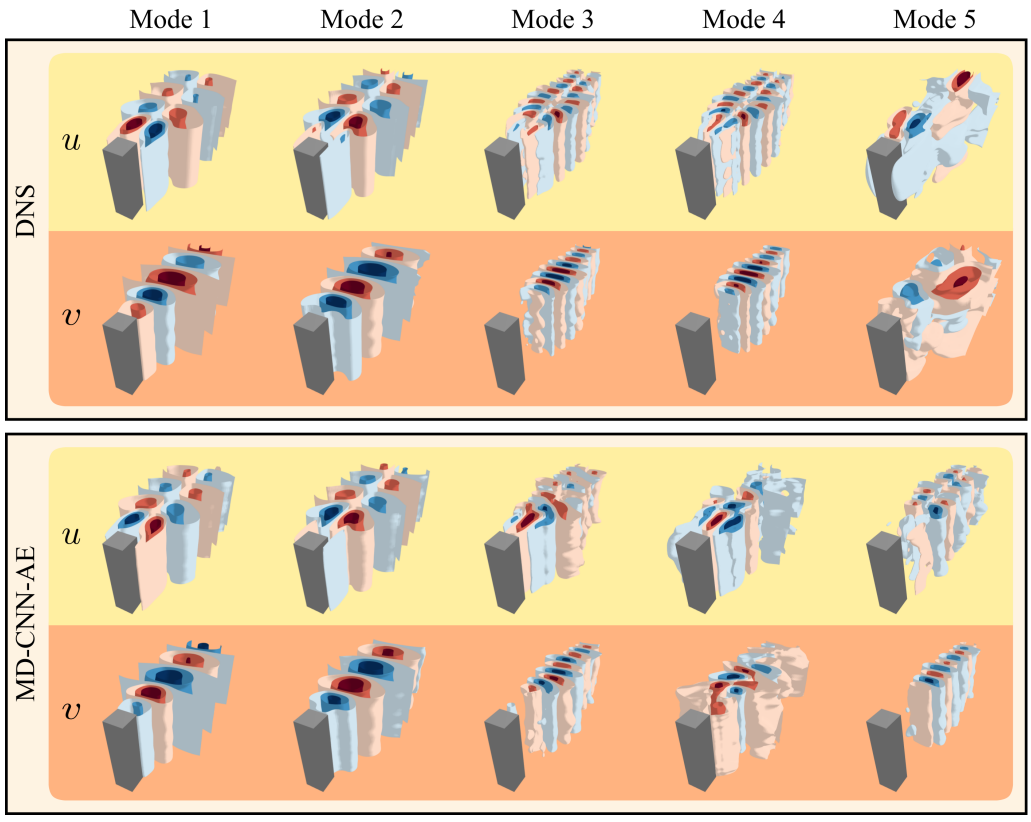

Figure 3: PCA modes of the reconstructed fields with MD-CNN-AE and the reference DNS fields.

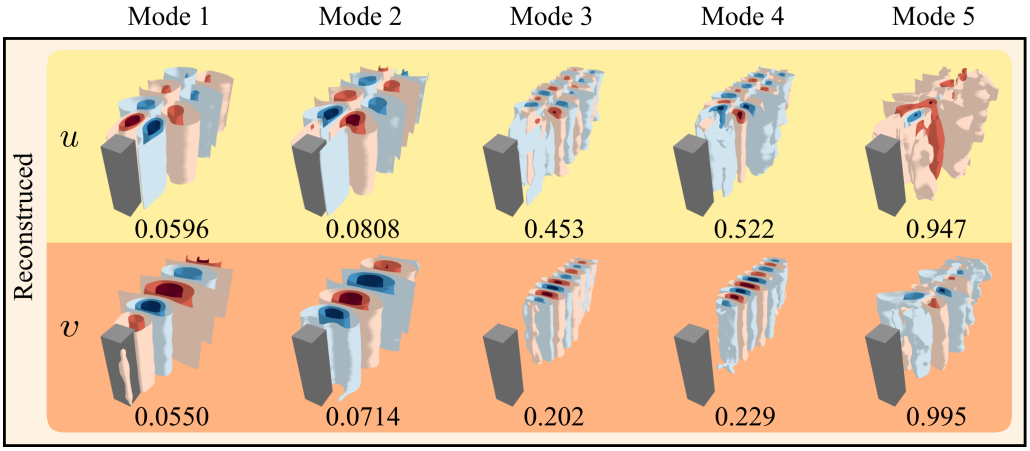

Figure 4: The reconstructed PCA modes of the DNS fields by the PCA modes of the reconstructed fields by MD-CNN-AE. Values underneath the reconstructed fields represent the L_2 error norm.

fields reconstructed by two PCA modes. The values underneath the reconstructed fields represent the L_2 error norm ϵ defined as $\epsilon = ||\boldsymbol{q} - \widetilde{\boldsymbol{q}}||_2/||\boldsymbol{q}'||_2$. As shown, the velocities u and v are well reconstructed with both MD-CNN-AE and PCA, although the reconstruction for the w component becomes difficult due to complex configurations. In particular, we find that MD-CNN-AE can capture finer-scale structure of velocities, that cannot be achieved with PCA, by seeing the zoomed-in portions of figure 2.

Let us then perform PCA for the reconstructed fields with MD-CNN-AE. This investigation tells us whether multiple orthogonal modes are included in the recovered fields via MD-CNN-AE or not [5]. Hereafter, we focus on the behaviors on the u and v components following the observation in figure 2. The isosurface contours of the u and v components of the PCA modes of the reconstructed fields with MD-CNN-AE are presented with PCA mode of the reference DNS fields in figure 3. Noteworthy here is that the MD-CNN-AE fields contain several orthogonal bases that akin to the PCA modes 1, 2, 3, and 4 of the DNS fields in its modes 1, 2, 3, and 5. This implies that the present MD-CNN-AE can capture more than two orthogonal modes despite that the number of latent space is set to two, that represents the better compression capability of the MD-CNN-AE.

To quantitatively evaluate this point, we express the PCA modes of the DNS fields using a linear combination of the PCA modes of the MD-CNN-AE fields as $\boldsymbol{\phi}_i = \sum_j \alpha_j \boldsymbol{\varphi}_j$, where $\boldsymbol{\phi}_i$ is the target ith PCA modes of the DNS fields, $\boldsymbol{\varphi}_j$ represents jth PCA modes of the reconstructed MD-CNN-AE fields, and α_j denotes the coefficient of jth modes of MD-CNN-AE fields. The coefficient $\boldsymbol{\alpha}$ can be obtained by linear regression, i.e., $\boldsymbol{\alpha} = \mathrm{argmin}_{\boldsymbol{\alpha}}||\boldsymbol{\phi} - \boldsymbol{\varphi}\boldsymbol{\alpha}||_2$. The results of the reconstruction are summarized in figure 4. The u and v components of PCA modes 1, 2, 3, and 4 of the DNS fields are well reconstructed with the PCA modes of the MD-CNN-AE fields. This indicate that the MD-CNN-AE reconstructed fields indeed contain the PCA modes 1, 2, 3, and 4 of the DNS fields thanks to the nonlinear activation function inside the model. Summarizing above, the present analyses with MD-CNN-AE enable us to clearly identify the strength of nonlinear functional low-dimensionalization in an understandable manner.

4 Conclusions

We applied mode-decomposing convolutional neural network-based autoencoder (MD-CNN-AE) to a three-dimensional flow field around a square cylinder at $Re_D = 300$. The flow fields were successfully low-dimensionalized to two latent variables and the reconstruction by MD-CNN-AE outperformed that by two PCA modes. Furthermore, the reconstructed fields by MD-CNN-AE were analysed by applying PCA to examine whether the fields contain multiple orthogonal PCA modes or not. We found that the PCA modes 1 to 4 of the reference DNS fields can be included in the reconstructed fields despite that the number of the AE latent variables is set to 2. This results

indicate that the present MD-CNN-AE can low-dimensionalize the flow fields more efficiently than the conventional linear theory-based reduced-order modeling.

In addition to the demonstrated strengths of MD-CNN-AE that can acquire an individual mode corresponding to each latent variable, it is also possible to monitor time-varying MD-CNN-AE mode fields $\boldsymbol{d}(t)$. This capability is extremely useful for investigations into controlled systems because we are able to observe how individual MD-CNN-AE modes respond depending on control effects. The present MD-CNN-AE can also be applied to high Reynolds number flows and turbulence, although more number of modes may be required for qualitative reconstruction than for the present case [4]. For such situations, the optimization of hyper parameters should be helpful since the number of branches at the decoder part increases with the number of latent modes [7].

References

[1] J. L. Lumley. The structure of inhomogeneous turbulent flows. In A. M. Yaglom and V. I. Tatarski, editors, *Atmospheric turbulence and radio wave propagation*. Nauka, 1967.

[2] G. Alfonsi and L. Primavera. The structure of turbulent boundary layers in the wall region of plane channel flow. *Proc. R. Soc. A*, 463(2078):593–612, 2007.

[3] S. L. Brunton, B. R. Noack, and P. Koumoutsakos. Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.*, 52:477–508, 2020.

[4] T. Nakamura, K. Fukami, K. Hasegawa, Y. Nabae, and K. Fukagata. Convolutional neural network and long short-term memory based reduced order surrogate for minimal turbulent channel flow. *Phys. Fluids*, 33:025116, 2021.

[5] T. Murata, K. Fukami, and K. Fukagata. Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *J. Fluid Mech.*, 882:A13, 2020.

[6] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. arXiv:1412.6980, 2014.

[7] T. Nakamura, K. Fukami, and K. Fukagata. Comparison of linear regressions and neural networks for fluid flow problems assisted with error-curve analysis. arXiv:2105.00913, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] The discussion of the computational cost presented in section 2.1.
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] The code and the data are proprietary.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] The details are presented in section 2 and 3.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See section 3.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [No] All the data utilized in this study belongs to the authors.
   (b) Did you mention the license of the assets? [No] All the data utilized in this study belongs to the authors.
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] All the data utilized in this study belongs to the authors.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No] All the data utilized in this study belongs to the authors.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]