ParCFD2024
35th International Conference on Parallel Computational Fluid Dynamics
Sep. 02-04, 2024, Bonn, Germany

Investigating the Effects of Spanwise Transversal Traveling Waves on a Turbulent Compressible Flat Plate Flow With the Aid of a Deep Autoencoder Network

Xiao Shao[a,*], Hilmi Oguzhan Ayan[a], Fabian Hübenthal[a], Mario Rüttgers[b], Andreas Lintermann[b] and Wolfgang Schröder[a]

[a]RWTH Aachen University, Institute of Aerodynamics, Wüllnerstraße 5a, 52062 Aachen, Germany
[b]Forschungszentrum Jülich GmbH, Jülich Supercomputing Centre, Wilhelm-Johnen-Straße, 52428 Jülich, Germany

ARTICLE INFO[†]

Keywords:
Drag Reduction;
Autoencoder;
Machine Learning;
Traveling Waves;
Deep Learning

ABSTRACT

The impact of spanwise traveling transversal surface waves on drag reduction in turbulent compressible flat plate flow is explored. The findings indicate that when the traveling phase speed approaches the freestream velocity at $M = 0.7$, a shock wave is induced in the spanwise direction. This shock wave effectively breaks down streamwise vortices into smaller scales, which significantly enhances drag reduction. The spanwise shock wave is a large-scale quasi-periodic phenomenon. To understand its impact on the multi-scale nature of turbulent flows, a nonlinear mode decomposing deep convolutional autoencoder is employed. The results show that the autoencoder reconstructs the flow field more accurately compared to Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD). Additionally, it effectively separates the large-scale spanwise shock wave and small-scale turbulent structures, which achieves a clearer distinction than POD and DMD.

1. Introduction

In civil aviation, skin friction contributes significantly to the total aerodynamic drag, accounting for up to 50% [1], which leads to higher fuel consumption. Therefore, reducing skin friction is paramount for environmental reasons. Drag reduction techniques can be classified into two main groups: active, e.g., spanwise wall oscillations [2] or spanwise traveling waves [3], and passive methods, e.g., riblets [4]. While the former requires an introduction of energy input, the latter doesn't depend on external energy. The basic idea of spanwise traveling transversal surface waves is to introduce a secondary flow field by introducing a wavy motion to the surface, resulting in a drag reduction.

Combining the fast-emerging field of machine learning (ML) and fluid dynamics shows great potential to enhance the understanding of fluid dynamics, e.g., by predicting the resulting flow under different inlet conditions [5]. One common order reduction method is Proper Orthogonal Decomposition (POD) introduced to the fluid-dynamics community by Lumley in 1967 [6], which breaks the flow into different modes for a deeper analysis. Although POD techniques allow a reasonably good analysis of different modes in flow fields, its ability to handle non-linear relations, which are crucial aspects of turbulent flows, is limited [7]. Another relatively novel method for analyzing nonlinear systems is Dynamic Mode Decomposition (DMD) [8], which can be interpreted as an extension to POD with an additional consideration for the temporal aspects. However, DMD is also a linear approximation, presenting challenges when analyzing heavily nonlinear systems.

Artificial neural networks can overcome this obstacle due to their ability to capture non-linearity. In [9], the so-called Mode Decomposing Conventional Neural Network Autoencoder (MD-CNN-AE) is employed for mode decomposition. Although a conventional autoencoder uses only one mode, increasing the number of decoder branches is proven to improve learning and the capability to observe more complex flow structures. The superiority of MD-CNN-AE over POD is further shown in [10], where flow fields around square cylinders under varying geometric conditions are analyzed.

The current work extends the previous studies to the turbulent compressible flat plate flow, which has a high amount of non-linearity. That is, the effects of the surface

[†]This paper is part of the ParCFD 2024 Proceedings. A recording of the presentation is available on YouTube. The DOI of this document is 10.34734/FZJ-2025-02462 and of the Proceedings 10.34734/FZJ-2025-02175.

*Corresponding author
x.shao@aia.rwth-aachen.de (X. Shao);
oguzhan.ayan@rwth-aachen.de (H.O. Ayan);
f.huebenthal@aia.rwth-aachen.de (F. Hübenthal);
m.ruettgers@fz-juelich.de (M. Rüttgers); a.lintermann@fz-juelich.de (A. Lintermann); office@aia.rwth-aachen.de (W. Schröder)
ORCID(s): - (X. Shao); 0009-0007-5452-2786 (H.O. Ayan);
0009-0000-7159-8220 (F. Hübenthal); 0000-0003-3917-8407 (M. Rüttgers);
0000-0003-3321-6599 (A. Lintermann); 0000-0002-3472-1813 (W. Schröder)

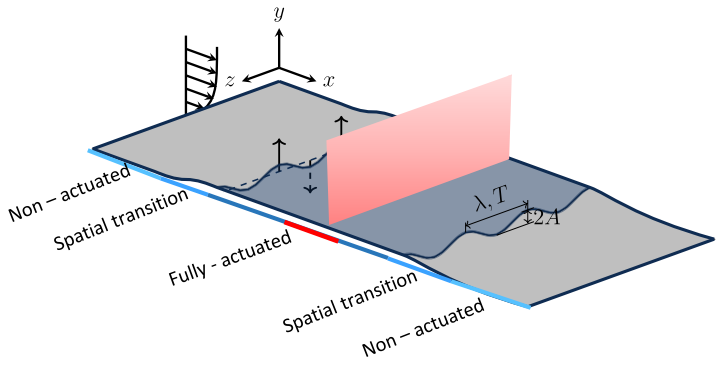

Figure 1: Schematic of the spanwise traveling transversal surface waves setup. The function $y|_{\text{wall}}(x,t) = g(x)A\cos\left(\frac{2\pi}{\lambda}(z-ct)\right)$ defines the wall-normal motion. The quantity A is the wave amplitude, λ is the wavelength, T is the period, c $(= \lambda/T)$ is the phase speed, and $g(x)$ are step functions to manage a gradual spatial implementation and decay of the actuation in the x direction.

actuation on the flow field are examined in detail including a comparison between POD, DMD, and the MD-CNN-AE.

2. Methods

Turbulence scale-resolving numerical simulations for actively controlled turbulent boundary layer flow are conducted by the finite volume (FV) module of the in-house multiphysics flow solver m-AIA (multiphysics Aerodynamics Institute Aachen)[1], formerly known as Zonal Flow Solver (ZFS) [11]. An advanced numerical methodology is utilized to precisely simulate the flow field over moving boundaries, incorporating high-order discretization techniques and effective shock-capturing schemes. More details on the numerical method are given in [12].

The flow domain is defined by the Cartesian coordinates x, y, and z representing the streamwise, wall-normal, and spanwise directions, illustrated in Fig. 1. The simulations are done for a momentum thickness based REYNOLDS number $\mathrm{Re}_\theta = \frac{u_\infty \theta}{\nu} = 1{,}000$, with a momentum thickness of $\theta = 1$ at location $x_1/\theta = 165$ and the freestream velocity u_∞, and $M = 0.7$. The domain size is defined by $L_x \times L_y \times L_z = 361\theta \times 101\theta \times 64.94\theta$. The actuation parameters are specified by wavelength λ^+, time period T^+, and amplitude A^+ in inner scaling based on the friction velocity u_τ and kinematic viscosity ν. The actuated parameter values are $\lambda^+ = 3{,}000$, $T^+ = 88$, and $A^+ = 74$.

As depicted in Fig. 2, the autoencoder follows a similar structure of [9], where the encoder is followed by two mode-decomposing decoder branches. The activation functions are selected to be linear and tanh or ReLU for the convolutional layers. The activation function is used in the intermediate convolutional layers to handle non-linearities, whereas the linear activation function is used at the final convolutional layer. The yz planes at the center of the domain $(x/\theta = 180)$, highlighted by the red rectangle in Fig. 1, are used for training. In total, 9230 snapshots are used and the model is trained for 2,000 epochs with the Adaptive Moment Estimation (Adam) [13] optimizer and a Mean Squared Error (MSE) loss function. The snapshots are randomly divided into training and validation sets and PyTorch's [14] distributed data parallel is utilized for distributed training on multiple GPUs. The parameters under investigation are the three-dimensional velocity components (u, v, w) and the pressure (p).

3. Results

The vortex topology based on the λ_2 criterion and the pressure contour in the yz plane is shown in Fig. 3. The isosurfaces of the vortex structures are generated by $\lambda_2 = -0.02$, colored by the relative MACH number defined by $M_{\text{rel}} = \sqrt{u^2 + v^2 + (c-w)^2}/\sqrt{\gamma p/\rho}$, where γ is the ratio of specific heats, and ρ is the density. At $c = 1.028$ relative to the stagnation sound speed a shock wave shown in Fig. 3(b) develops. The shock wave travels as the surface wave in the positive z direction. Within the shock wavefront, the vortices experience a reduction in number and size, influenced by the passing wave.

The reconstruction results for the autoencoder in comparison to the DMD and the POD with the 16 modes/16 decoders configuration for variables u, v, w, p are depicted in Fig. 4. The results indicate the high capability of the autoencoder relative to other methods, specifically in the three areas, namely in reconstructing the shock, the flow in the vicinity of the shock, and the turbulent structures. The inability of POD to regenerate the turbulent structures is an expected outcome as it heavily relies on linear spatial

[1] https://doi.org/10.5281/zenodo.13350586

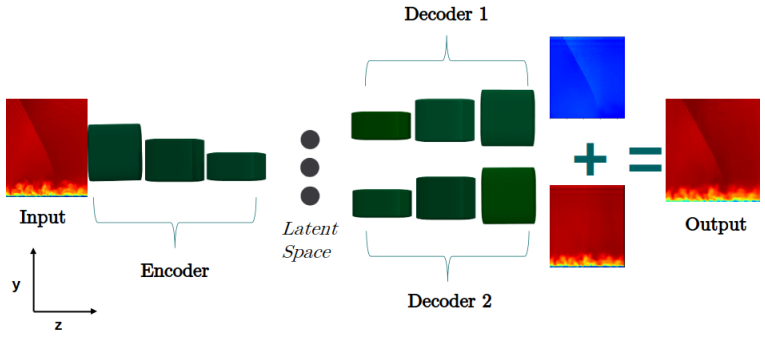

Figure 2: General concept the MD-CNN-AE for predicting two modes.

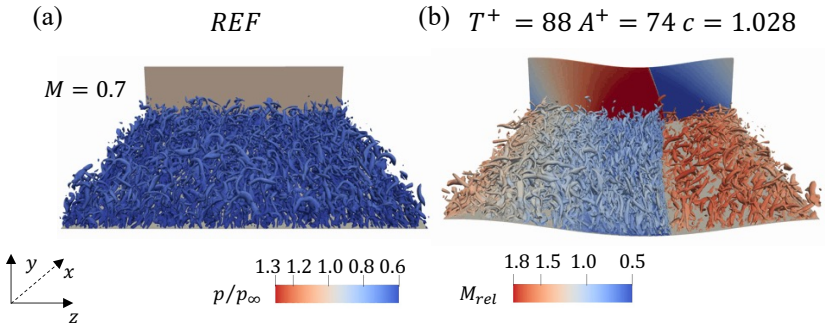

Figure 3: Instantaneous flow field for the non-actuated reference case (a), and the actuated case (b). Each illustration contains an yz plane pressure contour at $x/\theta = 180$ and a top view of the vortex topology in the streamwise direction and $y^+ < 60$.

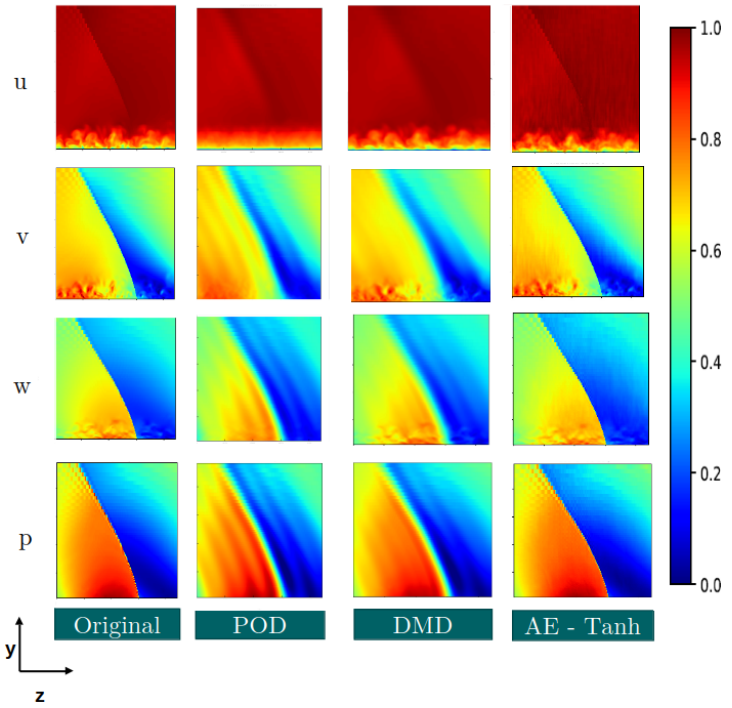

Figure 4: Original and reconstructed variables u, v, w, p for POD, DMD, and Autoencoder. The values are normalized by $(a_i - a_{min})/(a_{max} - a_{min})$, where a represents the variables under investigation.

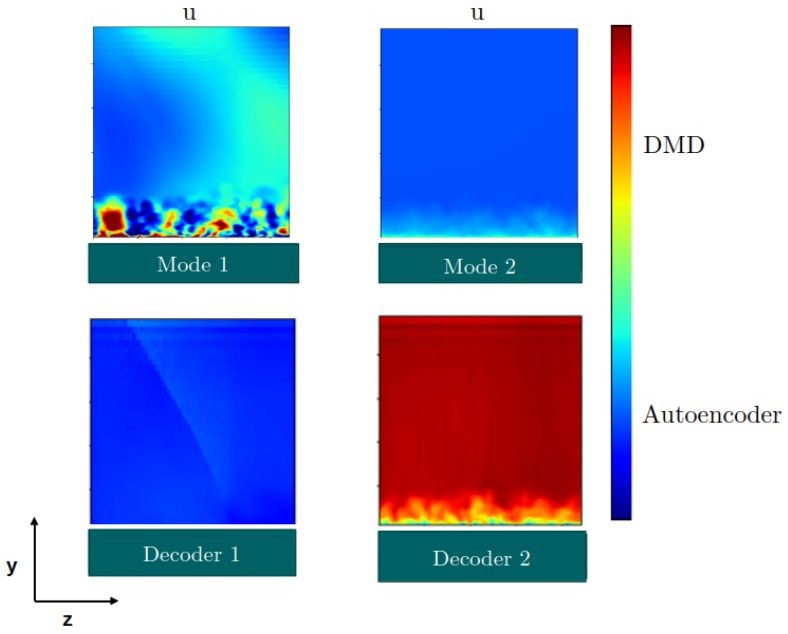

Figure 5: Decoder and the mode outputs of the channel u for the autoencoder and DMD for a 2 mode/decoder configuration. The values are normalized by $(a_i - a_{min})/(a_{max} - a_{min})$, where a represents the channels under investigation.

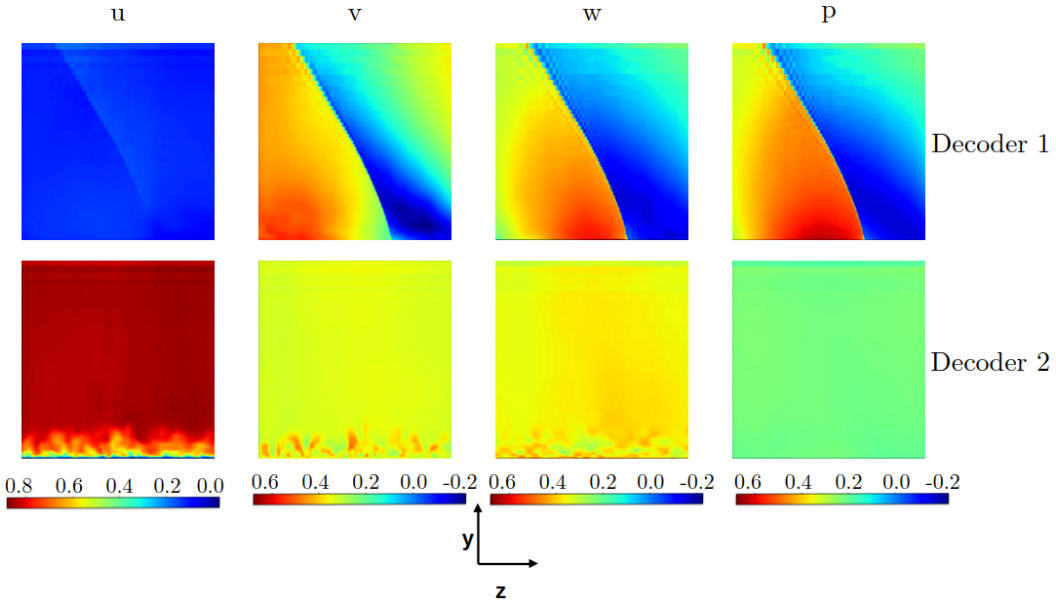

Figure 6: Decoder outputs of an autoencoder configuration with 2 decoders for all variables u, v, w, p. The values are normalized by $(a_i - a_{min})/(a_{max} - a_{min})$, where a represents the variable under investigation.

patterns, making it an unsuitable method for an in-depth analysis of turbulent flows. In contrast, DMD is able to reconstruct small-scale structures, which can be linked to its ability to consider the temporal dynamics of the system. However, DMD also fails to reconstruct the shock accurately as the shock wave is blurry, and the flow around the shock is also not well presented.

The individual outputs of the 2 decoders for the autoencoder and 2 modes for the DMD for variable u are shown in Fig. 5. It can be concluded that the autoencoder separates the shock and the turbulence in the flow, where the shock is

completely reconstructed. On the contrary, the DMD is not able to do this. The shock is vaguely represented and quite blurry in mode 1 in Fig. 5. The POD results here were not included as it was deemed an unfitting method, where the turbulence was not captured in detail. The ability to separate the large-scale shock waves and multi-scale turbulence structures is not unique to variable u. The same separation pattern exhibits in v, w, p.

The decoder results for all variables are shown in Fig. 6. In each velocity channel, decoder 1 captures the large-scale structures associated with the shock, whereas decoder 2 captures the multi-scale turbulence. On the other hand, only the large-scale structures are captured in the pressure channel, whereas decoder 2 does not output anything. This is affiliated with the nature of the pressure characteristics, which do not possess fluctuating turbulence.

In conclusion, the mode decomposing autoencoder can capture the shock and the high-frequency turbulent fluctuations accurately compared to the conventional methods, i.e., Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD). Moreover, the autoencoder can separate large-scale shock waves from multi-scale turbulence structures, which results in a continuous turbulence flow field. Clear decomposition makes it possible to examine the alter of turbulence structures due to the shock wave.

References

[1] P. Ricco, M. Skote, M. A. Leschziner, A review of turbulent skin-friction drag reduction by near-wall transverse forcing, Progress in Aerospace Sciences 123 (2021) 100713. doi:10.1016/j.paerosci.2021.100713.

[2] W. J. Jung, N. Mangiavacchi, R. Akhavan, Suppression of turbulence in wall-bounded flows by high-frequency spanwise oscillations, Physics of Fluids A: Fluid Dynamics 4 (8) (1992) 1605–1607. doi:10.1063/1.858381.

[3] M. Itoh, S. Tamano, K. Yokota, S. Taniguchi, Drag reduction in a turbulent boundary layer on a flexible sheet undergoing a spanwise traveling wave motion, Journal of Turbulence 7 (27) (2006) 27. doi:10.1080/14685240600647064.

[4] M. Walsh, Turbulent boundary layer drag reduction using riblets, in: 20th Aerospace Sciences Meeting, American Institute of Aeronautics and Astronautics, Reston, Virigina, 1982. doi:10.2514/6.1982-169.

[5] J. Ling, A. Kurzawski, J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, Journal of Fluid Mechanics 807 (2016) 155–166. doi:10.1017/jfm.2016.615.

[6] J. L. Lumley, The structure of inhomogeneous turbulent flows, Atmospheric Turbulence and Radio Wave Propagation (1967) 166–178.
URL https://cir.nii.ac.jp/crid/1574231874542771712

[7] S. L. Brunton, B. R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics, Annual Review of Fluid Mechanics 52 (2020) 477–508. doi:10.1146/annurev-fluid-010719-060214.

[8] P. J. Schmid, Dynamic mode decomposition of numerical and experimental data, Journal of fluid mechanics 656 (2010) 5–28. doi:10.1017/S0022112010001217.

[9] T. Murata, K. Fukami, K. Fukagata, Nonlinear mode decomposition with convolutional neural networks for fluid dynamics, Journal of Fluid Mechanics 882 (2020). doi:10.1017/jfm.2019.822.

[10] A. Higashida, K. Ando, M. Rüttgers, A. Lintermann, M. Tsubokura, Robustness evaluation of large-scale machine learning-based reduced order models for reproducing flow fields, Future Generation Computer Systems 159 (2024) 243–254. doi:10.1016/j.future.2024.05.005.

[11] A. Lintermann, M. Meinke, W. Schröder, Zonal Flow Solver (ZFS): a highly efficient multi-physics simulation framework, International Journal of Computational Fluid Dynamics 34 (7-8) (2020) 458–485. doi:10.1080/10618562.2020.1742328.

[12] M. Albers, P. S. Meysonnat, D. Fernex, R. Semaan, B. R. Noack, W. Schröder, Drag reduction and energy saving by spanwise traveling transversal surface waves for flat plate flow, Flow, Turbulence and Combustion 105 (1) (2020) 125–157. doi:10.1007/s10494-020-00110-8.

[13] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization (2017). arXiv:1412.6980.

[14] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An Imperative Style, High-Performance Deep Learning Library, in: Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.
URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf