The 8th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2022
5--9 June 2022, Oslo, Norway

NEURAL NETWORK PREDICTION OF THE FLOW FIELD IN A PERIODIC DOMAIN WITH HYPERNETWORK PARAMETRIZATION

HEIDLER V. [1], PECKA A. [1], BUBLÍK O. [1], AND VIMMR J. [1]

[1] Department of Mechanics, University of West Bohemia
Technická 8, Pilsen, Czech Republic,
e-mail: obublik@kme.zcu.cz

Key words: Compressible Fluid Flow, Convolution Neural Network, U-Net, Hypernetwork, Blade Cascade, Reynolds Parametrization

Abstract. This paper is concerned with fast flow field prediction in a blade cascade for variable blade shapes as well as variable Reynolds number using the machine-learning architecture called convolutional neural network. To generate flow field for a specific Reynolds number, an encoder-decoder convolutional neural network, also called U-Net, is used. The values 500, 1000 and 1500 of the Reynolds number are chosen as the training set. Three U-Nets were trained on CFD results for 100 blade profiles, each U-Net for a different Reynolds number. In order to get a prediction for variable Reynolds number, a so-called hypernetwork in employed. The hypernetwork essentially interpolates between the two trained U-Nets. The architecture of the hypernetwork is fully-connected feedforward neural network with one input neuron corresponding to the Reynolds number, one hidden layer and the output layer corresponds to the weights for the interpolated U-Net. The concept of the hypernetwork-based parametrization is tested on a problem of compressible fluid flow through a blade cascade with three unseen blade profiles and unseen Reynolds number.

1 INTRODUCTION

Convolution neural network (CNN) is a class of neural networks originally developed for recognition of patterns in images [2] and has become the dominant machine-learning technique for image and video processing in general. It is based on applying convolution filters on the image. After adding a deconvolutional layer, the architecture is also able to generate images. This encoder-decoder CNN architecture is also called a U-Net, the diagram of which is shown in Fig. 3.

Recently, CNNs have started to be used also to predict fluid flows. Convolution filters can be applied to flow fields the same way as they are applied to images. A discretized flow field is basically an image, where each pixel contains flow variables, such as velocity components, pressure and density, instead of the RGB code. The pioneer study on the subject of flow-field prediction using CNNs is [3]. In this paper, steady fluid flows were considered. The authors trained a CNN to predict velocity fields given an input image, which contains information about the fluid boundary. Soon after, the paper [4] concerned with the use of CNNs for the prediction of unsteady fluid flows appeared. In both papers, the U-Net architecture of CNN was used.

The above studies made a great progress in using machine learning in fluid dynamics, however, there is still a plenty of scope for improvement. The CNNs developed by [3] and [4] do not account for variable flow parameters, such as the Reynolds number or boundary conditions. To achieve flow parametrization, Zhang et al. [5] developed multiple CNNs that predict the lift coefficients of aerofoils with a variety of shapes, variable free-stream Mach numbers, Reynolds numbers and angles of attack. The limitation of the proposed model is that the output of these CNNs is only the lift coefficients of the aerofoil instead of the flow field around it. Bhatnagar et al. [6] and Chen at al. [7] developed CNN models that predict velocity and pressure fields around aerofoils with variable shape and with parametrized Reynolds number and angle of attack. Chen at al. used a more modern architecture for the neural network called a conditional generative adversarial network.

The aim of this study is to design a U-Net for the prediction of flow fields through a blade cascade. The blade cascade is modelled as a single interblade channel with periodic boundary conditions. Fast prediction of flow through turbine or compressor blades may have a significant impact on turbomachinery design. It is an important application that has its own hurdles. The high performance of a neural network yields a very promising alternative to classical CFD methods for complex problems such as flow control or shape optimization, in which case, a pressure field needs to be evaluated for many geometry variations.

To provide a parameter-dependent neural network model capable of predicting flow fields for various Reynolds numbers, a so-called hypernetwork is employed in this paper. Based on the given Reynolds numbers, the hypernetwork modifies weights of the main network in such a way that the main network produces flow fields which correspond to that Reynolds numbers. Hypernetworks can therefore be thought of as weight generators. In our model, the hypernetwork is a simple fully-connected feedforward neural network with one hidden layer, see Fig. 4. The neural network architecture was implemented in Python programming language using Keras [8] and TensorFlow [9] libraries.

2 GEOMETRY OF BLADE CASCADE

In this paper, the fluid flow is modelled in one interblade channel and periodic boundary condition is impose on the boundary with the neighbouring interblade channel. The computational domain is therefore formed by the space between two adjacent blades, which is covered by a non-Cartesian structured grid, see Fig. 2. Note that most of the other studies, such as [3, 4], use a Cartesian grid. The advantage of a non-Cartesian grid is that it accurately describes the boundary of the computational domain.

The blade profile is parametrized by two cubic splines as shown in Fig. 1. One spline describes the pressure side and the other spline describes the suction side. Each spline is defined by four points, the two inner points are variable and the two outer points are fixed. The splines share the two fixed points. One of the fixed points is placed on the leading edge at the position $[0, 0.5]$ and the other fixed point is placed on the trailing edge at the position $[1, 0]$. The blade is therefore described by six points - two points are fixed and four points called the design points are variable and therefore define the shape of the blade.

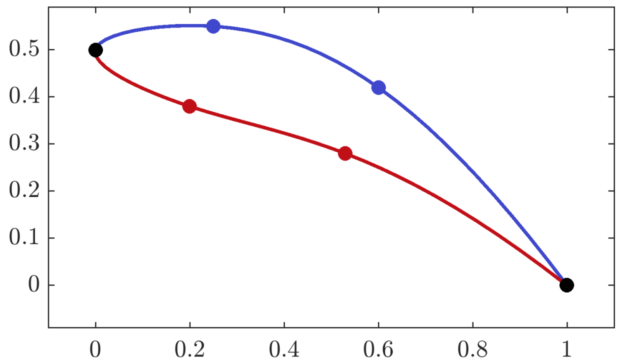

Figure 1: Blade profile.

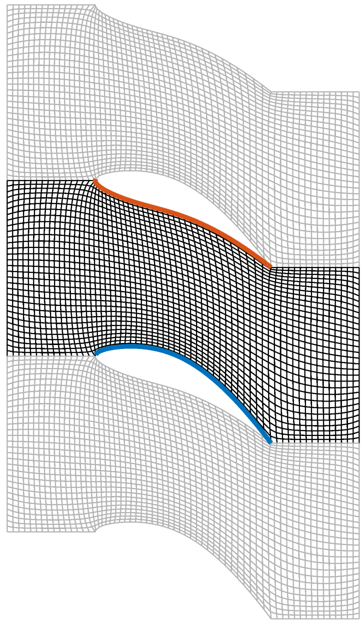
Figure 2: Computational mesh.

3 NEURAL NETWORKS ARCHITECTURE

For the prediction of flow fields with variable Reynolds number, two neural-network structures are used. One network structure, which is also called the main network, predicts the flow field for a given Reynolds number. A U-Net is used for this purpose, see Fig. 3. A U-Net is just an alternative name for an encoder-decoder CNN. The main network is actually trained for each of chosen Reynods numbers. As a result, a multiple U-Nets are obtained, each U-Net is trained to predict flow field with different Reynolds number.

Consequently, a so-called hypernetwork is employed to interpolate between the U-Nets. In this paper, the hypernetwork is a fully-connected feedforward neural network, with one input neuron, for the Reynolds number, and its output are weights for the interpolated main network.

The input to the U-Net are x and y coordinates of the structured non-Cartesian mesh with 64×32 points. An example of the input mesh is highlighted in Fig. 2. Each U-Net generates a flow field consisting of density, pressure and velocity components as the output. Each U-Net was trained on a set of 100 blade profiles and for each of the Reynolds numbers. The resulting weights for each of the U-Net were then used to train the hypernetwork.

The hypernetwork consists of a dense neural network, see Fig. 4. At the input, the Reynolds number is supplied. The weights of the U-Net are then generated at the output of the hypernetwork.

3.1 BOUNDARY CONDITIONS

It is not strictly necessary to include the boundary conditions in the CNN model, since the neural network would pick up the boundary conditions from the training dataset. However, one

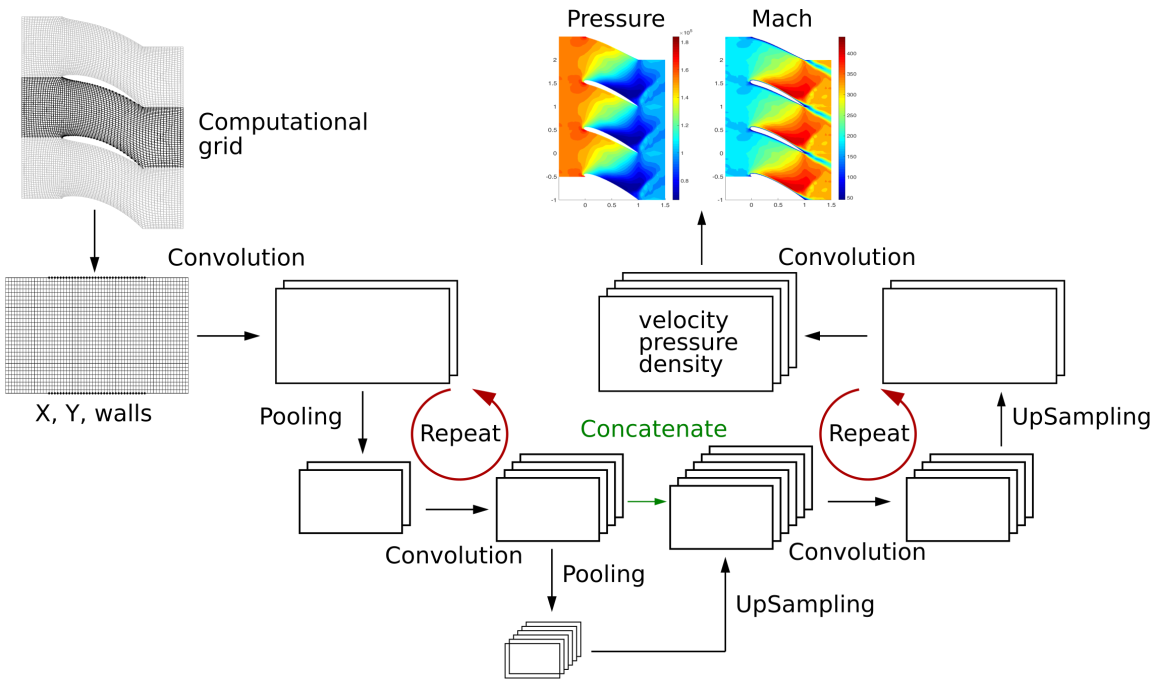

Figure 3: Main network architecture - U-Net.

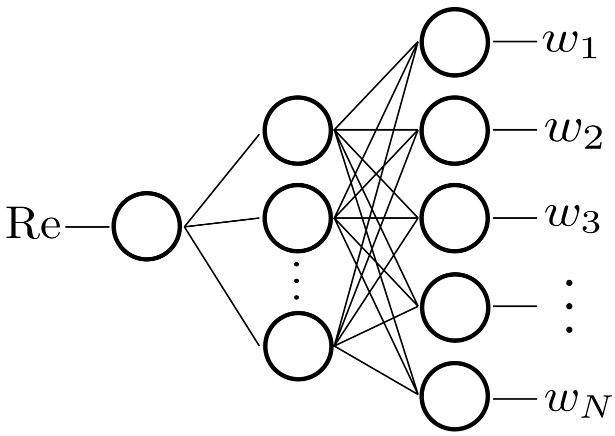

Figure 4: Hypernetwork architecture.

obtains better results if the boundary conditions are included in the model. For this reason, two types of padding (replication and periodic) are used according to the type of boundary condition. The algorithm is illustrated in Fig. 5. Before each convolution layer, the left column (the inlet) is copied into a new column on the left of the domain and likewise the right column (the outlet) into a new column on the right of the domain. Secondly, the top row is copied into a new row at the bottom and simultaneously copy the bottom row into a new row on the top.

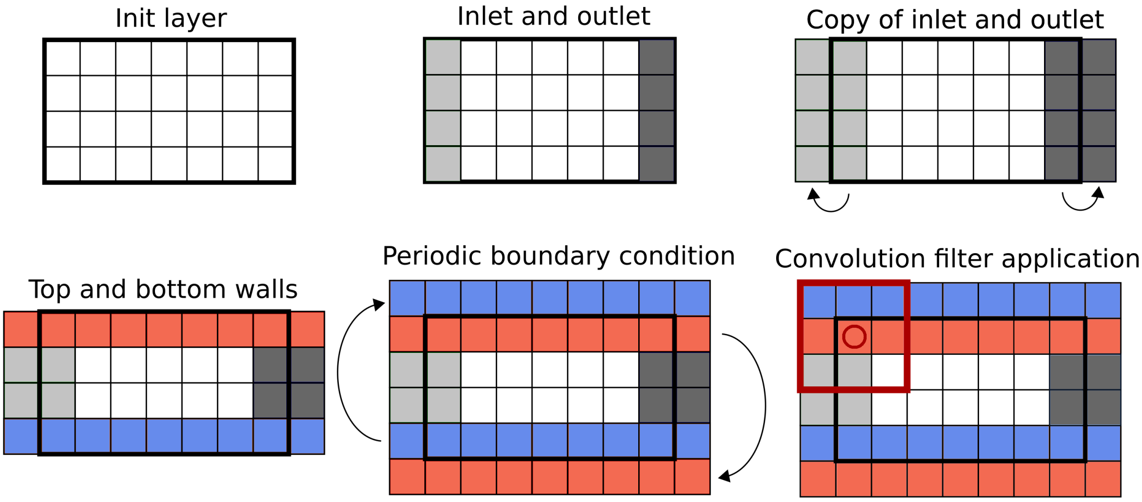

Figure 5: Illustration of the padding operation.

4 RESULTS

To test the described concept of the suggested flow field parametrization, a training set of 100 random profiles was generated in total. Furthermore, three values of the Reynolds number, namely Re = 500, 1000, 1500, were chosen for training of the U-Net. For each profile and each of the three values of the Reynolds number, a flow field was computed using the open-source CFD software FlowPro [1]. In these simulations, air in standard conditions is considered as the fluid. In all the forthcoming simulations, the angle of attack, $\alpha = 15°$, and the ratio of the outlet static and inlet stagnation pressure $p_{\text{out}}/p_{\text{in}_0} = 0.843$ is used. This correspond to the subsonic flow with the far-field Mach number $\text{Ma}_\infty = 0.5$.

Three U-Nets were trained on CFD results of flow through a blade cascade with the 100 blade profiles. Each U-Net was trained for a different value of the Reynolds number. Finally, a hypernetwork was trained on the three sets of weights corresponding to the three U-Nets. Once the U-Nets and hypernetwork are successfully trained on the training dataset, the prediction procedure is as follows. Firstly, the weights for the main network are generated for the chosen Reynolds number using the hypernetwork. Secondly, the resulting main network is used for fast flow field predictions in the cascade with unseen blade profiles.

In order to validate the developed concept against the results of the CFD simulation, the pressure lift L and pressure drag D, which are given by the following relations

$$L = \oint_\Gamma p\, n_y \, dS, \qquad\qquad D = \oint_\Gamma p\, n_x \, dS, \tag{1}$$

are compared. Here p is pressure, $\boldsymbol{n} = [n_x, n_y]$ is the outer unit normal vector to the surface Γ of the blade profile.

Three testing blade profiles were randomly generated, see Fig. 6. These profiles were not included in the training set. Three CFD simulations through blade cascades with the testing blade profiles for the Reynolds number 750 were then performed to obtain the testing dataset. Note that the value 750 of the Reynolds number is also not included in the training dataset. A

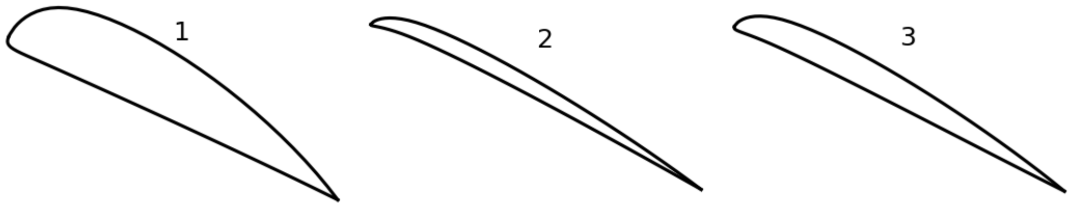

Figure 6: Testing blade profiles.

Table 1: Comparison of calculated and predicted results by evaluation of pressure lift and pressure drag for the Reynolds number 750 on the testing blade profiles.

Blade	Drag			Lift		
	CFD	DNN	Err [%]	CFD	DNN	Err [%]
1	4.533×10^{-2}	4.418×10^{-2}	2.5	5.641×10^{-2}	5.912×10^{-2}	4.8
2	3.556×10^{-2}	3.578×10^{-2}	0.6	6.923×10^{-2}	7.111×10^{-2}	2.7
3	3.710×10^{-2}	3.552×10^{-2}	4.3	6.621×10^{-2}	6.682×10^{-2}	0.9

comparison of the predicted and calculated velocity and pressure fields in cascades with each of the testing blade profile is shown in Figs 7 – 9. Moreover, the pressure around the blade profile is also plotted for a better comparison of the calculated and predicted results. Table 1 shows the error of the predicted flow fields with the Reynolds number 750 for unseen testing blade profiles.

5 CONCLUSIONS

Within this study, the concept of flow field parametrization using a hypernetwork was introduced. The hypernetwork was trained for three sets of U-net weights corresponding to three values of the Reynolds numbers, namely to 500, 1000 and 1500. The U-net with weights generated by the hypernetwork for the Reynolds number 750 was evaluated for three test profiles. The lift and drag errors were evaluated against the CFD calculation. All of the average errors are less than six percent. The results indicate that the concept of the flow field parametrization by the hypernetwork could be a very promising alternative to classical CFD methods for tough problems such as flow control or shape optimization. The future studies will be focused on a fully parametrized flow field.

6 ACKNOLEDGEMENTS

This research is supported by project ”Fast flow-field prediction using deep neural networks for solving fluid-structure interaction problems” GA21-31457S of the Grant Agency of the Czech Republic.

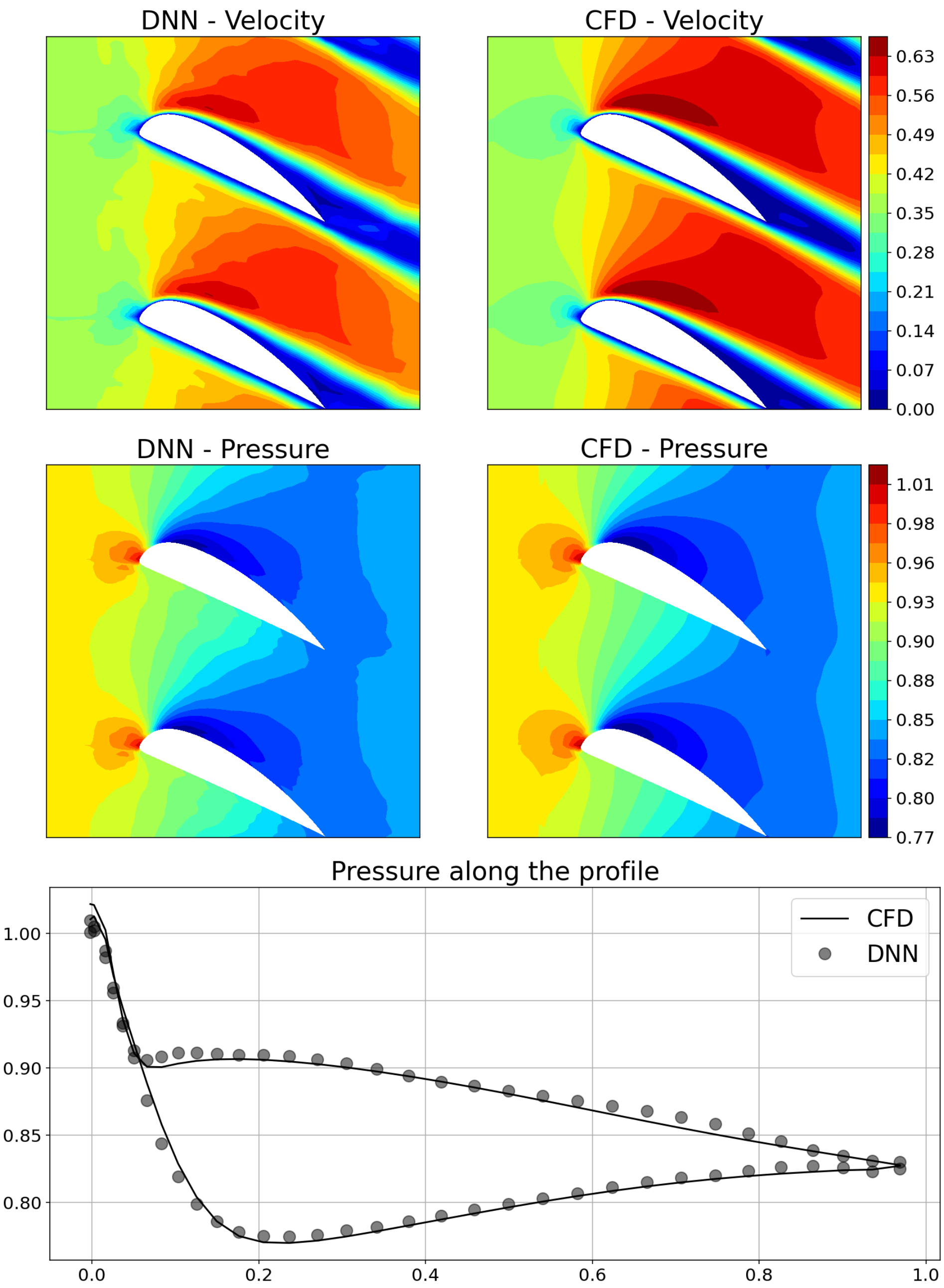

Figure 7: Results for blade profile number 1 and Re = 750.

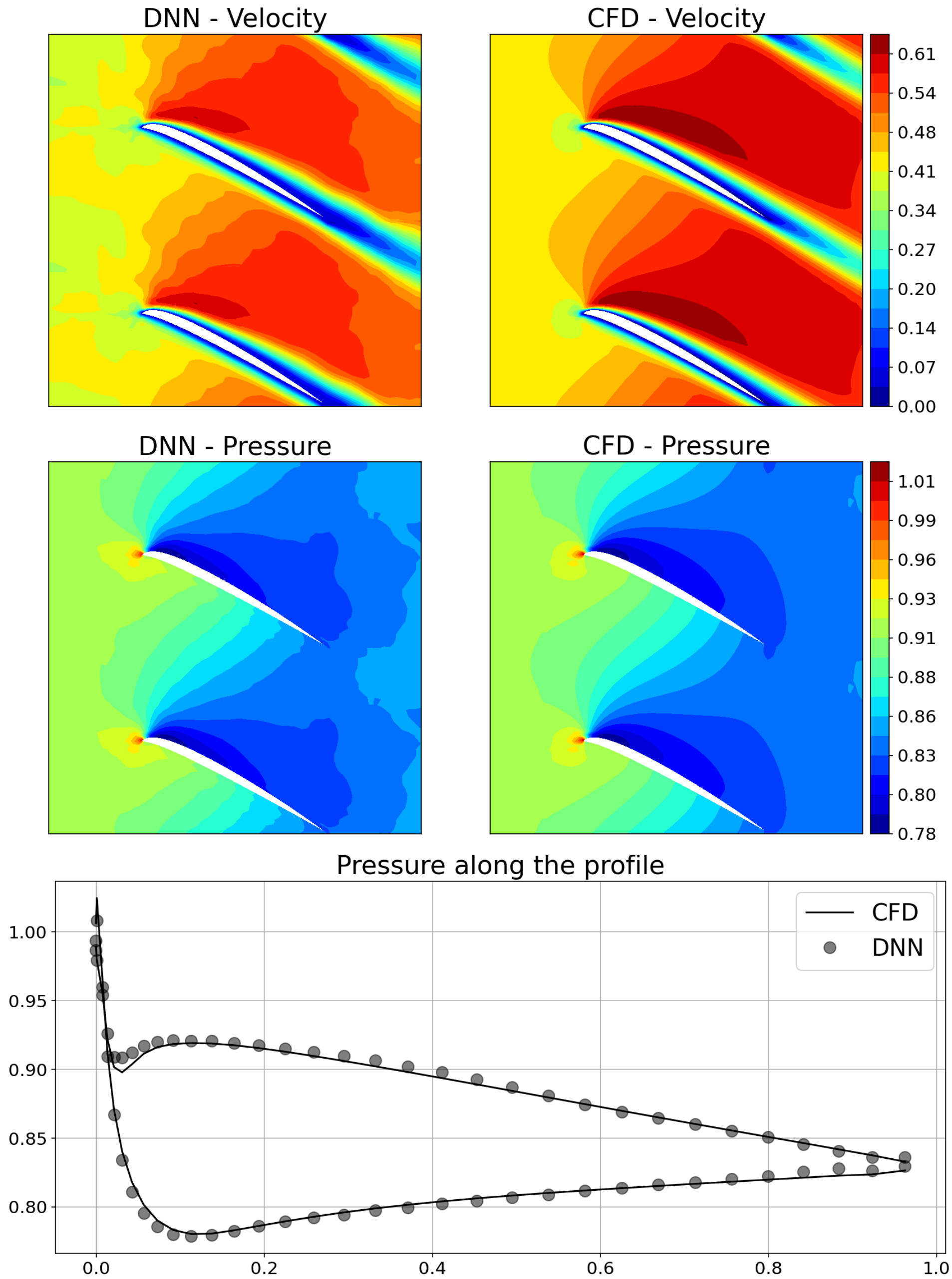

Figure 8: Results for blade profile number 2 and Re = 750.

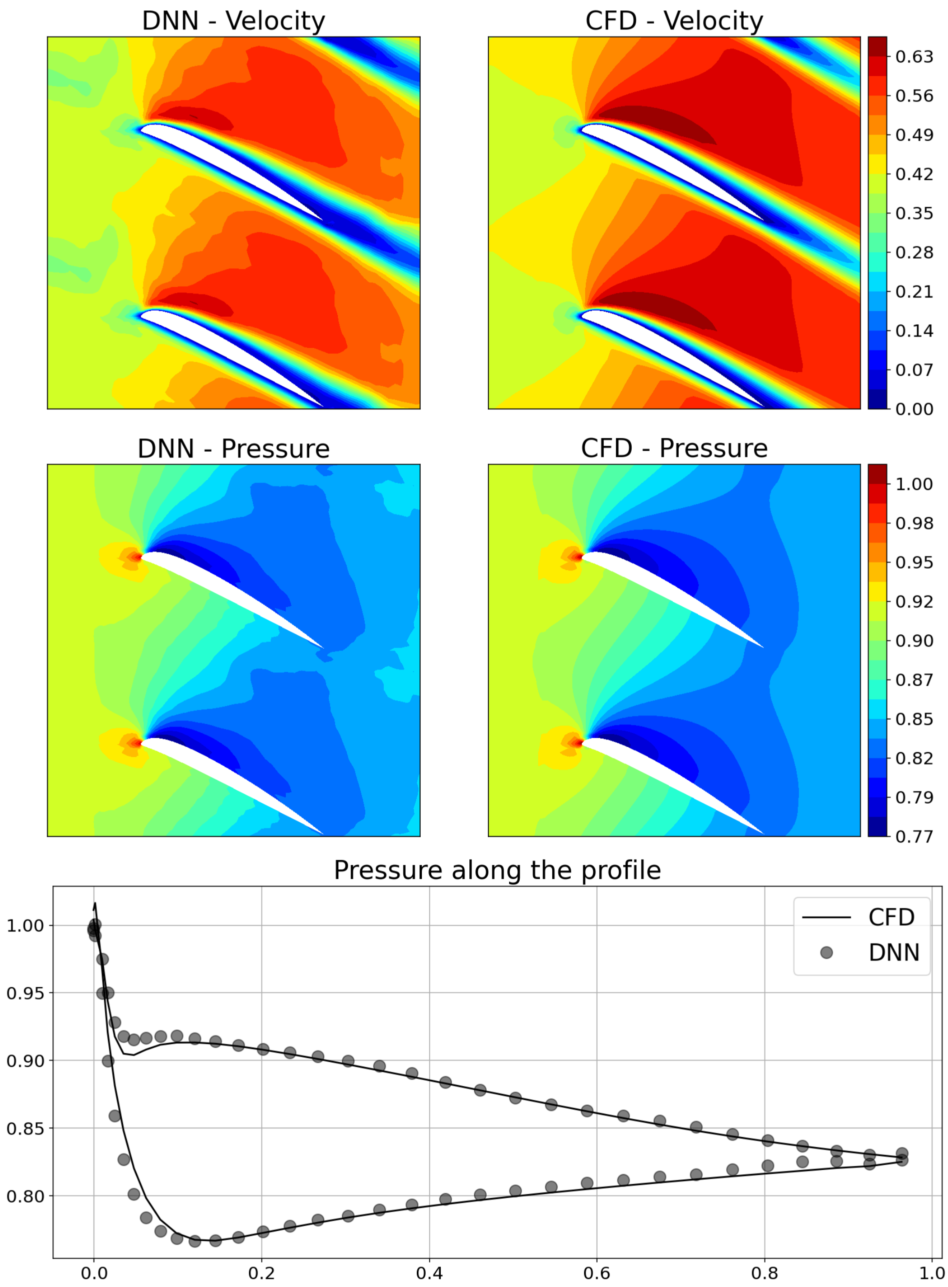

Figure 9: Results for blade profile number 3 and Re = 750.

REFERENCES

[1] Bublík, O. and Pecka, A. and Vimmr, J., FlowPro - multipurpose CFD software written in Java, *Proc. of Computational Mechanics*, University of West Bohemia (2000).

[2] Fukushima, K., Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position, Biological Cybernetics (1980) **36 (4)**: 193–202.

[3] Guo, X., Li, W., Iorio, F., Convolutional neural networks for steady flow approximation, *Proc. the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (2016), pp. 481.

[4] Hennigh, O., Lat-Net: Compressing Lattice Boltzmann Flow Simulations using Deep Neural Networks (2017).

[5] Zhang, Y., Sung, W. J., Mavris D. N., Application of convolutional neural network to predict airfoil lift coefficient, AIAA/ASCE/AHS/ASC, Structures, Structural Dynamics, and Materials Conference, American Institute of Aeronautics and Astronautics, Kissimmee, Florida (2018).

[6] Bhatnagar, S., Afshar, Y., Pan S., Duraisamy K., Kaushik S., Prediction of aerodynamic flow fields using convolutional neural networks, Computational Mechanics (2019) **64 (2)**: 525–545.

[7] Chen, D., Gao X., Xu C., Chen S., Fang J., Wang Z., Wang Z., Flowgan: A conditional generative adversarial network for flow prediction in various conditions, 2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI) (2020), pp. 315–322.

[8] Chollet, F. and others, Keras (2015), https://keras.io.

[9] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, Ch., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X. (2015), TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.