POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

A Deep Learning Framework to Infer Functional and Spatial Properties from CFD

LAUREA MAGISTRALE IN COMPUTER SCIENCE AND ENGINEERING - INGEGNERIA INFORMATICA

Author: FRANCESCO MONTANARO

Advisor: PROF. GIACOMO BORACCHI

Co-advisor: ANDREA SCHILLACI PHD

Academic year: 2021-2022

1. Introduction

Computational Fluid Dynamics (CFD), i.e., solving the differential equations of the fluid motion using numerical methods, plays a crucial role in a large number of applications, ranging from health to industry. However, the ultimate objective of CFD analysis remains often elusive: the final information relevant to the end user may not be provided directly by the CFD itself or expressed as an analytical function of the CFD solution.

In this work, we present a data-driven framework to extract these additional information from CFD solutions. For instance, these may include details for the diagnosis of nasal breathing difficulties or the presence of anomalies in an industrial product. This is a challenging problem because of the huge dimensionality of CFD outcomes, and the limited data that can be typically gathered. By pursuing a Deep Learning (DL) pipeline of pre-processing, dimensionality reduction and model training, we demonstrate that relevant patterns can be learned from CFD data to obtain accurate predictions. Several experiments support our claim that the flow field and its convective properties can be exploited to retrieve useful information that do not admit an analytical definition with respect to the input domain. By examining different examples concerning the airflow around wing sections, as shown in Figure 1, we show that even in CFD-related problems, the extraction of effective and meaningful information can be demanded to the algorithm itself, leading to very good predictive results. Despite of the relative simplicity of both the geometrical characteristics and turbulence models, we demonstrate that the combination of Deep Learning and CFD can model a system in terms of high-level functional and spatial properties.

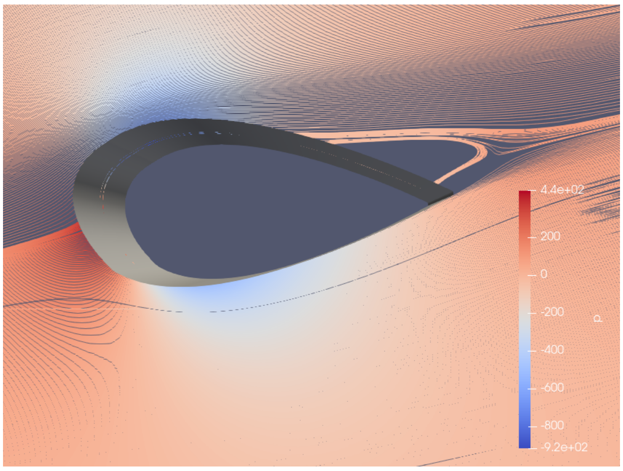

Figure 1: CFD solution of the airflow around a *NACA 1145* airfoil.

2. Related Work

In the last years, applications of Artificial Intelligence in fluid mechanics have grown significantly. Most of the researches focus on the development of data driven models for solving the differential equations of fluid dynamics, using CFD as input, and obtaining fluid mechanical quantities as output. In fact, with the availability of large and diverse data sets, researchers have explored methods to systematically inform turbulence models with data, with the goal of quantifying and reducing model uncertainties [1].

There are numerous examples of custom neural network architectures being used to enforce physical solutions for applications in fluid mechanics. Among the several experiments, Ling et al. [2] were the first to employ Deep Learning techniques to enforce a correction to the popular Spalart-Allmaras RANS turbulence model by embedding the required Galilean invariance into the model-predicted tensor of the turbulent stresses.

To the best of our knowledge, the research aimed at inferring quantities that cannot be directly computed from CFD is still in its early stages. This research area can certainly be pursued given the many applications in which it can range: from the medical to the industrial one. A considerable contribution to this problem was given by Schillaci et al. [3] that inspired and contributed to this thesis project.

3. Problem formulation

The output of a CFD simulation is a set of scalar or vector fields defined over a domain $\Omega \subset \mathbb{R}^3$, which is discretized into many small volumes or a computational mesh. These fields are obtained by solving the discretized Navier-Stokes equations together with boundary conditions applied at the geometrical boundary $\Gamma \subset \mathbb{R}^3$. For instance, Γ includes the geometry of a phisical body affecting the free stream, such as an airfoil (Figure 1).

A CFD simulation results in several output fields, which in general are also time-dependent. However, this work will only consider time-averaged quantites, in particular the vector field of the mean velocity ($\vec{U}$) and the scalar field of the mean pressure (p), both expressed in a Cartesian reference system:

$$\vec{U}(x,y,z) = \begin{bmatrix} u(x,y,z) \\ v(x,y,z) \\ w(x,y,z) \end{bmatrix}, \qquad p(x,y,z) \tag{1}$$

Given that the computational mesh is a representation of a larger geometric domain by smaller discrete cells, all the flow quantites referring to the generic $i-th$ cell as well as the spatial coordinates of its vertices, can be stacked into vectors $Q_i \in \mathbb{R}^4$ and $V_i \in \mathbb{R}^{j\times 3}$:

$$Q_i = \begin{bmatrix} u_i \\ v_i \\ w_i \\ p_i \end{bmatrix}, \qquad V_i = \begin{bmatrix} (x_1,y_1,z_1) \\ (x_2,y_2,z_2) \\ \cdots \\ (x_j,y_j,z_j) \end{bmatrix} \tag{2}$$

where $u_i = u(x_i, y_i, z_i)$ refers to the flow quantity at the cell's center, while $(x_j, y_j, z_j) \in \Omega$ are the spatial coordinates of its $j-th$ vertex.

Since the spatial domain Ω is discretized into n cells, which in our elementary case is about $n \sim 10^6$, the CFD output is a very large matrix:

$$M \subset \mathbb{R}^{(4+j\times 3)\times n} \tag{3}$$

Our goal is to build a model Φ that predicts a target value Y associated to the input matrix M provided by CFD:

$$\Phi : M \mapsto Y \tag{4}$$

The target variable Y can be either categorical (as for a classifier that indentifies the presence of anomalies in an industrial product) or real (as for a regressor that estimates some geometrical quantities from Γ or some spatial properties from Ω). In general, we should assume to deal with problems where training samples are scarce, as for medical applications, where the retrieval of diversified and reliable data is quite challenging. Therefore, the major constraints to address in our setting are: i) The large dimensionality of the input data (namely n). ii) The limited number of training samples (namely l).

4. Proposed solution

We propose our solution to the problem as a sequence of steps of a classical Deep Learning pipeline, namely *pre-processing*, *dimensionality reduction* and *model training*.

Pre-processing

As a first pre-processing step, we compute the velocity magnitude of each cell $|U_i|$ and normalize it by the free stream's velocity magnitude $|U_f|$ of the CFD simulation.

$$|U_i| = \frac{1}{|U_f|}\sqrt{u_i^2 + v_i^2 + w_i^2} \tag{5}$$

By applying (5), vectors in (2) can be replaced by:

$$Q_i = \begin{bmatrix} |U_i| \\ p_i \end{bmatrix}, \qquad V_i = \begin{bmatrix} (x_1,y_1,z_1) \\ (x_2,y_2,z_2) \\ \cdots \\ (x_j,y_j,z_j) \end{bmatrix} \tag{6}$$

A sensing plane S is now sampled from the input domain Ω, $S \subset \Omega$, at a predefined spatial location (x_s, y_s, z_s) and whose direction is given by its normal versor $\vec{n}_s = x\vec{i} + y\vec{j} + z\vec{k}$.

Dimensionality reduction

Although the sensing plane represents a subset of the input domain $S \subset \Omega$, its dimension m could range from a few hundred to even several thousands cells. This high dimensionality allows more information to be stored, but practically increases the possibility of noise and redundancy in real world data, making models prone to overfitting. To address the problem, we derive an equivalent representation of the geometrical domain that is both compact and meaningful. Starting from S, our goal is to perform a series of geometrical transformations to reduce its dimensionality as well as to standardize the input data:

$$\{c_i \mid i = 1, \dots, m\} \mapsto \{b_i \mid i = 1, \dots, r\} \tag{7}$$

where m is the number of cells from the original mesh, while r is the decreased dimension of the new input domain, being $r < m$. To reduce the dimensionality of S, a ***binning operation*** is performed by grouping a large number of cells into a smaller number of equally spaced and fixed-sized surfaces, i.e. the *bins* $\{b_i \mid i = 1, \dots, r\}$. This procedure can be performed by adopting two different approaches:

1. *One-dimensional binning*: the sensing plane S is discretized along one of the axes of the Cartesian plane, resulting into a one-dimensional signal representing the spatial profile of the flow quantities.
2. *Two-dimensional binning*: the sensing plane S is discretized along both the axes of the Cartesian plane, resulting into a single-channel image depicting the distribution of the flow quantities across the space.

Figure 2 shows an example of both a one-dimensional and two-dimensional binning operation performed on the generic sensing plane S. It is an application choice that between one type of discretization rather than another. For example, a one-dimensional binning compresses all the information spread within the sensing plane S along a single spatial direction. So, it will be more suitable in simpler applications and in general for all those where the variance of the flow quantities is almost zero along a specific axis.

The discretization factor r corresponds to its effective ***spatial resolution***. Using a larger resolution allows to represent the evolution of flow quantities across different regions more accurately; on the other hand, it increases the dimensionality of the input data as well as the computational cost. Resolution r is a key parameter that should be properly chosen based on the problem's complexity. For instance, domains characterized by complex vortical structures, recirculation zones, and high variability of the flow quantities, need higher resolutions to accurately represent the evolution of fluid dynamic phenomena as well as to avoid that relevant information are attenuated by the discretization process. Figure 3 shows different spatial resolutions of the same sensing plane S discretized by using a two-dimensional binning procedure.

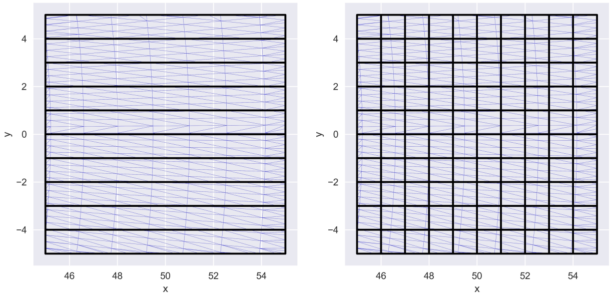

Figure 2: One-dimensional and two-dimensional binning operation of a generic sensing plane S.

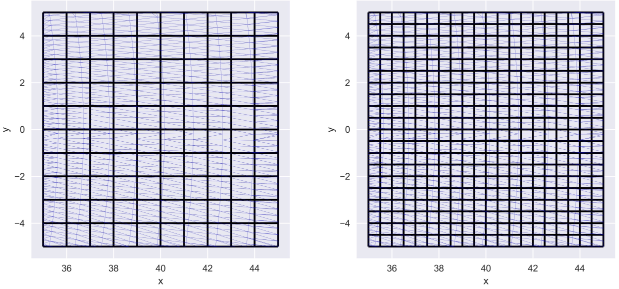

Figure 3: Different spatial resolutions for the same sensing plane S, discretized by using a two-dimensional binning operation.

The cells of the sensing plane S are now segmented to fit each bin of the new discretized domain $\{b_i \mid i = 1, \dots, r\}$, as shown in Figure 4. This process consists of splitting cells as to make the geometrical representation to remain consistent across different regions.

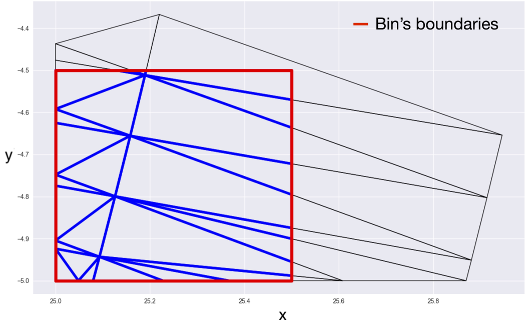

Figure 4: Cells of the sensing plane S segmented to properly fit the $i-th$ bin b_i.

The information associated to each bin is now extracted from the cells themselves. We compute the average of the cells' flow quantities Q_i, weighted for their corresponding surfaces. This is to ensure bigger cells to have a major contribution on the final outcome than the smaller ones:

$$\bar{p_i} = \frac{\sum_{j=1}^{n_i} A_j \cdot p_j}{\sum_{j=1}^{n_i} A_j} \tag{8}$$

$$|\bar{U_i}| = \frac{\sum_{j=1}^{n_i} A_j \cdot |U_j|}{\sum_{j=1}^{n_i} A_j} \tag{9}$$

For each bin, $\bar{p_i}$ and $|\bar{U_i}|$ represent respectively the averaged value of the pressure and the velocity magnitude, n_i is the number of cells enclosed whithin its boundaries while A_j is the surface of the $j-th$ cell.

Model training
The pre-processing and dimensionality reduction steps map the CFD's output matrix $M \subset \mathbb{R}^{(4+j\times 3)\times n}$ to a compact representation $S \subset \mathbb{R}^{r}$, which combines r bins being the results of either a one-dimensional or a two-dimensional discretization. Depending on the nature of the target variables, any classifier or regressor Φ can be trained from the set of labeled input data $\{(S_j, Y_j), j = 1, \dots, l\}$. In the experiments described below, we train a Convolutional Neural Network (CNN) to perform regression over the space of target variables to show that a limited number of data is often enough to provide very accurate predictions.

5. Experiments

We describe two experimental setups to show the flexibility and effectiveness of the proposed solution for problems of different nature.

1. Prediction of geometrical properties of an airfoil

a) Dataset and task: We consider a popular family of airfoils four-digits NACA (National Advisory Committee for Aeronautics). The shape of a NACA airfoil is described by their four-digits code, which corresponds to three integer numbers, and the length of the chord c. The first number (I) corresponds to the first digit (integer, $[0-9]$) and quantifies the maximum camber of the airfoil in units of $c/100$; the second number (II) corresponds to the second digit (integer, $[0-9]$) and locates the point of maximum camber along the chord measured from the leading edge, expressed in $c/10$; the third number (III) has two digits (integer, $[05-50]$) and quantifies the maximum thickness expressed in $c/100$. Our dataset is composed of *3025* CFD solutions, obtained from the corresponding numerical simulations by solving the Reynolds-Averaged Navier–Stokes equations (RANS). The goal is to train a regressor Φ (4) to predict the shape of the airfoil by means of the three numbers of its four-digits NACA code.

$$\Phi : S \mapsto (I, II, III) \tag{10}$$

b) Extraction of Non-Expert-Driven features: Three sensing planes S_{1-3} are extracted from the input space Ω. Each of this is orthogonal to the airfoil chord, being $\vec{n}_{S_{1-3}} = \vec{i}$, and has a vertical length of $256c$. The first plane S_1 lies at $x = -c$ upstream the airfoil, the second one S_2 at $x = 2c$ downstream the airfoil while the third one S_3 at $x = 10c$ downstream the airfoil. Figure 5 shows an high-level view of the three sensing planes S_{1-3} extracted at different spatial sections. Since the generic airfoil has a constant shape along its horizontal length, being the one spreading along the z-axis, we expect the variance of the flow quantities on the z-axis itself to be almost equal to zero. Therefore, a one-dimensional binning performed on the vertical direction y, has enough information to properly represent the evolution of fluid dynamic phenomena as a function of the airfoil itself. This result in spatial signals representing the profile of the flow quantities along the y axis at the specified x section. Figure 6 shows the spatial profiles of the *pressure* field of resolution $r = 256$, extracted from planes S_{1-3} for a *NACA 0005* airfoil.

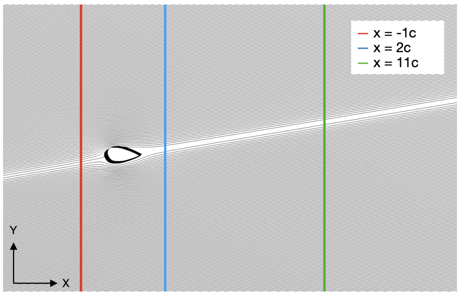

Figure 5: High-level view of the sensing planes S_{1-3} orthogonal to the airfoil chord ($\vec{n}_{S_{1-3}} = \vec{i}$).

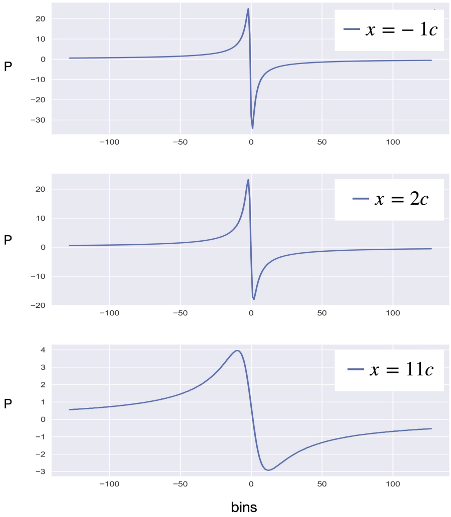

Figure 6: Spatial profiles of the *pressure* field of resolution $r = 256$, extracted from planes S_{1-3} for a *NACA 0005* airfoil.

c) Model training and performance assessment: Since the input data is traceable to an actual spatial signal, we train a 1D Convolutional Neural Network (CNN-1D) to perform regression of the target variables. Being the estimated labels real numbers, they are rounded to the closest integer to obtain the corresponding NACA number, therefore the problem is also evaluated in terms of classification accuracy.
i) *K-fold cross validation*: A 5-fold cross validation technique is performed to obtain a reliable evaluation of the model's performance. Given the whole dataset of $l = 3025$ observations, the splitting criterion is of

$S_{r=256}$	I		II		III		$\|e\|$	$a[\%]$
	$\|e\|$	$a[\%]$	$\|e\|$	$a[\%]$	$\|e\|$	$a[\%]$		
$S_{1,p}$	0,13	98,35	0,21	92,23	0,27	96,10	0,20	95,56
$S_{1,\|U\|}$	0,27	86,27	0,45	70,79	0,38	85,64	0,36	80,90
$S_{2,p}$	0,08	99,31	0,11	98,86	0,09	99,12	0,09	99,10
$S_{2,\|U\|}$	0,16	96,82	0,26	87,50	0,20	94,18	0,21	92,83
$S_{3,p}$	0,21	92,62	0,33	79,09	0,25	87,63	0,27	86,45
$S_{3,\|U\|}$	0,22	91,6	0,49	64,81	0,33	79,2	0,35	78,54

Table 1: 5-fold cross validation results on the inference of the geometrical properties of an airfoil.

using *80% (2420 samples)* of the available samples for the *training* set and the remaining *20% (605 samples)* for the *test* set. Table 1 shows the results obtained on the test set in terms of the regression Mean Absolute Error $|e|$ and the classification Accuracy $a[\%]$ from spatial profiles of both the *pressure* (p) and *velocity* ($|U|$) fields with resolution $r = 256$. When profiles of the pressure field are used, the model achieves very small absolute errors and high classification accuracy, between *86%* and *99,5%*. As expected, some vertical sections are more informative. This result is not surprising as the distribution of the flow quantities tends to become uniform as the distance from the body perturbing the free stream increases, resulting in smoother but less informative profiles, as shown in Figure 6 for section $x = 11c$.

ii) *Interpolation*: We study the evolution of the performance in a context of data scarcity. Based on the previous results, we restrict our study to spatial profiles of the *pressure* field to assess how the performance vary as a function of the training set's size. Starting from an initial size of *100* observations, the experiments are repeated n times by increasing, at each iteration, the number of training samples of *100* units. As shown in Figure 7, for the most informative section, i.e. $x = 2c$, about *700* training samples are sufficient to obtain an accuracy $a[\%] > 90\%$.

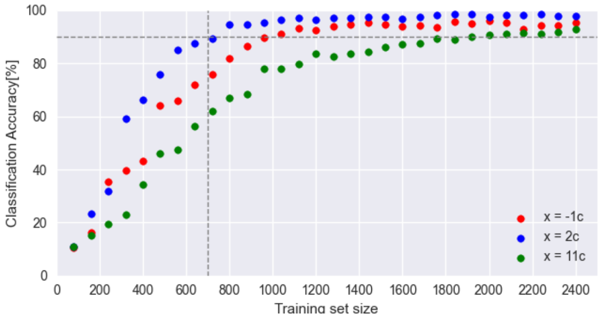

Figure 7: Classification Accuracy $a[\%]$ as a function of the training set size.

2. Prediction of spatial properties of a geometric domain

a) Dataset and task: Our dataset consists of l CFD simulations corresponding to domains $\{\Omega_i, i = 1, \ldots, l\}$ and geometries $\{\Gamma_i, i = 1, \ldots, l\}$, i.e. different NACA airfoils. Even if the available simulations are limited, by placing a sensing plane S at a generic location in the input space, a large number of training samples can potentially be computed. Each sensing plane S is a square of side $10c$ whose direction is parallel to the stream, being its normal versor $\vec{n}_S = \vec{k}$. Figure 8 shows an high-level view of the generic sensing plane S parallel to the stream.

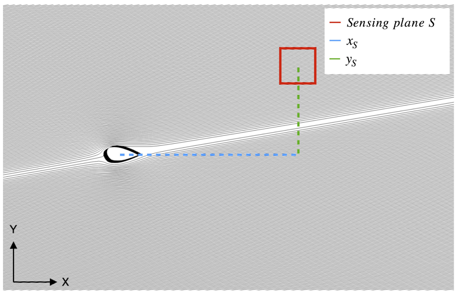

Figure 8: High-level view of a sensing plane S parallel to the stream ($\vec{n}_S = \vec{k}$).

Our goal is to train a regressor Φ (4) to predict the relative position of the airfoil Γ by means of its horizontal x_s and vertical y_s distance from S:

$$\Phi : S \mapsto (x_s, y_s) \tag{11}$$

b) Extraction of Non-Expert-Driven features: The evolution of flow quantities cannot be considered constant over any spatial direction. Therefore, we adopt a two-dimensional discretization. This results in single-channel images I_{2D} depicting the distribution of flow quantities in the space. However, we consider a one-dimensional discretization too, resulting in spatial profiles of the flow fields S_{1D}. Figure 9 shows both a one-dimensional and two-dimensional discretization of the pressure field for the same sensing plane S.

c) Model training and performance assessment: The algebraic characteristics of the input data lead us to the choice of two different models, being a CNN-1D for spatial profiles S_{1D}, and a CNN-2D for single-channel images of the flow quantities I_{2D}. Both of them are trained to perform regression over the target labels.

i) *K-fold cross validation*: To assess the performance

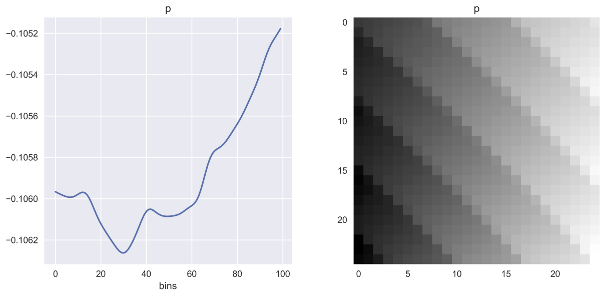

Figure 9: One-dimensional and two-dimensional discretization of the pressure field of S, located at a distance of ($x_s = 50c, y_s = 50c$) from Γ.

of the models in problems of increasing complexity, two configurations are considered:

- Single geometry Γ: The dataset is composed of l CFD simulations about the same geometry Γ_i.
- Multiple geometries Γ: The dataset is made of l CFD simulations computed on different shapes $\{\Gamma_i,\ i = 1, \dots, l\}$. This obviously increases the complexity of the problem, since different geometries of Γ produce dissimilar effects on the flow's evolution.

Considering the total extent of the input domain Ω, being $\Omega_x \in [0, 550c]$ and $\Omega_y \in [-550c, 550c]$, we evaluate the models in terms of the percentage error $e[\%]$ of the predicted labels, as shown in Table 2.

Dataset	Features	Evaluation metrics				
		$\lvert e_x \rvert$	$e_x[\%]$	$\lvert e_y \rvert$	$e_y[\%]$	$\bar{e}[\%]$
Single Γ	$S_{p,1D,r=100}$	46,12	8,39%	46,10	4,19%	6,29%
	$I_{p,2D,r=(50\times50)}$	19,22	3,49%	18,13	1,65%	2,57%
Multiple Γ	$S_{p,1D,r=100}$	65,32	11,9%	73,01	7,61%	9,74%
	$I_{p,2D,r=(50\times50)}$	24,47	4,08%	28,92	2,63%	3,36%

Table 2: 5-fold cross validation results on the inference of spatial properties.

Single-channel images $I_{p,2D}$ clearly provide very accurate results, with a regression error $e[\%] < 4\%$. Spatial profiles $S_{p,1D}$ still manage to achieve good performance, reaching errors $e[\%] < 7\%$. However, the discrepancy increases for more complex domains, i.e. the one consisting of multiple shapes of Γ. In this case, 2D images keep the regression error below 4%, while 1D signals fail to fall below 9%. This is because data yielded from a one-dimensional discretization contain more damped spatial information that negatively affect the quality of results obtained.

ii) *Variation of the spatial resolution*: To study how the performance vary as a function of the discretization factor r, we train different models by progressively incresing the spatial resolution of the input images $I_{p,2D}$, as shown in Figure 10. Specifically, these are sampled from the following list: $[(25 \times 25), (50 \times 50), (75 \times 75), (100 \times 100)]$.

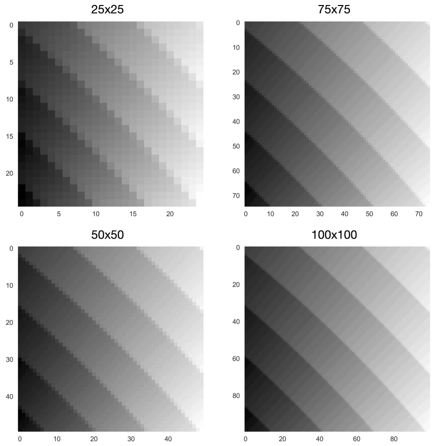

Figure 10: Different resolutions of the single-channel images of the pressure field $I_{p,2D}$ for the same sensing plane S.

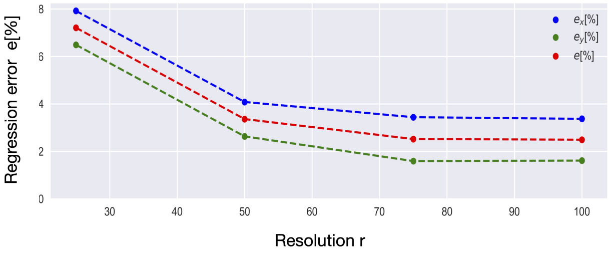

Figure 11: Regression error $e[\%]$ as a function of the discretization factor r.

As reported in Figure 11, it turns out that even with relatively low resolutions, i.e. $r = (50 \times 50)$, the model obtains very good results, with a regression error $e[\%] < 4\%$. Then, it reaches a plateau with higher values of r, proving that, despite the increased details, the input data achieved their maximum informativeness with respect to the problem's complexity.

6. Conclusions

In light of the results obtained, we have demonstrated that Deep Learning can effectively extract and learn relevant patterns from the CFD data to predict useful properties of fluid mechanical systems, when the knowledge of the flow field does not immediately provide required high-level target information. By pursuing a DL pipeline of data processing and model training, we have shown that even in CFD-related problems, the extraction of effective and meaningful information can be demanded to the algorithm itself, leading to very good predictive results. The flexibility of the proposed framework is demonstrated experimentally through the analysis of two examples concerning the airflow around wing sections. Despite the relative simplicity of both the geometrical characteristics and turbulence models, the goal of predicting the shape of airfoils and their spatial location certainly provides interesting insights into its possible future applications, ranging from industrial to medical and beyond.

References

[1] K. Duraisamy, G. Iaccarino, and H. Xiao. Turbulence modeling in the age of data. *Annual Reviews of Fluid Mechanics*, 2019.

[2] J. Ling, A. Kurzawsk, and J. Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 2016.

[3] A. Schillaci, M. Quadrio, C. Pipolo, M. Restelli, and G. Boracchi. Inferring functional properties from fluid dynamics features. *25th International Conference on Pattern Recognition (ICPR)*, 2020.