Politecnico di Torino

ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

POLITECNICO MILANO 1863

Doctoral Dissertation
Doctoral Program in Artificial Intelligence (38th cycle)

Machine Learning Models for Inference from Flow Fields

Feature Extraction and Data Augmentation in Computational Fluid Dynamics

Riccardo Margheritti

* * * * *

Supervisors
Prof. Giacomo Boracchi, Supervisor
Prof. Maurizio Quadrio, Co-supervisor

Politecnico di Torino,
Politecnico di Milano
2026

I hereby declare that, the contents and organisation of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

. . . . . .

Riccardo Margheritti
Milan, January 28, 2026

This publication is part of the project PNRR–NGEU, which has received funding from the Ministero dell'Università e della Ricerca (MUR) – DM 351/2022.

Summary

High-fidelity Computational Fluid Dynamics (CFD) simulations provide detailed descriptions of fluid flows across a wide range of scientific and engineering domains. This intrinsic richness makes CFD data an ideal yet challenging candidate for Machine Learning (ML), as it contains latent, high-level information that is not directly accessible through classical analysis but can be extracted through data-driven models. However, especially in realistic settings, the resulting flow fields tend to be extremely high-dimensional, strongly dependent on geometry, and computationally expensive to generate, which limits their direct usability within ML pipelines. In practical applications, these challenges translate into two coupled limitations: the scarcity of labelled simulations (*Small-n*) and the very large number of degrees of freedom per sample (*Large-p*), which render direct end-to-end learning from CFD data ill-posed and unable to generalise across geometries under realistic data budgets. This thesis addresses ML-based inference from CFD precisely in this regime, where the goal is not to reproduce physical quantities already computable from governing equations, but to infer high-level, non-computable properties such as geometric defects or pathological conditions.

The methodological contribution of this work is structured along two complementary directions, each targeting one of the two limitations aforementioned. The first direction addresses *Small-n* through a geometry-based data augmentation framework, with particular emphasis on diagnostic scenarios involving pathological conditions. Instead of attempting to augment flow fields directly, an operation that can easily violate physical constraints or alter semantic labels, the proposed approach acts on the computational domain. Starting from a single reference geometry on which specific pathologies are explicitly defined, computational geometry and shape correspondence techniques are used to transfer these deformations onto anatomies extracted from healthy patients, generating synthetic pathological variants in a controlled and physically consistent manner. CFD simulations performed on the resulting augmented geometries yield an enlarged labelled dataset while preserving physical admissibility and label consistency. This strategy is designed to increase variability in a principled way and to reduce the dependence on costly expert annotation and simulation campaigns.

The second direction addresses *Large-p* by developing feature extraction strategies that compress CFD outputs into compact, informative, and learnable representations. Two strategies are investigated. The first is a physics-based clustering approach, where the computational domain is segmented into meaningful regions by clustering quantities derived from the governing equations (e.g., contributions associated with advection, diffusion, pressure gradients, and turbulence-related terms). Features are then computed as regional averages and geometric descriptors, enabling adaptive region definition without manual definition. The second is a morphing-based approach, in which heterogeneous simulations are aligned onto a common reference domain through smooth deformation models; this alignment enables expert-defined regions to be specified once on the reference geometry and then reused consistently across samples. Together, these strategies balance adaptability (physics-based region definition) and transferability (cross-sample region consistency), providing scalable feature definition in the presence of strong geometric variability.

The proposed framework is validated on application scenarios of increasing complexity, including two-dimensional aerodynamic flows around airfoils with controlled geometric variations and three-dimensional simulations of airflow in patient-specific upper airways for pathology classification. Across these settings, the combination of geometry-based augmentation and physically grounded feature extraction enables robust inference, improving scalability and generalisation relative to fully manual, case-dependent feature engineering. As an additional contribution, this thesis also produces and releases curated datasets of geometries, CFD simulations, and derived features, intended to support reproducible research and further developments in ML-based inference from CFD.

Contents

List of Tables x

List of Figures xi

Motivation 3

1 Introduction 5
- 1.1 Large-p vs. Small-n: The Data Availability Paradox 7
- 1.2 Overview 9
- 1.3 Inference from CFD data: Mathematical Formulation 10
- 1.4 Published Works 10
- 1.5 Thesis Outline 11

2 ML & CFD: Existing Literature and Datasets 13
- 2.1 ML Applications in CFD 13
- 2.2 Inference from CFD Data: Baseline Approach 17
- 2.3 CFD Datasets and Benchmarking 19

3 Application Scenarios 21
- 3.1 Aerodynamic Scenarios: NACA Airfoils 21
  - 3.1.1 Task 1: Airfoil Shape Identification 22
  - 3.1.2 Task 2: Surface Defect Detection 23
- 3.2 Biomedical Scenario: Human Upper Airways 24
  - 3.2.1 Task 3: Pathology Classification 26

4 Augmenting CFD Data through Computational Geometry 29
- 4.1 Motivation and Overview 29
- 4.2 Related Works and Background 31
- 4.3 Data Augmentation: Problem Formulation 33
- 4.4 Methodology 34
  - 4.4.1 Reference Surface S^{ref} and Deformation Primitives . . . 36
  - 4.4.2 Shape Matching: Cleaning $\tilde{S}_i$ and Matching with S^{ref} . . 39
  - 4.4.3 Deformation Transfer via Functional Correspondence . . 43

4.4.4 From Augmented Geometry to CFD Data 44
4.5 Experiments 45
4.5.1 Training Set Generation 46
4.5.2 CFD Simulations and Feature Extraction 47
4.5.3 Evaluation Protocol and Experimental Objectives 50
4.5.4 ML Model Training 51
4.6 Results 53
4.6.1 Synthetic Validation via Leave-One-Patient-Out Cross-Validation 53
4.6.2 Generalisation to Real Patients 56
4.6.3 Effect of Pathology Severity and Consistency under Deformation Strengthening 57
4.6.4 Explainability 59
4.7 Conclusions 63

5 Feature Extraction from CFD Data via Clustering and Morphing 67
5.1 Motivation and Overview 67
5.2 Related Works and Background 70
5.3 Region Definition and Feature Extraction: Problem Formulation 72
5.4 Methodology 73
5.4.1 Clustering-Based Method 76
5.4.2 Morphing-based Method 79
5.4.3 Extraction of Features from Regions $\{R_{i,j}\}_{j=1}^{r_i}$ 83
5.4.4 Inference Models 84
5.5 Experiments 86
5.5.1 Tasks and Employed Datasets 87
5.5.2 Baseline: Hand-Crafted Regions $\{R_{i,j}\}_{j=1}^{r_i}$ 90
5.5.3 Conducted Experiments 91
5.5.4 Morphing onto the Reference Geometry M^* 92
5.5.5 Models Training and Evaluation 95
5.5.6 Challenges in the 3D Extension and Computational Costs 96
5.6 Results 97
5.7 Conclusions 100

6 CFD Datasets Generation 103
6.1 Defected NACA Airfoils 103
6.1.1 Geometry Parametrization 103
6.1.2 Selection of Baseline NACA Airfoils and Defect Configurations 108
6.1.3 CFD Simulation Setup 110
6.1.4 Data Post-processing 114
6.2 Human Upper Airways from CT Scans 116

6.2.1 Extraction of the Surface $\tilde{S}_i$ from CT Scans 116
6.2.2 Data Augmentation based on Computational Geometry . 117
6.2.3 CFD Simulation Setup 118
6.2.4 Data Post-processing 121

7 Concluding Remarks 125
7.1 Research Directions 126

A Functional Maps and Spectral Shape Correspondence 129
A.1 Functional Maps Formulation 129
A.2 Choice of the Basis and Functional Map Estimation 130
A.3 Partial Correspondence and Registration in Clutter 131
A.4 Multi-scale Refinement via ZoomOut 132

Nomenclature 135

Bibliography 139

List of Tables

4.1 Summary of the sets we use in the experiments 47
4.2 Summary of ML classifier architectures 52
4.3 Classification results obtained during Leave-One-Patient-Out Cross-Validation 53
4.4 Classification accuracy for classifiers trained on progressively enriched synthetic datasets 58
5.1 Summary of the features composing the set of features $\mathbf{P}_i$. . . . 85
5.2 Mapping between the five-digit and three-digit codes representing the airfoil surface deformations 88
5.3 Summary of the conducted experiments. Each setting differs in how regions are defined, which features are extracted, and the inference model employed. 93
5.4 Summary of the inference model architectures used for ML training 96
5.5 Test accuracy across all tasks. 98
5.6 Mean Absolute Error and standard deviation (σ) in *AirNACA* and *AirDEF*. 100
6.1 NACA deformation codes and corresponding shapes for the reference NACA0012 airfoil. 109
6.2 Airfoils: summary of the CFD simulation setup. 113
6.3 Arfoils: summary of the quantities stored per-cell 115
6.4 Nasal airways: summary of the CFD simulation setup. 122

List of Figures

1.1 Comparison of data regimes across domains. 8
2.1 Schematic representation of the classical expert-driven approach for feature extraction from CFD data. 18
3.1 Schematic airfoil representation 22
3.2 CT Scan and 3D Surface 25
3.3 Nasal pathologies investigated in this study visualised through a CT scan 26
4.1 A schematic representation of the data augmentation method adopted in this thesis. 35
4.2 Example of deformation functions defined on the reference surface S^{ref} 38
4.3 Example of geometric clutter arising from CT-based surface extraction. 40
4.4 Correspondence estimation in a registration-in-clutter setting. . 41
4.5 Visualization of pointwise correspondence induced by functional maps. 42
4.6 Transversal sections used for feature extraction in the human upper airways. 48
4.7 ROC curves for LOPO-CV predictions 54
4.8 Confusion matrices illustrating classifier performance 55
4.9 Leave-One-Patient-Out Cross-Validation performance on test patients 56
4.10 Shapley values of the top-ranked CFD-derived features for the classifier K_{Full} 61
4.11 Mean absolute Shapley values aggregated at the level of transversal sections. 62
5.1 Overview of the two region extraction operators in the *Moprhing-based* and *Clustering-based* approaches. 75
5.2 Examples of clustering-based region definition across different application domains 79
5.3 Illustration of the boundary-driven morphing operator $\mathcal{T}_i$ based on RBF interpolation. 81
5.4 ransfer of a scalar field q_i across non-conforming meshes 83

5.5 Cumulative percentage distribution of Cell Volume and Number of Cells with respect to the size of the cells. 89
5.6 Handcrafted regions used in the baseline approach 91
5.7 Schematic representation of a circular domain with a centred NACA airfoil. 94
6.1 Geometry and reference parameters of a cambered NACA airfoil. 104
6.2 Upper airways: visualisation of the computational domain and setup. 118

Motivation

Can fluid dynamics be used not only to study flows and their behaviour, but to reason about the physical systems in which they evolve? Fluid flows are the observable manifestation of geometry, boundary conditions, and underlying physical mechanisms. When simulated through CFD, they encode the integrated effect of these factors into a coherent physical response. This suggests a compelling hypothesis: the flow field itself may act as an informative signature of the system, carrying information that goes beyond local flow quantities. If this assumption holds, CFD simulations move from merely numerical tools for evaluating predefined metrics, to a medium for inference. Properties that are not directly computable, such as the presence of geometric defects, structural anomalies, or pathological conditions, may be inferred indirectly from how they shape the resulting flow. In this perspective, the flow field is not the final outcome, as is typically the case in fluid dynamics studies, but an intermediate representation through which latent characteristics and properties of the physical system can be revealed.

Training ML models on flow field data has specific and non-trivial implications. In engineering applications, ML can enable the identification of defects or performance-limiting features directly from flow responses, without requiring explicit geometric or structural descriptors. In biomedical contexts, simulated airflow can encode clinically relevant information about anatomical conditions, which can be extracted to support data-driven diagnosis. This represents a departure from standard clinical practice, where diagnosis is primarily based on static anatomical information, such as CT scans, and qualitative visual assessment.

This thesis addresses the feasibility and relevance of inference from fluid dynamics, exploring how high-fidelity CFD simulations can be transformed into actionable knowledge by coupling physics-based modelling with ML techniques designed to extract and exploit the latent information embedded in flow fields.

Chapter 1

Introduction

Fluid dynamics plays a central role in engineering and science, as it governs a wide range of natural and technological processes in which fluid motion mediates the interaction between geometry, forces, and transport mechanisms. Understanding how these interactions shape flow behaviour can be essential across many application domains, including aerodynamics, propulsion, biomedical flows, automotive engineering, and environmental systems. Within this context, CFD has become a cornerstone methodology, enabling the analysis of fluid motion through the numerical solution of the Navier–Stokes equations. By discretising the physical domain into a computational mesh, CFD makes it possible to access detailed flow quantities, such as velocity, pressure, and turbulence-related measures, allowing the investigation of complex flow phenomena that are otherwise inaccessible to analytical or experimental approaches [20, 65, 83, 104]. Despite its accuracy and versatility, CFD is rather a computationally intensive technique: high-fidelity simulations require substantial resources and produce vast multidimensional data, often composed of millions of spatial points and several physical variables. This richness of information, while valuable, generates challenges of scale, interpretation, and storage, making it difficult to directly exploit CFD outputs for analysis or learning tasks.

In parallel, ML has rapidly evolved into a general framework for extracting structure and building predictive models from complex data, enabling inference when explicit analytical modelling is unavailable or impractical [50, 59, 94]. This perspective has progressively influenced CFD, giving rise to an expanding body of work at the intersection of ML and CFD. Within this context, a substantial portion of the ML-for-CFD literature employs learning-based models to approximate, accelerate, or complement numerical solutions of fluid flow problems [11, 107]. Typical applications include the reconstruction of flow fields from sparse or partial observations [39, 38, 87], the acceleration of numerical simulation workflows [109], and the improvement of turbulence closures through

data-driven modelling [85, 86]. Taken together, these efforts position ML primarily as a surrogate or auxiliary component within traditional CFD pipelines, with the objective of reproducing, enhancing, or accelerating quantities that are already explicitly defined by the governing equations.

The direction explored in this thesis builds upon this foundation, but shifts the focus from computing and modelling to inference. Rather than using ML to accelerate simulations or improve numerical solvers, CFD results are exploited here as a source of information to infer higher-level quantities that are not directly computable from the governing equations. In this perspective, CFD is no longer regarded primarily as a numerical tool for integrating flow equations, but as a generator of rich and physically grounded data. These data can be structured and analysed through ML to support inference beyond the scope of first-principles models, with CFD acting as an intermediate representation that bridges physical modelling and inference.

Engaging in this research direction, however, poses several inherent challenges. The first concerns the dimensionality of CFD data: the numerical solution of the governing equations produces extremely large flow fields, defined over fine spatial discretisations and involving multiple physical quantities, which makes the direct application of ML computationally prohibitive. The second challenge lies in the limited availability of labelled data, as generating annotated CFD simulations is costly and often requires expert supervision. Together, these aspects lead to a *Large-p, Small-n* regime [48], where n denotes the number of available labelled simulations and p the dimensionality of the discretised input representation provided to the learning model. In this setting, the number of variables vastly exceeds the number of observations, making the learning problem ill-posed due to the inability to reliably generalise across geometries from the limited available data, in contrast with the classical *Small-p, Large-n* regime for which most ML methods have been originally developed.

In many data-driven applications, this imbalance is alleviated through automated representation learning techniques, such as autoencoding or unsupervised dimensionality reduction [37, 72, 71, 73], which map high-dimensional inputs onto compact latent spaces. In CFD, these approaches have proven effective for tasks such as compression, reconstruction, or surrogate modelling of flow fields. However, when the goal is inference rather than reconstruction, their limitations become apparent. When dealing with *Small-n*, only a limited number of latent modes can be reliably extracted, and these modes primarily reflect dominant sources of global variance in the data. In CFD, such modes are typically associated with energetic, large-scale flow structures. However, variations that are critical for inference, often induced by geometric differences, may manifest as localised or structurally specific changes in the flow field, which do not necessarily dominate the global variance and are therefore not captured by the leading latent modes.

At the same time, geometric variability constitutes a fundamental challenge for CFD-based inference. Even small differences in the underlying geometry, such as those arising across individual anatomies or manufacturing defects, can induce substantial and highly structured modifications of the flow. These changes, while functionally significant, are geometry-dependent and often localised, causing flow fields to be defined over different spatial domains and hindering direct comparison across samples. Crucially, adequately accounting for this variability would require a prohibitively large number of geometrically distinct samples, which is infeasible in practice.

Finally, the high computational cost of generating CFD data further limits the feasibility of training large models or collecting extensive labelled datasets. High-fidelity simulations such as RANS, LES, or DNS [83] require large computational infrastructures and can take from hours to days to complete, making data generation a severe bottleneck. To mitigate the aforementioned limitations, it is essential to design representations that are both physically meaningful and computationally efficient, capturing the relevant flow information while remaining robust to geometric and numerical variability.

1.1 Large-p vs. Small-n: The Data Availability Paradox

The challenges outlined so far ultimately stem from the intrinsic data regime induced by CFD simulations, where the dimensionality p vastly exceeds the number of available samples n. This regime is characteristic of learning problems arising in scientific and engineering contexts, where data are generated through the numerical solution of governing equations, rather than through passive observation of large-scale datasets [5, 53]. Unlike learning pipelines typically developed in conventional Deep Learning (DL) settings, where data are abundant and approximately IID (independent and Identically Distributed), the present framework must operate on structured, high-dimensional fields governed by physical constraints, extracting predictive structure from very few samples.

In classical DL, models are typically trained on datasets comprising millions of samples ($n \sim 10^6$), where each observation has a comparatively moderate dimensionality, with p ranging up to $\sim 10^5$, corresponding to the number of pixels and channels in an image. Widely used benchmarks such as MNIST [58], CIFAR-10 [56], ImageNet [24], and RoboNet [23] are representative of this *Small-p, Large-n* regime. In contrast, high-fidelity Computational Fluid Dynamics operates in a markedly different data regime. In this case, the dimensionality p scales with the number of computational cells multiplied by the number of stored physical variables defining the flow state. A single Direct Numerical Simulation (DNS) of turbulent flow may therefore involve up to $p \sim 10^9$ degrees

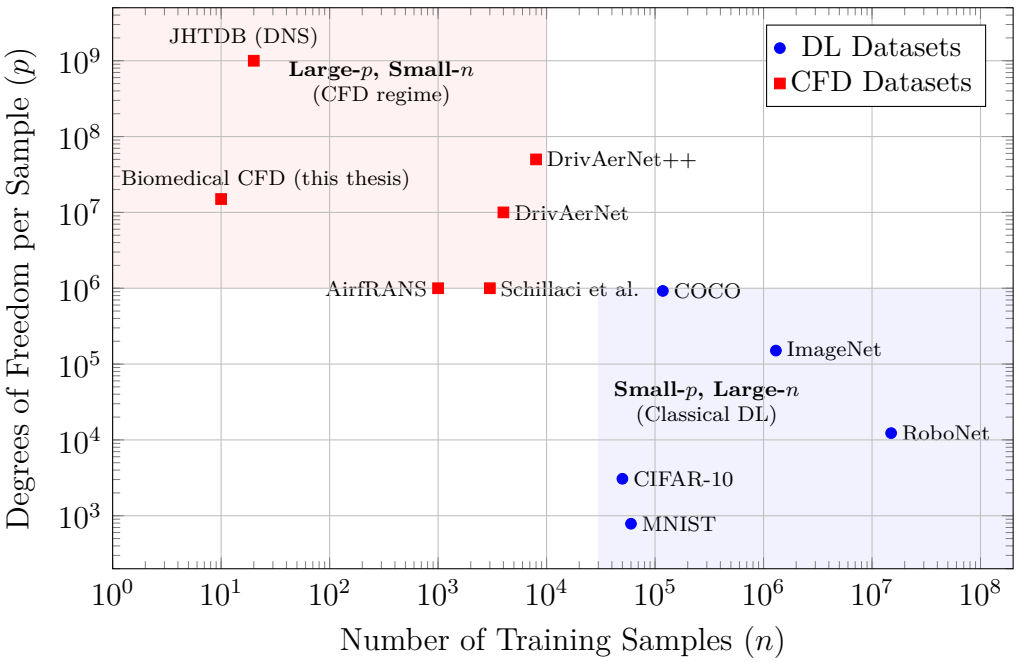


Figure 1.1: Comparison of data regimes across domains. Standard Deep Learning/ Computer Vision datasets (blue) operate in a Small-p, Large-n regime. CFD and turbulence simulations, as well as biomedical CFD (red), lie in the *Large-p, Small-n* region. Values are approximate and based on [52, 10, 97, 32, 31, 56, 24, 61, 60, 105, 23].

of freedom [60], while individual Large Eddy Simulation (LES) snapshots routinely exceed 10^7–10^8 degrees of freedom. Due to the extreme computational cost of generating such data, the number of available simulations is typically limited to $n \sim 10^1$–10^2 samples. Comparable constraints arise in aerodynamic benchmarks [105, 52, 10, 32, 31, 96, 56], and in biomedical CFD applications, where additional factors such as anatomical variability, data acquisition effort, and privacy restrictions further limit the availability of labelled simulations.

This mismatch is illustrated in Figure 1.1, which contrasts standard DL datasets with representative CFD datasets. The comparison highlights how CFD-based inference operates in a regime characterised by extremely high-dimensional states and severe sample scarcity, explaining why approaches effective in conventional DL settings may fail when applied naively to CFD data. Addressing this imbalance requires specialised strategies to mitigate overfitting and preserve physical consistency, which naturally decompose into two complementary methodological directions:

1. **Domain-Specific Data Augmentation** Increasing n by generating synthetic samples that preserve physical structure, symmetries, or PDE residuals.

2. **Physics-Based Dimensionality Reduction** Reducing p by extracting structured, interpretable representations, e.g., modal coefficients, clusters, reduced-order models, rather than operating on raw CFD fields.

The methodological contributions of this thesis are structured around these two directions.

1.2 Overview

The objective of this work is therefore to develop scalable ML-based methods for the efficient handling of CFD data, enabling reliable inference even in the presence of limited training data and complex geometric variability. This is pursued by reducing dimensionality while preserving physical content, exploiting the structure of the governing equations for feature extraction, and progressively automating preprocessing steps that have traditionally been performed by domain experts.

The proposed methodology builds upon three ideas, each designed to address a specific challenge in learning from CFD data. The first component, a data-augmentation strategy grounded in computational geometry, targets the scarcity of labelled data (*Small-n*). Computational Geometry techniques are employed to transfer expert-defined deformations across geometries, generating synthetic yet physically consistent geometries aimed at expanding the CFD datasets while ensuring a consistent and unambiguous labelling. The second component, a physics-based clustering approach, tackles the challenge of high dimensionality (*Large-p*) and feature scalability, partitioning the CFD flow field into coherent regions based on local physical balances. This enables the automatic extraction of compact and physically meaningful features, removing the need for predefined spatial divisions or manual region selection. Finally, the third component, a morphing-based alignment technique, addresses the problem of geometric variability by mapping different geometries and their associated flow fields onto a shared reference domain through radial basis function deformation, ensuring consistency and comparability across samples.

Overall, these contributions define a coherent methodological approach that integrates physical modelling, geometric tools, and expert knowledge to mitigate key limitations of CFD-based inference. Validation across both aerodynamic and clinical scenarios shows that this approach can be effectively applied to the identification of NACA airfoil shapes and surface defects, as well as to the classification of nasal pathologies from airflow simulations. In all cases, it proves capable of handling high-dimensional CFD data, generalising across strongly heterogeneous geometries, and remaining effective even when only limited training data are available. These results support the use of CFD not

only as a numerical simulation tool, but as a basis for extracting higher-level, diagnostic information from complex flow fields.

1.3 Inference from CFD data: Mathematical Formulation

In this Section, we formalise the core problem that underpins the work presented in this thesis. Let $\Omega_i \subset \mathbb{R}^3$ be the domain of the i-th CFD simulation, bounded by the surface S_i and discretized by a mesh

$$M_i = \{\mathbf{x}_{i,k}\}_{k=1}^{n_i}, \quad \mathbf{x}_{i,k} \in \Omega_i,$$

where, $\mathbf{x}_{i,k}$ denotes the k-th node of the i-th mesh. Each CFD simulation can be seen as a data matrix

$$\mathbf{F}_i \in \mathbb{R}^{n_i \times H},$$

where each row of $\mathbf{F}_i$ contains the H quantities computed at the k-th node of M_i (e.g., spatial coordinates, pressure, and velocity components).
We denote by $\mathcal{F} = \{\mathbf{F}_i\}_{i=1}^N$ the set of available CFD data and by $\mathcal{Y} = \{Y_i\}_{i=1}^N$ the corresponding set of target labels, each Y_i representing a physical or diagnostic class (for instance, a pathology, a deformation type, or an aerodynamic condition). The goal is to learn a mapping

$$K : \mathcal{F} \longrightarrow \mathcal{Y},$$

such that $K(\mathbf{F}_i) \approx Y_i$ for all $i = 1, \ldots, N$. Each CFD matrix $\mathbf{F}_i$ can contain up to $n_i \sim 10^7$ rows, making it impractical to be directly used as input to any ML model. Moreover, the number of available simulations N is typically small, since generating each tuple $(S_i, M_i, \mathbf{F}_i)$ requires high-fidelity CFD runs and expert labelling. Here, the dimensionality of each sample ($p = n_i \times H$) is orders of magnitude larger than the number of training instances N.

The inference from CFD data is therefore formulated as a supervised learning problem under the following constraints:

$$\begin{cases} n_i \gg N \\ \mathbf{F}_i \text{ depends non-linearly on the geometry } S_i \\ Y_i \text{ is not directly computable from governing equations.} \end{cases}$$

1.4 Published Works

The research activities carried out during the Ph.D. program led to the development of novel methodologies that have been published or submitted to

international peer-reviewed conferences and journals. The following list reports the bibliographic details of these works and their relationship with the chapters of this dissertation:

- **Chapter 4** is based on the methodology for data augmentation via computational geometry presented in:

  R. Margheritti, A. Schillaci, C. Pipolo, M. Quadrio, and G. Boracchi. "Leveraging Computational Geometry for Data Augmentation in Medical Flow Fields Classification". In *Proceedings of the Engineering Applications of Neural Networks (EANN 2025)*, Springer Nature Switzerland, pp. 109-122, 2025 [66].

  R. Margheritti, A. Schillaci, C. Pipolo, M. Quadrio, and G. Boracchi. "Data Augmentation Based on Computational Geometry for Neural Network Training in Medical Flow Field Classification". **Submitted** to *Computer and Graphics*.

- **Chapter 5** builds upon the physics-based clustering framework introduced in:

  R. Margheritti, O. Semeraro, M. Quadrio, and G. Boracchi. "Physics-Based Region Clustering to Boost Inference on Computational Fluid Dynamics Flow Fields". In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2025)*, Springer Nature Switzerland, 2025 [68].

  R. Margheritti, O. Semeraro, M. Quadrio, and G. Boracchi. "Feature Extraction from Flow Fields: Physics-Based Clustering and Morphing with Applications". *Applied Sciences*, MDPI, 2025. [67]

In addition to these contributions, this research also led to the generation of a novel CFD simulation dataset and to the curation and public release of an existing one, with the aim of promoting reproducibility and supporting further research in the field.

1.5 Thesis Outline

This thesis is structured as follows:

- Chapter 2 reviews the relevant literature at the intersection of CFD and ML. In addition, it introduces the baseline inference approaches adopted throughout the thesis and surveys existing CFD datasets commonly used in learning-based studies.

- Chapter 3 defines the application scenarios and inference tasks considered in this work. It covers both aerodynamic and biomedical domains, formalising the prediction problems addressed in the thesis.

- Chapter 4 addresses the problem of data scarcity (*Small-n*) in learning-based inference from CFD simulations. It introduces a geometry-based data augmentation framework that leverages computational geometry and shape correspondence techniques to generate additional labelled samples while preserving physical admissibility and semantic consistency. The effectiveness of the proposed augmentation strategy is evaluated on biomedical CFD data involving patient-specific anatomies.

- Chapter 5 focuses on the complementary challenge of high-dimensionality (*Large-p*) in CFD data. It introduces scalable feature extraction strategies based on physics-informed clustering and morphing-based geometric alignment, which enable compact and interpretable representations across heterogeneous geometries. The proposed methods are validated on both aerodynamic and biomedical datasets, demonstrating improved scalability and performance compared to expert-driven baselines.

- Chapter 6 details the datasets employed throughout the thesis.

- Finally, Chapter 7 summarises the main contributions of the thesis, discusses their implications for learning-based inference from CFD data, and outlines possible directions for future research. challenges.

Chapter 2

ML & CFD: Existing Literature and Datasets

2.1 ML Applications in CFD

In recent years, the integration of ML and CFD has gained popularity across multiple scientific and engineering domains. This convergence has been largely driven by the increasing computational cost of high-fidelity simulations and the growing flexibility of ML architectures in modelling high-dimensional systems. As documented in several seminal reviews [12, 11, 107, 53, 74], a broad and rapidly expanding body of literature has emerged at the intersection of ML and CFD, with the dominant objective of approximating, accelerating, or enhancing the output of traditional solvers. In these settings, in fact, ML is typically employed as a functional surrogate, trained to reproduce or improve physical quantities that are already governed by the Navier–Stokes equations.

From a historical perspective, CFD has long relied on first-principle equations, discretisation strategies, and High-Performance Computing (HPC). However, as the complexity of target systems increased, ranging from turbulent aerodynamic flows to multi-physics biomedical environments, standard numerical solvers began to encounter intrinsic limitations in terms of computational cost, memory footprint, and scalability. In parallel, ML techniques have matured significantly, evolving from simple pattern recognition tools into flexible function approximators capable of learning nonlinear operators, latent structures, and high-dimensional mappings [50, 59, 94]. This conceptual shift has made ML a natural candidate for augmenting or complementing classical CFD pipelines.

Early data-driven modelling efforts in CFD were primarily motivated by the need to reduce the dimensionality of high-fidelity simulations, and relied on linear modal decomposition techniques such as Proper Orthogonal Decomposition (POD) and Dynamic Mode Decomposition (DMD) [92, 102, 37, 72, 71, 73].

These methods project the flow field onto a low-dimensional subspace spanned by a small number of dominant modes, providing compact representations that are well suited for analysis, reduced-order modelling, and flow reconstruction. However, despite their effectiveness in capturing energetically dominant structures, such approaches remain fundamentally linear and are primarily designed to approximate or reconstruct the underlying physical fields.

As a result, the field has progressively shifted towards non-linear DL architectures, which offer greater expressive power and flexibility in modelling complex, high-dimensional dynamics. A prominent class of methods in this strand of work is represented by PINNs, which embed the residuals of the governing equations directly into the training loss of a neural network. These models are capable of solving forward and inverse problems with limited data, and have been successfully applied to a range of canonical PDE systems [87, 112, 79, 30]. However, PINNs and similar approaches operate under the explicit goal of approximating physical fields (e.g., velocity, pressure, vorticity) that are already defined by first-principles models, albeit at high computational cost. In PINNs, governing equations are treated as constraints and data are used to regularise the solution, without attempting to extract higher-level semantic or functional attributes.

Another major research direction concerns data-driven turbulence modelling, where ML is employed to improve or replace classical closure models within Reynolds-Averaged Navier–Stokes (RANS) and related formulations. In this setting, ML models are trained to infer unresolved turbulent quantities from resolved flow variables [85, 86]. A seminal contribution in this area was introduced by Ling et al. [62], who proposed deep neural networks to regress Reynolds stress tensors from mean-flow quantities while explicitly enforcing physical invariance properties. This line of work was subsequently extended in [57] through training strategies aimed at improving robustness and generalisation across flow configurations, and further generalised in [3, 110] by incorporating additional physical constraints directly into the model architecture.

Beyond neural-network-based approaches, alternative learning paradigms have also been explored. Decision-tree methods [28] and ensemble learning techniques [69] have been proposed to enhance interpretability and generalisation, while tensor-basis networks [63] explicitly encode the structure of the Reynolds stress tensor into the learning process. More recently, symbolic regression approaches, such as the genetic programming framework introduced in [115], have been investigated to derive interpretable, closed-form turbulence models from data. Yet across all these efforts, the learning objective remains confined to reconstructing physical quantities within the space of the solver: the ML model predicts Reynolds stresses, turbulent viscosity, or subgrid fluxes, quantities that are either directly or indirectly computable from first principles.

Related strategies have also been proposed for regression and super-resolution

of CFD outputs. Fukami et al. [38], for example, applied convolutional neural networks (CNNs) to reconstruct fine-scale turbulent fields from coarsened inputs, using paired datasets of low- and high-resolution simulations. Their models accurately recovered velocity and vorticity fields across benchmark datasets. However, as in the previous cases, the learning task consists of mapping from one physical representation to another, coarse to fine, under-resolved to resolved, without extending the semantic scope of the simulation.

In contrast to this class of problems, where the objective is to *compute* or *enhance* fluid quantities, the present thesis addresses a fundamentally different challenge: *inferring* high-level, non-computable information from CFD simulations, as discussed in Section 1.2. In this formulation, ML is used to extract semantic attributes that are not encoded in the governing equations. These include functional or diagnostic labels, such as the presence of a structural defect, a pathological condition, or a discrete classification of the underlying geometry. The flow field becomes the input to the learning model, rather than its output, and the inference task is defined in terms of system-level understanding rather than numerical approximation.

A limited number of studies have attempted to perform inference from CFD data, though typically under highly constrained and simplified settings. Zakeri et al. [113] simulated blood flow in cerebral aneurysms using RANS, and extracted scalar descriptors such as wall shear stress and oscillatory shear index to predict rupture risk via classical supervised classifiers. While the target variable is non-physical and clinically relevant, the model operates exclusively on a small set of quantities, computed globally from each simulation. The CFD fields themselves are not used as structured inputs, and geometric variability is handled implicitly through population-level statistics. No spatial structure is preserved in the learning process.

A similar strategy is followed by Eastvedt et al. [29], who used transient CFD simulations to identify faults in subsea pipelines. Here, the inference model takes as input time-series of pressure and velocity recorded at fixed locations, and predicts the presence or type of fault. The approach treats CFD as a black-box simulator that generates global diagnostic signals, but does not process the spatial distribution of the field, nor account for varying geometries or topologies. In both cases, learning is performed on tabular data extracted from the simulation, and the inference problem is framed in terms of global classification from low-dimensional signals.

A more structured yet still ad hoc approach is presented in González et al. [42], who trained convolutional networks to detect embedded defects in composite moulding processes, using pressure signals synthetically generated from CFD. The model predicts the position, size, and permeability of dissimilar material regions by processing 2D grayscale images derived from temporal sensor

data. Although the input representation is spatially organized, it remains abstracted: the CFD field is not used directly, but transformed into a surrogate view of sensor readings over time. Furthermore, the entire setup assumes a fixed geometry and sensor layout, with no strategy for generalising across configurations or geometries. The model is tailored to a single scenario, and no intermediate representations of the flow are extracted or interpreted.

In all three cases, the use of CFD is limited to generating simplified or indirect inputs to the learning model, scalar descriptors, time-series, or synthetic sensor images. None of these methods operate directly on the full-resolution flow fields, nor do they attempt to preserve or align the geometric and spatial structure of the domain. A similar approach to the inference from CFD was also introduced by Schillaci et al. [97], whose work constitutes the methodological baseline for this thesis. In that study, CFD simulations were performed over parametric geometries, including NACA airfoils and simplified models of nasal cavities, where each configuration was described by a low-dimensional set of geometric parameters. To reduce dimensionality and enforce spatial consistency, the authors manually defined a set of regions, cross-sections or anatomical slices, over which flow quantities were averaged. The resulting low-dimensional feature vector was used to train standard classifiers to perform regression, demonstrating that CFD flow fields encode sufficient information to recover the parameters governing the generation of the underlying geometry. Despite its effectiveness in controlled settings, this framework remains tightly coupled to the availability of a parametric description of the geometry and to expert-defined spatial partitions. As such, it does not readily extend to realistic or heterogeneous anatomies for which no explicit parametric model is available. Moreover, the reliance on handcrafted, domain-specific preprocessing limits scalability and generalisation beyond the initial geometric templates considered in the study.

A subsequent study [95] further explored the comparative value of CFD-derived features versus anatomical descriptors in nasal pathology identification. The analysis compared models trained on geometric quantities (e.g., curvature, cross-sectional area, local thickness) with those trained on flow-based descriptors extracted from CFD simulations (e.g., velocity, pressure drops, shear stress), using the same class of simplified, parametrically generated anatomies as in [97]. Results showed that flow-derived models consistently outperformed their geometry-based counterparts, indicating that CFD simulations encode latent, diagnostically relevant information not recoverable from shape descriptors alone. While still limited to synthetic geometries, this study reinforces the central assumption that flow fields carry discriminative information about high-level targets, and that CFD can serve as a physically grounded input representation for semantic inference.

Altogether, these contributions highlight both the feasibility and the limitations of current approaches to inference from CFD. Most existing methods

rely on either low-dimensional global descriptors or handcrafted features, and do not account for the spatial structure of the field, the variability of the underlying geometries, or the need for generalisation across topologies. Only a few studies attempt to operate directly on CFD fields, and those that do typically neglect geometric alignment and lack mechanisms for extracting structured or transferable representations.

To formally benchmark the methods proposed in this thesis, we adopt as a baseline the expert-driven inference approach introduced in [97], which demonstrated that semantic classification can be performed from CFD-derived features in controlled, synthetic settings. The present work retains the core structure of that methodology, but evaluates its applicability under more challenging conditions involving realistic geometries and limited data. The following section details the implementation and assumptions of this baseline, which serves as the primary reference for all comparative evaluations throughout the thesis.

2.2 Inference from CFD Data: Baseline Approach

To address the challenge of the *Large-p, Small-n* regime, we adopt as baseline the expert-driven feature extraction methodology originally proposed by Schillaci et al. [97]. As in the original formulation introduced in Section 1.3, the objective is to perform inference from CFD data, that is, to predict high-level, non-computable attributes from simulated flow fields. The baseline approach, illustrated in Figure 2.1, compresses the information contained in the CFD data matrix $\mathbf{F}_i$ into a compact feature vector $\mathbf{P}_i$ by averaging selected flow quantities over a set of manually defined subdomains. These regions are specified a priori based on geometric or anatomical criteria, and reflect expert assumptions about where diagnostically relevant flow patterns are likely to emerge. The following describes the method in full detail as applied in this thesis, serving as a reference for all comparative evaluations.

Given the discretized computational domain M_i associated with the i-th simulation $\mathbf{F}_i$ (left-hand side of Figure 2.1), the baseline approach requires the a priori definition of a set of spatial regions $\mathcal{R}_i = \{R_{i,j}\}_{j=1}^{r_i}$, with each $R_{i,j} \subset \Omega_i$. The number of such regions, r_i, is typically much smaller than the number of mesh cells n_i, and reflects a dimensionality reduction guided by domain expertise. These regions are manually defined based on geometric or anatomical criteria specific to the surface S_i. In [97], this includes volumetric subdomains delimited by anatomical landmarks in upper airway geometries (as illustrated in Figure 2.1), or predefined sections positioned along the chord line in airfoil configurations.

Once the regions are defined, the high-dimensional field $\mathbf{F}_i$ is mapped into

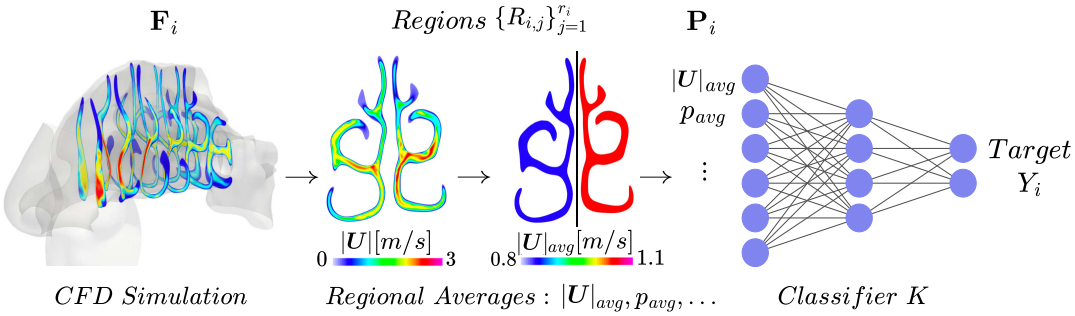


Figure 2.1: Schematic representation of the classical expert-driven approach for feature extraction from CFD data. Starting from the flow field $\mathbf{F}_i$, experts manually define a set of regions $\{R_{i,j}\}_{j=1}^{r_i}$ within the flow domain. Regional averages of selected fluid-dynamic quantities are then computed (e.g., velocity magnitude $|\boldsymbol{U}|$ and pressure p), yielding a feature set $\mathbf{P}_i$ that serves as input to the inference model K for predicting the target Y_i.

a low-dimensional vector $\mathbf{P}_i$ by computing regional averages. Let q denote a generic flow quantity in $\mathbf{F}_i$ (e.g., pressure, velocity components). The regional average related to the j-th region of the i-th CFD simulation is computed as the weighted average of the variable q over the cells belonging to that region. Formally:

$$\bar{q}_{i,j} = \frac{\sum_{k \in \mathcal{I}_{i,j}} q_k w_k}{\sum_{k \in \mathcal{I}_{i,j}} w_k}, \tag{2.1}$$

where $\mathcal{I}_{i,j}$ is the set of indices of the mesh cells belonging to the region $R_{i,j}$, q_k is the value of the flow variable at the k-th cell, and w_k is the geometric weight (volume or area) of the cell. The weighting is essential to account for the non-uniform discretisation typical of unstructured CFD meshes. By reiterating this computation for all r_i regions and for all selected physical variables, we obtain the feature set $\mathbf{P}_i \in \mathbb{R}^{r_i \times D}$. This set serves as input for the inference model K, and the learning task reduces to the approximation of the mapping:

$$K(\mathbf{P}_i) \approx Y_i. \tag{2.2}$$

Limitations of the Expert-Driven Approach While the expert-driven approach ensures direct interpretability of the input features, it suffers from intrinsic limitations that hinder its application to large and diverse datasets:

1. **Dependence on Expert Supervision:** The definition of $\mathcal{R}_i$ requires significant manual intervention and deep domain expertise, making the process time-consuming and costly.

2. **Lack of Geometric Scalability:** In scenarios characterised by high anatomical or geometric variability, ensuring spatial consistency of regions across different samples is non-trivial. Manually redefining regions for every new geometry prevents the automation of the inference pipeline.

3. **Potential Information Loss:** By restricting the analysis to pre-selected regions, there is a risk of discarding localised flow patterns, such as secondary vortices or separation bubbles, that may manifest in areas not monitored by the expert but are nonetheless crucial for the diagnostic task.

These limitations motivate the development of the automated and geometrically robust methods presented in the subsequent chapters.

2.3 CFD Datasets and Benchmarking

The growing adoption of ML techniques in CFD has highlighted the central role of curated, high-fidelity datasets in enabling reproducibility, fair comparison of methods, and systematic benchmarking. Despite this, the availability of publicly accessible CFD datasets remains limited. Most CFD data are still generated ad hoc for specific studies, often tailored to a single configuration, geometry, or flow condition, and are therefore not suitable for large-scale learning, reuse, or systematic evaluation of ML methods.

Many of the classical CFD data resources available in the literature are focused on numerical validation, solver benchmarking, or canonical physical problems. Examples include the ERCOFTAC "Classic Collection" database [1], which provides curated test cases for turbulence and flow modelling developed since the 1990s, and various NASA turbulence modelling and validation resources that collect diverse CFD cases and experimental comparisons [2]. Such databases are invaluable for physical model validation and traditional CFD benchmarking, but they are generally not structured to support ML training over broad geometric or parametric variability.

Among public datasets that are explicitly intended to support machine learning research, the dataset by Schillaci et al. [96] is a notable example in the aerospace domain. It consists of two-dimensional RANS simulations of airfoil flows and is designed for inverse inference of shape parameters from flow fields under fixed flow conditions. More recently, AirfRANS [10] has been introduced as a large-scale ML benchmark providing RANS simulations of two-dimensional

[1] ERCOFTAC "Classic Collection" database: `http://cfd.mace.manchester.ac.uk/ercoftac/doku.php`

[2] Nasa, Turbulence Modeling Resource: `https://turbmodels.larc.nasa.gov/`

airfoils across a wide range of geometries, angles of attack, and Reynolds numbers. Additional airfoil-focused CFD datasets, such as the HAM2D [88] airfoil dataset containing flow fields for thousands of shapes and multiple flow conditions, have been released via open data portals and aim to serve as benchmarks for AI/ML analysis [89].

Canonical turbulence databases also play an important role in ML benchmarking by providing highly controlled flow configurations. The Johns Hopkins Turbulence Database (JHTDB) [52] offers direct numerical simulation data for homogeneous isotropic turbulence, channel flows, and boundary layers that have been widely used for developing and validating learning-based models for turbulence analysis, super-resolution, and flow-field reconstruction. Beyond aerodynamics and turbulence, other emerging CFD datasets with relevance for ML include parametric automotive aerodynamic datasets such as DrivAerNet and DrivAerNet++, which comprise thousands of high-fidelity CFD simulations of diverse vehicle shapes for data-driven aerodynamic design and prediction [2, 31, 32]. These larger-scale collections help address the scarcity of diverse 3D CFD training data in engineering applications.

In biomedical CFD, the availability of large, annotated datasets is even more limited due to the cost of medical imaging, segmentation, and simulation, as well as ethical and privacy constraints. As a result, many studies rely on small sets of patient-specific geometries or combine real data with synthetic augmentation strategies to increase sample diversity.

The persistent scarcity of labelled CFD data highlights the need for methods capable of extracting compact and physically meaningful representations from limited samples, while remaining robust to geometric variability. Taken together, existing datasets and benchmarking efforts reveal a fragmented landscape in which data availability, geometric complexity, and learning objectives vary substantially across application domains. This fragmentation hinders the systematic assessment and comparison of learning-based approaches, especially in severely data-scarce settings such as biomedical CFD.

These observations motivate the adoption of scalable and flexible processing pipelines, able to operate across different CFD domains and data regimes, and inform the methodological choices pursued throughout this thesis. As part of this effort, this work also contributes to improving data accessibility by generating and curating CFD datasets intended to support reproducible research under realistic constraints.

Chapter 3

Application Scenarios

This chapter introduces the application scenarios used to evaluate the methods developed in this thesis. The proposed data augmentation and feature extraction frameworks are assessed on two representative domains: aerodynamics and biomedical engineering. These scenarios are deliberately chosen to span different problem characteristics, including geometric complexity, data availability, and computational cost. Together, they enable a systematic analysis of the robustness, scalability, and practical relevance of the proposed methods under different operating conditions.

The first scenario (*Airfoil Shape Identification and Defect Detection*) represents a controlled engineering environment involving 2D geometries and steady-state RANS simulations. Here, large datasets can be generated parametrically, allowing for extensive statistical validation. The second scenario (*Pathology Classification in Human Upper Airways*) introduces a real-world medical challenge involving complex 3D anatomical geometries and high-fidelity LES simulations. This setting is characterised by extreme data scarcity and high inter-subject variability, representing the primary target for the data augmentation strategies proposed in Chapter 4. Here, we introduce the specific tasks, while the generation of the datasets and the computational setup for the CFD simulations in both domains are described in Chapter 6.

3.1 Aerodynamic Scenarios: NACA Airfoils

In the aerospace domain, we address the problem of inferring geometric properties of an airfoil directly from the surrounding flow field. This inverse problem is relevant for applications such as non-destructive testing, where one aims to detect structural deformations or ice accretion based on aerodynamic measurements. We consider two distinct tasks: shape identification and surface defect detection.

3.1.1 Task 1: Airfoil Shape Identification

The first task consists of predicting the exact shape of a NACA 4-digit airfoil given its flow field. As illustrated in Figure 6.1, the geometry of these airfoils is uniquely defined by three geometric parameters, which are encoded into a four-digit string:

- **Maximum Camber (a):** Represented by the first digit, it quantifies the maximum curvature of the airfoil. Specifically, it measures the maximum distance between the chord line (dashed blue in Figure 6.1) and the mean camber line (dash-dotted red). In the NACA standard, this value is expressed as a percentage of the chord length c.
- **Position of Maximum Camber (b):** Represented by the second digit, this parameter defines the longitudinal location x along the chord where the maximum camber a occurs. It is expressed in tenths of the chord length. In Figure 6.1, this is indicated by the top horizontal dimension line, showing $b = 4$ (i.e., at 40% of the chord).
- **Maximum Thickness (t):** Represented by the last two digits, this parameter indicates the maximum distance between the upper and lower surfaces of the airfoil, corresponding to the Suction Side and the Pressure Side, respectively. The thickness t is depicted in green in Figure 6.1 and is expressed as a percentage of the chord.

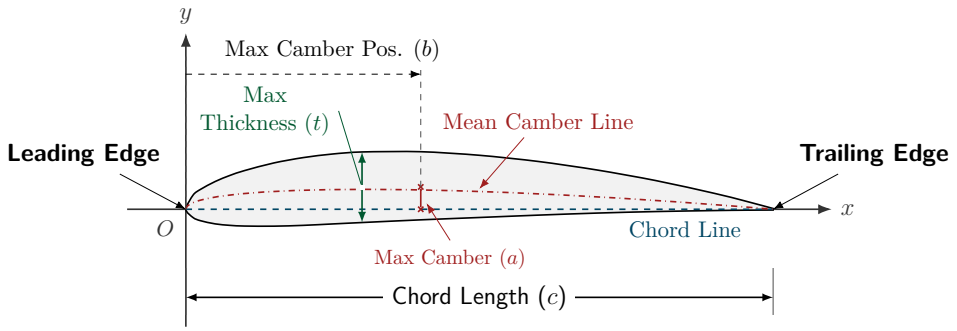


Figure 3.1: Schematic representation of a cambered NACA airfoil in the Cartesian coordinate system (x, y). The diagram illustrates the relationship between the chord line (blue) and the mean camber line (red), separating the domain into the suction side (light blue background) and the pressure side (light red background). The airfoil geometry is described through the chord length (c), the maximum thickness (t) highlighted in green, the maximum camber (a) shown in red, and its chordwise location (b).

The problem is formulated as a multi-output regression task where the goal is to estimate these three parameters (a, b, t) directly from the CFD data. The dataset employed for this task is public [96] and comprises 3025 distinct flow fields generated by varying the digits within standard aerodynamic ranges.

3.1.2 Task 2: Surface Defect Detection

To introduce non-parametric variability and simulate realistic structural anomalies, we generated a synthetic dataset which extends the standard parametric NACA 4-digit definitions by superimposing localised geometric perturbations representing manufacturing defects, operational damage, or surface icing. The generation pipeline modifies the baseline airfoil geometry through two primary mechanisms: Gaussian surface perturbations and chordwise truncation.

Surface Deformations Localised deformations on both the suction and pressure sides are modelled as additive Gaussian perturbations applied to the baseline coordinates. For a given baseline coordinate (x, y_{base}), the modified coordinate y_{mod} is defined as:

$$y_{mod}(x) = y_{base}(x) + \delta \cdot A \cdot \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{3.1}$$

where:

- μ represents the chordwise location of the defect center;
- σ controls the spatial extent (width) of the deformation;
- A denotes the peak intensity or amplitude of the deformation;
- $\delta \in \{+1, -1\}$ determines the nature of the defect. Specifically, $\delta = +1$ produces a *bump* (local protrusion), while $\delta = -1$ generates a *cavity* (local depression).

These perturbations are applied independently to the upper and lower surfaces based on the specific defect configuration required for each sample.

Trailing Edge Cuts To account for realistic manufacturing constraints, we simulate imperfect trailing edges by truncating the airfoil geometry. In practical applications, producing a perfectly sharp trailing edge is often unfeasible due to machining limitations, resulting in a finite thickness or blunt edge. This geometric imperfection is modeled by removing all coordinate points beyond a specified threshold. The final domain is defined as:

$$\mathcal{D}_{cut} = \{(x, y) \in \mathcal{D}_{base} \mid x \leq (1 - C_{cut})\} \tag{3.2}$$

where C_{cut} represents the percentage of the chord removed from the trailing edge.

Encoding Scheme Each defected airfoil is associated with a 3-digit defect code that governs the generation parameters. This code encapsulates the type and severity of deformations present on three distinct zones:

1. **Suction Side:** Presence of bump, cavity, or nominal surface;
2. **Pressure Side:** Presence of bump, cavity, or nominal surface;
3. **Trailing Edge:** The magnitude of the truncation applied (if any).

This parametric approach allows for the generation of 3600 unique simulations. It serves as a test case for assessing the robustness of the proposed methods, specifically regarding the ability to detect localised flow features induced by these surface irregularities. Details on the generation of the dataset can be found in Section 6.1.

3.2 Biomedical Scenario: Human Upper Airways

The second application domain concerns the diagnostic classification of nasal pathologies from airflow simulations. Compared to the aerodynamic benchmark cases, this setting introduces a substantial increase in both geometric and physical complexity. Airfoil data consist of clean, parametric 2D shapes and steady RANS simulations; in contrast, the biomedical scenario relies on patient-specific 3D anatomies reconstructed from medical imaging, each exhibiting unique structural features that cannot be reduced to a simple set of parameters.

This complexity stems from multiple factors. Anatomically, the nasal cavity presents a highly irregular morphology that varies significantly across individuals, with narrow passages, sharp curvatures, and delicate structures that strongly influence airflow. From a fluid dynamical point of view, the nasal airflow regime involves transitional and turbulent features that require high-fidelity Large Eddy Simulations (LES) to be accurately resolved. On top of these challenges, the medical domain suffers from a severe scarcity of labeled CFD datasets: acquiring reliable diagnostic labels requires qualified medical expertise, data collection is constrained by privacy regulations, and CFD simulations on full nasal geometries are computationally expensive.

The geometries used for the CFD simulations are extracted from CT scans, which are segmented to isolate the internal air-filled regions of the upper airways (left-hand side of Figure 3.2). This process (described in Section 6.2) yields a

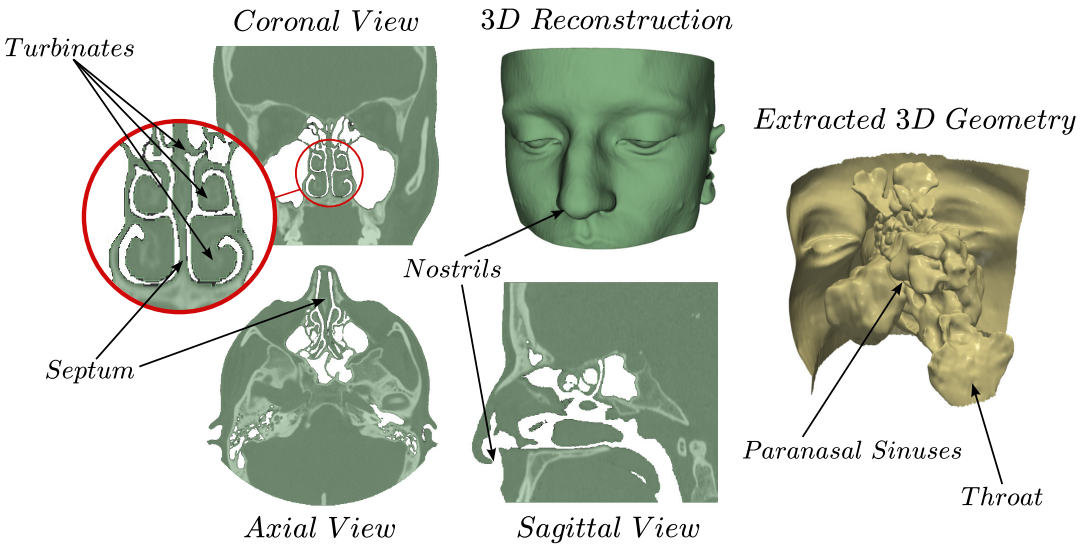
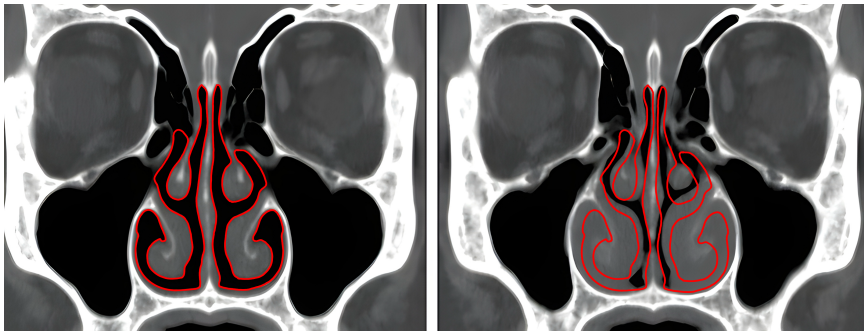


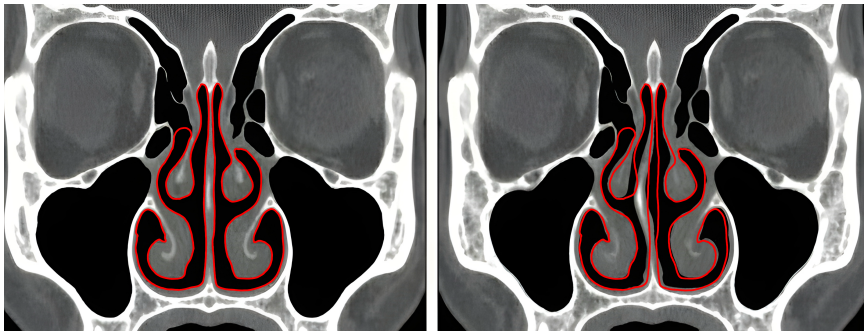
Figure 3.2: Overview of the geometry extraction pipeline. The left and center columns display the orthogonal CT slices (Coronal, Axial, Sagittal) together with the 3D skin reconstruction. The anatomical structures relevant for nasal airflow are visible in these slices: the *nasal septum*, which separates the two nasal fossae; the *turbinates*, lining the lateral walls and shaping the intranasal flow passages; and the *nostrils*, which form the inlet of the respiratory tract. The sagittal view further reveals the transition towards the *throat*. The right image shows the final segmented surface of the nasal airways used as the boundary for the CFD simulations, where the *paranasal sinuses*, present in the raw CT segmentation but not involved in the inspiratory flow, are also visible before being removed in the preprocessing stage.

detailed external surface of the nasal cavities (right-hand side of Figure 3.2), which serves as the computational boundary for the airflow simulations. During segmentation, the paranasal sinuses will be deliberately removed from the final computational surface. These structures play a negligible role in the inspiratory airflow yet introduce [8], irregular side chambers that significantly increase geometric complexity and mesh size. For this reason, they are excluded to obtain a streamlined and physiologically relevant domain for CFD analysis.

As shown in the orthogonal slices and the resulting 3D reconstruction in Figure 3.2, the extracted domain captures the intricate structures that govern airflow. The coronal and axial views highlight the nasal septum, which divides the cavity into two fossae, as well as the turbinates, the bony and mucosal elements lining the lateral walls. The final domain extends from the nostrils (inlet) to the throat (outlet), providing a comprehensive volumetric representation of the entire respiratory pathway relevant to inhalation.



(a) Turbinate Hypertrophy



(b) Septal Deviation

Figure 3.3: Nasal pathologies investigated in this study visualised through a CT scan. Panel (a) shows Turbinate Hypertrophy, where enlarged tissue obstructs the airway. Panel (b) illustrates a Septal Deviation, where the septum shifts laterally, creating a geometric imbalance in airflow distribution.

3.2.1 Task 3: Pathology Classification

The learning task addressed in this scenario is the classification of nasal pathologies based on the resulting CFD flow fields. We focus on two common structural conditions that significantly alter airflow distribution and resistance:

- **Turbinate Hypertrophy:** An abnormal enlargement of the mucosal tissue surrounding the turbinates. As illustrated in Figure 3.3a, the swollen tissue constricts the passage and increases flow resistance.
- **Septal Deviation:** A displacement of the nasal septum toward one side produces a geometric asymmetry inside the nasal cavity. Figure 3.3b shows how this deformation narrows one side while enlarging the other, frequently leading to unbalanced airflow and breathing difficulties.

The CT-based geometries used in this work come from a curated dataset provided by the *ASST Santi Paolo e Carlo* hospital in Milan. However, the

development of a supervised learning framework in this biomedical setting is hindered by an acute *Large-p, Small-n* configuration discussed in detail in Section 1.1: the dimensionality of each CFD sample is extremely high, whereas the number of clinically diagnosed pathological cases is intrinsically limited.

Due to the scarcity of labelled patient data, relying solely on real pathological anatomies would make the learning task intractable. To overcome this limitation, we employ the Data Augmentation strategy detailed in Chapter 4.

Chapter 4

Augmenting CFD Data through Computational Geometry

4.1 Motivation and Overview

One of the central challenges highlighted so far throughout the thesis is the severe scarcity of labelled data in CFD, particularly in applications involving complex geometries and high-fidelity simulations. As discussed in Chapters 2, 3, generating a single CFD sample entails substantial computational costs and often requires expert supervision, making the construction of large and diverse datasets impractical in many realistic settings. ML models in fact typically rely on a sufficiently large number of training samples to infer reliable input–output relationships, whereas in CFD the available datasets are often limited to a small number of simulations. This mismatch between the cost of data generation and the data requirements constitutes a fundamental obstacle in applying ML to CFD-driven inference problems. This chapter addresses the *small-n* regime of learning from CFD data, focusing on the limited number of labelled simulations available. Rather than attempting to directly reduce the dimensionality of the data, which is the subject in Chapter 5, here we tackle data scarcity by introducing a geometry-based data augmentation framework.

A natural response to data scarcity is *Data Augmentation* (DA) [99], a strategy that is widely adopted in many ML domains to artificially increase the effective size and diversity of the training set by generating additional samples that preserve the semantic label of the original data. In the context of CFD, however, the direct transfer of standard augmentation practices is problematic, as flow data are constrained by physical laws, boundary conditions, and domain geometry. As a result, augmentation strategies that are effective in purely data-driven settings cannot be applied directly to CFD flow fields without explicitly accounting for physical consistency. This suggests that, in CFD, effective augmentation should target those aspects of the problem that can be varied while

preserving both physical admissibility and semantic meaning.

In many practical CFD applications, especially in aerodynamics and biomedical flows, the dominant source of variability is not the flow itself, but the geometry of the computational domain. Small geometric modifications can induce large and highly nonlinear changes in the resulting flow fields, while preserving a clear semantic interpretation, such as the presence of a surface defect or an anatomical pathology. This observation motivates a different perspective on data augmentation for CFD: instead of perturbing the flow fields, we act on the *geometry* on which the flow is simulated.

The key idea underlying this chapter is that realistic and labelled CFD datasets can be expanded by generating new geometries through controlled and physically meaningful deformations, and by subsequently recomputing the corresponding flow fields. By operating in the geometry space, augmentation preserves the validity of the governing equations and boundary conditions, while allowing the generation of diverse samples with unambiguous labels. The central challenge then becomes how to define and transfer such geometric deformations across different shapes in a consistent and automated manner, without requiring case-by-case expert intervention. To address this, in this chapter, we employ tools from computational geometry and non-rigid shape correspondence. Starting from a reference geometry and a set of expert-defined deformation functions, we leverage spectral shape correspondence techniques to transfer these deformations to heterogeneous geometries. This enables the synthesis of large sets of geometrically diverse but semantically consistent CFD-ready domains, which can be used to significantly enrich the training data available for inference tasks.

The remainder of this chapter is organised as follows. Section 4.2 reviews the relevant background and related work on data augmentation strategies for CFD. Section 4.3 introduces the formulation of the data augmentation problem in the CFD setting. Section 4.4 describes the adopted augmentation methodology in detail, including how clinically meaningful deformation primitives are defined on the reference surface (Section 4.4.1), how the shape matching problem is addressed through dense intrinsic correspondence (Section 4.4.2), and how deformations are transferred to patient-specific anatomies to generate synthetic pathological geometries (Section 4.4.3). Section 4.5 presents the experimental setup, including the generation of the synthetic training data, the feature extraction procedure, and the model training strategy. Section 4.6 reports and discusses the experimental results, analysing generalisation performance on synthetic data and real patient-specific anatomies, as well as robustness with respect to pathology severity. Finally, Section 4.7 summarises the main findings of this chapter and discusses their implications within the context of the thesis.

4.2 Related Works and Background

DA is a fundamental component of modern ML, where it is routinely employed to mitigate data scarcity, improve generalisation, and increase robustness of learned models. In classical computer vision pipelines, DA typically relies on simple geometric or photometric transformations, such as rotations, cropping, mirroring, or noise injection [99]. These operations are effective because they preserve the semantic content of natural images while expanding the empirical training distribution. Such assumptions, however, do not carry over to CFD. CFD data are indeed structured physical fields that must satisfy governing equations, boundary conditions, and geometric constraints. As a result, naively applying image-based augmentation strategies to flow data often leads to physically inconsistent samples. For example, rotating or mirroring velocity or pressure fields without a corresponding transformation of the computational domain alters the physical problem and violates the Navier–Stokes equations. Similarly, injecting uncorrelated noise disrupts essential properties such as mass conservation and incompressibility, leading to flow fields that are not numerically admissible. These considerations make clear that DA in CFD cannot be treated as a purely statistical operation, but must explicitly account for physical consistency.

Several works have explored augmentation directly in the flow-field space, with the goal of generating additional CFD samples without rerunning computationally expensive simulations. A prominent direction relies on deep generative models. Wu et al. [111] proposed a Navier–Stokes GAN capable of synthesizing turbulent velocity fields while enforcing physical priors such as energy spectra and vorticity statistics. Related approaches employ Variational Autoencoders and physics-informed generative models to reconstruct or interpolate flow fields [40, 75], as well as convolutional neural networks for super-resolution and reconstruction of under-resolved turbulence data [39]. While these methods can generate statistically plausible flow realisations, they typically assume a fixed computational domain, which limits their applicability to canonical configurations such as channel flows or homogeneous turbulence.

A distinct line of work introduces perturbations within the flow space, explicitly leveraging known similarity principles or reduced representations of the dynamics. Abucide-Armas et al. [1] proposed a data augmentation strategy grounded in Reynolds-number similarity, enabling the generation of multiple synthetic realisations from a single simulation while preserving dynamic similarity. Other studies apply controlled perturbations to turbulence quantities [4] or inject physics-based constraints and forcing terms into the governing equations to induce diverse flow states [30]. Modal and spectral approaches further explore augmentation by perturbing reduced-order representations based on POD, DMD, or Fourier modes [38, 33, 102]. Despite their physical grounding,

all these approaches also operate under the implicit assumption of fixed geometries and boundary conditions. As a consequence, they are inherently limited to exploring variability within a single computational domain and remain unable to capture the dependence of CFD solutions on geometric variations.

As discussed throughout the thesis, in many realistic scenarios, geometric variability represents the primary source of variability in the CFD data. This motivates the development of augmentation strategies that operate directly in the space of geometries, where new samples are obtained by smoothly deforming the computational domain and subsequently recomputing the associated CFD solutions. A wide range of mesh deformation techniques has been developed to generate families of geometries while preserving mesh validity and quality. Classical mesh morphing approaches, including free-form deformation [82], elasticity-based deformation models [35], and radial basis function interpolation [9, 90], have been extensively used in aerodynamic design and shape optimisation. In biomedical modelling, statistical shape models offer a principled framework to capture population-level anatomical variability and to generate synthetic but realistic geometries [45, 17]. More recently, spectral and intrinsic shape representations have enabled dense correspondence between non-rigid and heterogeneous geometries. Functional Maps [80] and their extensions [91, 70] provide a compact and robust way to establish intrinsic mappings between shapes, facilitating deformation transfer and geometric harmonization across datasets. Related frameworks, including functional map networks [46], informative correspondence models [78], and deep functional map approaches [25], further improve robustness in the presence of large geometric variability. While these methods differ in how correspondences are estimated and regularised, ranging from purely spectral formulations to learning-based models, they all share the objective of establishing dense, intrinsic, and transferable mappings across collections of heterogeneous shapes.

In CFD-based learning pipelines, geometry-space augmentation can be particularly relevant when labels depend on structural or morphological properties rather than on flow statistics alone. This setting arises in aerodynamic shape classification, design exploration, and biomedical flows in patient-specific anatomies. Few studies have exploited geometric perturbations to improve the generalisation of surrogate models in aerodynamic contexts [108, 26], while biomedical investigations have used mesh morphing and parametric deformations to analyse flow behaviour under different anatomical conditions [15]. Despite the challenges associated with non-parametric shape variability, geometry-space augmentation remains one of the most effective strategies for expanding CFD datasets. By operating directly on the dominant source of variability, it preserves physical admissibility, maintains semantic consistency of the labels, and provides a principled foundation for the augmentation framework developed in this chapter.

4.3 Data Augmentation: Problem Formulation

Building on the general inference framework introduced in Section 1.3, we now specialise the problem formulation to the task of data augmentation. We consider a dataset

$$\mathcal{D} = \{(S_i, M_i, \mathbf{F}_i, Y_i)\}_{i=1}^{N},$$

where each sample consists of a geometry S_i, its associated computational mesh M_i, the corresponding CFD data matrix $\mathbf{F}_i$, and a target label Y_i. Obtaining each element of $\mathcal{D}$ requires high-fidelity CFD simulations and expert supervision, which naturally limits the dataset size N. Moreover, the large variability among the geometries S_i, especially in complex settings such as anatomical cavities or deformed aerodynamic components, makes it difficult for the available samples to adequately cover the underlying shape space. As a result, learning models trained on $\mathcal{D}$ are prone to overfitting and often exhibit poor generalisation to unseen configurations.

The goal of data augmentation is to enlarge $\mathcal{D}$ by generating additional, physically meaningful samples that reflect plausible geometric variations of the original ones. Let V denote the number of synthetic variants generated from each original geometry S_i. The augmented dataset can then be expressed as

$$\mathcal{D}^{\star} = \{(S^{\star}_{i,j}, M^{\star}_{i,j}, \mathbf{F}_{i,j}, Y_{i,j})\}_{i=1,\dots,N}^{j=1,\dots,V},$$

where $S^{\star}_{i,j}$ denotes the j-th geometry derived from S_i, $M^{\star}_{i,j}$ is the corresponding computational mesh, and $\mathbf{F}_{i,j}$ contains CFD data defined over $M^{\star}_{i,j}$.

Formally, the augmentation process can be described by an operator

$$\mathcal{A}_i : (S_i, M_i, \mathbf{F}_i, Y_i) \longrightarrow \{(S^{\star}_{i,j}, M^{\star}_{i,j}, \mathbf{F}_{i,j}, Y_{i,j}\}_{j=1}^{V},$$

whose objective is to generate new samples satisfying the following requirements:

$$\begin{cases} |\mathcal{D}^{\star}| \gg |\mathcal{D}|, \text{where } |.| \text{ denote the set cardinality} \\ S^{\star}_{i,j} \text{ and } M^{\star}_{i,j} \text{ are geometrically plausible and respect boundary constraints,} \\ Y_{i,j} \text{ is uniquely determined by the applied geometric modification.} \end{cases}$$

The central challenge is therefore the design of an augmentation operator $\mathcal{A}_i$ capable of producing diverse and physically realistic samples $(S^{\star}_{i,j}, M^{\star}_{i,j}, \mathbf{F}_{i,j})$ without requiring additional manual labelling. Such an operator must preserve the semantic meaning of the labels while ensuring that the augmented dataset remains representative of the underlying physical phenomena and effectively supports the learning of the inference model K.

4.4 Methodology

Section 4.3 formalised data augmentation as the construction of an operator $\mathcal{A}_i$ that expands a limited set of labelled samples into a larger collection while preserving physical admissibility and label consistency. Although the proposed framework is rather general, the discussion in this section focuses on the biomedical application introduced in Chapter 3, namely airflow in the human upper airways. This scenario represents the most challenging data regime addressed in this thesis: the number of available labelled samples is extremely limited, the geometries exhibit strong inter-subject variability, and the anatomical shapes cannot be described by low-dimensional parametric models. As a consequence, geometry-based synthesis is not merely beneficial but necessary.

The objective of this chapter is therefore to find a way to construct a large, realistic, and reliably labelled dataset of CFD flow fields starting from a limited set of healthy anatomical geometries. The proposed solution is illustrated in Figure 4.1 and formally summarised in Algorithm 1. At a high level, the method relies on defining pathological geometric alterations on a single reference anatomy and systematically transferring them to multiple healthy patient-specific geometries.

The starting point of the proposed pipeline is a database of CT scans provided by *ASST Santi Paolo e Carlo* in Milan, a medical institution involved in this research project. From this database, a subset of CT scans $\{T_i\}_{i=1,\dots,N}$ corresponding to patients diagnosed as healthy is identified by otolaryngology (ear, nose and throat, ENT) experts. For each selected scan T_i, a three-dimensional surface $\tilde{S}_i$ is extracted following the procedure detailed in Section 6.2.1, where the tilde $\tilde{\cdot}$ denotes that these surfaces contain spurious portions of the volume which will be removed. The resulting set of surfaces $\{\tilde{S}_i\}_{i=1,\dots,N}$ constitutes the geometric basis for the subsequent analysis.

In collaboration the same experts, a single *reference* surface $S^{\text{ref}} \subset \mathbb{R}^3$ is selected among the available anatomies in the provided database and extracted from the corresponding CT scan. This reference geometry is chosen to be representative of a nominal healthy nasal anatomy, both in terms of morphological regularity and absence of pathological alterations, and serves as the common template throughout the methodology. On this surface, a set Δ of V *surface deformation functions* $\Delta = \{(\delta_j, Y_j)\}_{j=1,\dots,V}$ is defined, as shown in the top-left part of Figure 4.1. Each deformation δ_j encodes a controlled morphological alteration associated with a specific pathology label Y_j, and transforms the reference surface S^{ref} into a pathological variant $S_j^{\text{ref},\star}$. From a conceptual standpoint, these deformations can be interpreted as inverse *virtual surgery* operations, whereby experts manually modify the healthy reference anatomy to reproduce clinically plausible pathological configurations.

The reference surface S^{ref}, the set of deformation functions $\{(\delta_j, Y_j)\}_{j=1,\dots,V}$,

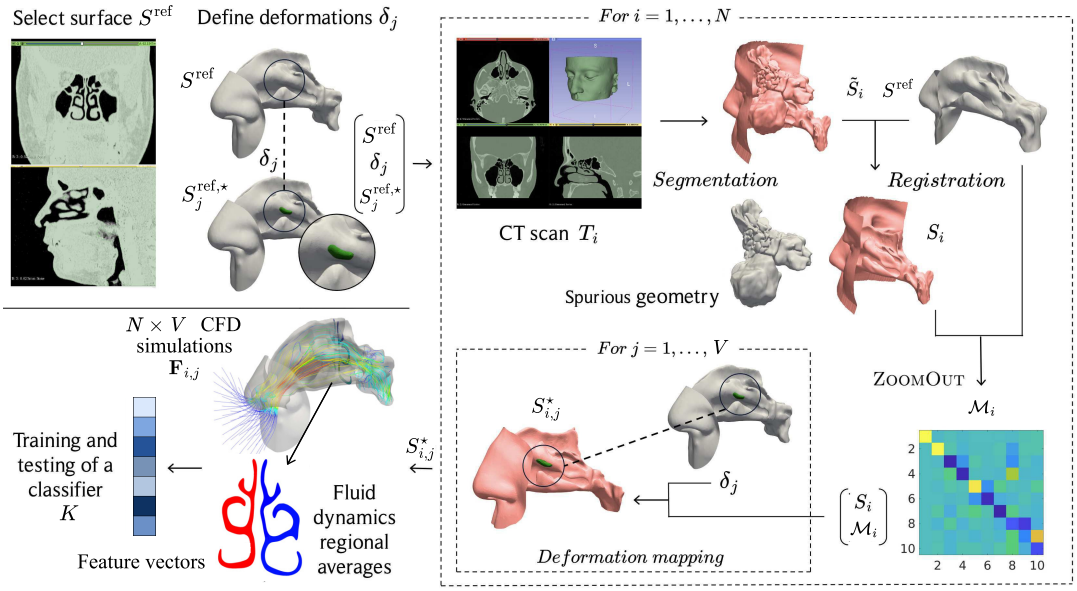


Figure 4.1: A schematic representation of the data augmentation method adopted in this thesis. At the top-left corner, we illustrate the two preliminary steps: selecting a reference nose S^{ref} and defining the collection of deformation functions $\{\delta_j\}_{j=1,\ldots,V}$. The deformation function δ_j transforms S^{ref} into a variant $S_j^{\text{ref},\star}$ affected by the pathology Y_j. On the right side of the figure, we show the procedure for generating the synthetic pathological surfaces $S^{\star}_{i,j}$. For each of the N CT Scans from healthy individuals T_i, we extract the surface $\tilde{S}_i$ by applying a threshold to the CT scan data. Then, using the ZoomOut algorithm [70], we register the reference S^{ref} over $\tilde{S}_i$ to remove from $\tilde{S}_i$ spurious parts (i.e., the paranasal sinuses), obtaining surface S_i. The bottom-right corner shows the compact representation of mapping $\mathcal{M}_i$. $\mathcal{M}_i$ associates at each point of S^{ref} a point of S_i and enables the transfer of functions δ_j from R to S_i producing $S^{\star}_{i,j}$, the variant of S_i affected by pathology y_j. Eventually, we assess the effectiveness of our method by running CFD simulations on each $S^{\star}_{i,j}$ and using CFD data to train and test a classifier, as shown in the bottom-left corner.

and the set $\{\tilde{S}_i\}_{i=1,\ldots,N}$ of healthy geometries are then used to generate a collection of annotated anatomies. The key assumption underlying this step is that each healthy patient-specific anatomy can be regarded as a non-rigid deformation of the reference surface S^{ref}. Based on this observation, functional maps [80] are adopted to estimate dense point-to-point correspondences between S^{ref} and each $\tilde{S}_i$. These correspondences serve two distinct purposes within the pipeline. First, they are used to preprocess each surface $\tilde{S}_i$ by automatically removing non-essential anatomical regions, i.e., the paranasal sinuses, yielding a cleaned

surface S_i suitable for CFD simulations (Algorithm 1, line 9; top-right corner in Figure 4.1). Second, the same mappings enable the transfer of each deformation function δ_j from the reference surface S^{ref} to the cleaned patient-specific surface S_i (Algorithm 1, lines 10–17; bottom-right corner in Figure 4.1), resulting in a synthesized pathological geometry $S^{\star}_{i,j}$, which represents a variant of S_i affected by the pathology Y_j. By repeating this procedure for all healthy baseline geometries and all deformation functions, a labelled set of surfaces $\{S^{\star}_{i,j}\}_{i=1,\dots,N;\, j=1,\dots,V}$ is obtained. CFD simulations are then performed on each synthesized geometry $S^{\star}_{i,j}$ to produce the corresponding flow fields $\{\mathbf{F}_{i,j}\}_{i=1,\dots,N;\, j=1,\dots,V}$, which collectively form the training set of the classifiers considered in this work.

The proposed method ensures diversity in the generated CFD dataset by leveraging the anatomical variability of healthy individuals, while simultaneously guaranteeing unambiguous and automatically defined labels through explicit control of the applied deformations. Moreover, the framework naturally supports the combination of multiple pathologies and the modulation of their severity by adjusting the intensity of the deformation functions, thereby substantially increasing the size and expressiveness of the training set.

The individual components of the pipeline are discussed in detail in the following subsections. In particular, Sections 4.4.1 describes the selection of the reference anatomy and the definition of deformation primitives. Sections 4.4.2 introduces the shape correspondence problem, complemented in Appendix A. Section 4.4.3 details ow deformations are transferred from the reference to healthy geometries, while Section 4.4.4 briefly presents the CFD simulation setup adopted for the synthesised geometries and described in Chapter 6.

4.4.1 Reference Surface S^{ref} and Deformation Primitives

A central component of the proposed augmentation framework is the definition of a common reference surface, denoted by $S^{\text{ref}} \subset \mathbb{R}^3$. The role of S^{ref} is to provide a common geometric domain on which semantic deformation primitives can be defined independently of any specific subject anatomy. This choice enables a clear separation between the conceptual definition of a morphological alteration and its instantiation on individual geometries, which is essential for scalable data augmentation in non-parametric settings.

The reference surface S^{ref} is selected to represent a nominally *healthy* anatomy and is required to satisfy several practical conditions. First, it must be free of topological artefacts, self-intersections, and segmentation errors, as such defects would propagate through the correspondence and deformation stages. Second, the surface should exhibit a high-quality discretisation, with sufficient resolution to capture anatomically relevant structures while avoiding unnecessary geometric noise. Finally, S^{ref} should be representative of the population under study, so that deformations defined on it can be meaningfully transferred

Algorithm 1 Data augmentation and feature extraction pseudo code

Input:
Reference surface S^{ref}
Deformation set $\{(\delta_j, Y_j)\}_{j=1}^{V}$
CT scans $\{T_i\}_{i=1}^{N}$
Output:
Feature vectors $\mathbf{P}_{i,j}$ with labels Y_j
for $i = 1, \ldots, N$ **do** ▷ Process each CT scan
    Extract surface $\tilde{S}_i \leftarrow \text{Segment}(T_i)$
    Preprocess geometry $S_i \leftarrow \text{Register}(\tilde{S}_i, S^{\text{ref}})$
    Compute correspondence $\mathcal{M}_i : S^{\text{ref}} \to S_i$
    for $j = 1, \ldots, V$ **do** ▷ Apply each deformation δ_j
        Generate augmented surface

$$S^{\star}_{i,j} \leftarrow S_i + \delta_j(\mathcal{M}_i^{-1}(S_i))$$

        Run CFD simulation $\mathbf{F}_{i,j} \leftarrow \text{CFD}(S^{\star}_{i,j})$
        Extract transversal sections from $\mathbf{F}_{i,j}$.
        Compute regional averages of $|\boldsymbol{U}|$, $|\nabla p|$, E, k_t
        Assemble feature vector $\mathbf{P}_{i,j}$
    end for
end for

to heterogeneous target geometries.

S^{ref} is chosen by ENT experts among the available CT scans in the database provided by *ASST Santi Paolo e Carlo*, and segmented as detailed in Section 6.2.1. The selected surface is subsequently *manually* corrected and simplified by the same experts through an intensive and time-consuming procedure, which involves the careful removal of anatomical regions that are unnecessary for CFD simulations [8, 49], most notably the paranasal sinuses shown in Figure 4.3a.

After that, on S^{ref} a set of deformation primitives is defined by the same ENT experts who identified S^{ref},

$$\Delta = \{(\delta_j, Y_j)\}_{j=1}^{V},$$

where each deformation $\delta_j : S^{\text{ref}} \to \mathbb{R}^3$ is a vector field encoding a controlled geometric modification, and $Y_j \in \mathcal{Y}$ denotes the semantic label associated with that modification. Each deformation primitive is obtained by manually transforming the healthy reference surface into a pathological variant, denoted as $S_j^{\text{ref},\star}$, and by tracking the displacement of all surface vertices between the healthy configuration and its pathological counterpart, so that for each point $\mathbf{x} \in S^{\text{ref}}$,

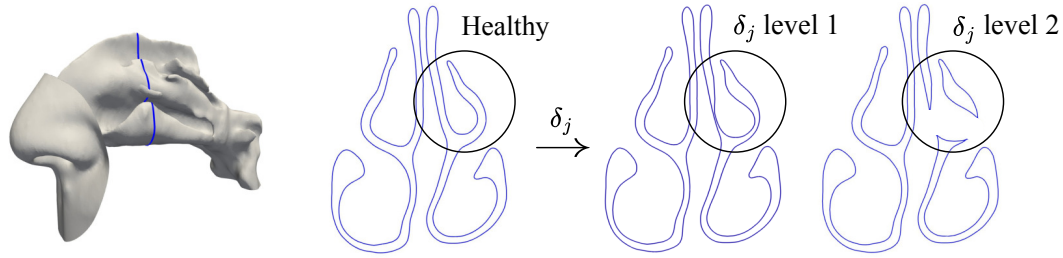


Figure 4.2: Example of deformation functions defined on the reference surface S^{ref}. A sectional view is shown to visualize the geometric modification. Starting from the healthy reference configuration (center), a turbinate hypertrophy is introduced by domain experts with two different severity levels (right), corresponding to the same semantic label Y. The pathological condition and its clinical relevance are discussed in detail in Chapter 3.

$\delta_j(\mathbf{x})$ represents the vector displacement prescribed by the expert. In this sense, each deformation primitive explicitly encodes how a specific pathology alters the anatomy in a localised and anatomically meaningful manner.

Applying a deformation primitive δ_j to the reference surface produces a synthetic pathological variant

$$S_j^{\text{ref},\star} = \left\{ \mathbf{x} + \delta_j(\mathbf{x}) \;\middle|\; \mathbf{x} \in S^{\text{ref}} \right\}. \tag{4.1}$$

From a conceptual standpoint, the definition of deformation primitives on a single reference surface can be interpreted as an *inverse virtual surgery*: rather than correcting a pathological anatomy, the expert deliberately introduces controlled pathological alterations on a healthy patient. This strategy drastically reduces the manual annotation burden, as domain experts are not required to independently model pathologies on each subject-specific geometry.

The deformation fields δ_j result in smooth, spatially localised, and anatomically plausible displacements. Smoothness ensures that the resulting geometries remain suitable for volumetric meshing and subsequent CFD simulations, while localisation allows multiple primitives to be combined without inducing unrealistic global distortions. The magnitude of each deformation could, in fact, be modulated through scalar coefficients, allowing explicit control over the severity of the simulated condition. An important property of this formulation is in fact its modularity. Because deformation primitives are defined independently on S^{ref}, multiple primitives can be composed to synthesise more complex morphological configurations, including anatomies exhibiting coexisting alterations. Crucially, the semantic label Y_j associated with each deformation primitive is known by construction and does not require additional manual annotation when the deformation is transferred to other geometries.

The next step of the pipeline consists in transferring the deformation primitives from the reference surface to each healthy geometry. However, surfaces extracted directly from CT scans, which we denote as $\tilde{S}_i$, include the same anatomical structures that were manually removed by domain experts when defining the reference surface S^{ref}, most notably the paranasal sinuses. Accordingly, each extracted surface $\tilde{S}_i$ is first processed through a geometry cleaning step aimed at removing exactly these regions, yielding a simplified subject-specific surface S_i that is anatomically consistent with S^{ref}.

Only after this preprocessing step can deformation primitives be meaningfully transferred to subject-specific geometries. A critical implication of this formulation is that deformation primitives δ_j are defined on the reference surface S^{ref}, while the geometries to be augmented are the cleaned subject-specific surfaces S_i. In order to apply a deformation defined on S^{ref} to a target geometry S_i, it is therefore necessary to establish a correspondence between the two surfaces. Without such correspondence, there is no principled way to determine where a given deformation should be applied on S_i, and even small misalignments may result in anatomically implausible modifications or loss of semantic meaning associated with the label Y_i.

4.4.2 Shape Matching: Cleaning $\tilde{S}_i$ and Matching with S^{ref}

Starting from the subject-specific surfaces $\tilde{S}_i$ extracted from CT scans, the objective in this section is to establish a dense mapping with the reference surface S^{ref}. Such a mapping is required to consistently transfer deformation primitives defined on S^{ref} to subject-specific geometries. To this end, two subsequent problems must be addressed: the automatic removal of the paranasal sinuses from $\tilde{S}_i$ to get the cleaned geometry S_i which is topologically similar to S^{ref}, and the estimation of a point-to-point correspondence between the resulting surface S_i and the reference anatomy S^{ref} to transfer each δ_j.

Cleaning $\tilde{S}_i$ via clutter-aware registration. As discussed in Section 4.4.1, surfaces $\tilde{S}_i$ extracted from CT scans include the paranasal sinuses, which are not relevant for CFD simulations. While these regions are manually removed from the reference surface S^{ref} by medical experts, performing the same manual operation for each subject-specific surface would be time-consuming, costly, and not scalable. An automatic cleaning procedure is therefore required to process large collections of anatomies without repeated expert intervention.

The key idea is to clean the surface $\tilde{S}_i$ by isolating the portion that corresponds anatomically to a reference surface S^{ref}. This is achieved through a

(a) Surface extracted from CT data, including paranasal sinuses highlighted in green.

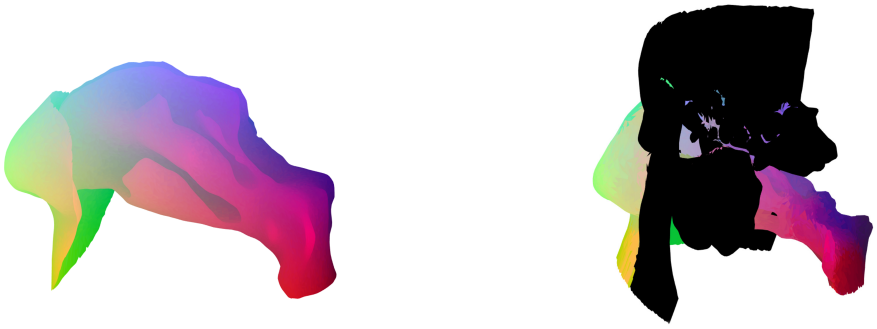
(b) Target surface after removal of paranasal sinuses.

Figure 4.3: Example of geometric clutter arising from CT-based surface extraction. The complete surface (left) includes paranasal sinuses and secondary cavities, which are not present in the reference surface S^{ref}. These structures must be excluded in order to establish a meaningful correspondence. The cleaned surface (right) represents the effective domain used for shape correspondence and deformation transfer.

shape-registration framework based on functional correspondence. In particular, we rely on the method introduced by Cosmo et al. [22], originally developed for deformable object recognition and dense correspondence in cluttered 3D scenes.

The registration algorithm takes as input the target surface $\tilde{S}_i$, which includes the paranasal sinuses (highlighted in green in Figure 4.3), and the reference surface S^{ref}, which does not contain these structures. Within this formulation, the paranasal sinuses are treated as clutter. A robust non-rigid registration procedure is then employed to estimate a point-to-point correspondence between S^{ref} and the subset of $\tilde{S}_i$ that represents the same anatomical region. More details can be found in Appendix A. As a result of the registration, we identify a sub-surface of $\tilde{S}_i$, denoted by S_i, which best matches the geometry of S^{ref}. This selection is encoded through a binary segmentation mask defined on the vertices of $\tilde{S}_i$: vertices that are successfully matched to a point on S^{ref} are assigned a value of 1, while unmatched vertices, corresponding to regions to be discarded, are assigned a value of 0. Since S^{ref} acts as the source shape in the registration, all its vertices are preserved. Conversely, the mask on $\tilde{S}_i$ explicitly identifies the vertices that must be removed in order to enforce geometric consistency with S^{ref}.

Figure 4.4 illustrates the outcome of this procedure. Vertices with null mask



(a) Reference surface S^{ref}. (b) Target surface S_i^{raw}.

Figure 4.4: Correspondence estimation in a registration-in-clutter setting. The reference surface S^{ref} (left) matches only a subset of the target surface $\tilde{S}_i$ (right). The colored region identifies the subset retained for correspondence, with colours indicating matching points across the two surfaces. Vertices shown in black are assigned a zero mask value, are excluded from the correspondence estimation, and removed to obtain the cleaned surface S_i used in subsequent deformation transfer.

value are shown in black, indicating regions excluded from the cleaned geometry, whereas vertices retained in S_i are colored according to their correspondence with S^{ref}.

This clutter-aware registration step enforces anatomical consistency between S^{ref} and S_i and provides a reliable identification of the corresponding anatomical regions. However, since its primary objective is object recognition in cluttered scenes, the resulting correspondence is not optimised for high local accuracy at the vertex level, and achieving such accuracy through direct refinement would be computationally expensive. For this reason, a subsequent correspondence estimation step is required to compute a dense and highly refined point-to-point mapping $\mathcal{M}_i$, which is addressed in the following paragraph using the ZoomOut algorithm.

Dense non-rigid correspondence via ZoomOut. To estimate a more refined mapping $\mathcal{M}_i$ between S_i and S^{ref}, we rely on the ZoomOut algorithm proposed by Melzi et al. [70]. This choice is motivated by the fact that, after the clutter-aware cleaning stage, the two surfaces represent the same anatomical structure and share the same topological organisation. In practical terms, this means that corresponding regions on S^{ref} and S_i are connected in the same way, even though their exact geometry, proportions, and discretisation may differ. This property allows ZoomOut to start from a coarse correspondence that is globally consistent across the surface and to progressively improve its spatial accuracy without introducing mismatches between distant anatomical regions.

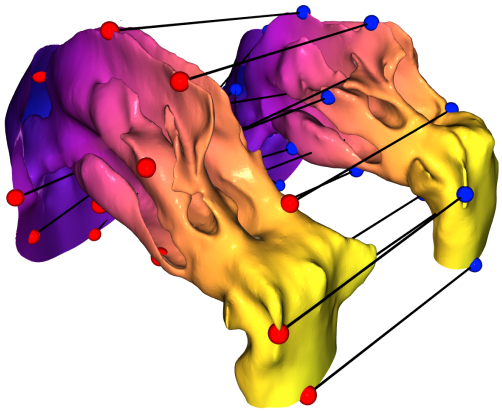


Figure 4.5: Visualization of the pointwise correspondence $\mathcal{M}_i$ between the reference surface S^{ref} (left) and a target surface S_i (right). Colored markers indicate corresponding vertices, while black segments connect matched points in $\mathbb{R}^3$. Only a subset of correspondences is shown for visualisation purposes.

To this end, the algorithm incrementally increases the spectral resolution of the functional representation and alternates between functional and pointwise mappings, thereby refining the spatial localisation of the correspondence. Rather than directly matching vertices in Euclidean space, which is unreliable under large non-rigid deformations, ZoomOut constructs the correspondence in an intrinsic spectral domain, enabling stable and efficient computation even for high-resolution meshes.

The input to this stage consists of the reference surface S^{ref}, the cleaned target surface S_i, and an initial coarse mapping, while the output is a dense correspondence map

$$\mathcal{M}_i : S^{\text{ref}} \to S_i$$

which associates each vertex of the reference surface with its anatomically corresponding location on the subject-specific geometry. This correspondence constitutes the key enabling component for the deformation transfer process described in the next subsection. The resulting pointwise correspondence $\mathcal{M}_i$ can be visualised as a sparse set of anatomically consistent point-to-point associations between the reference surface S^{ref} and the target surface S_i, as illustrated in Figure 4.5. Additional algorithmic and mathematical details are provided in Appendix A.

4.4.3 Deformation Transfer via Functional Correspondence

Once the mapping $\mathcal{M}_i$ between the reference surface S^{ref} and a target geometry S_i has been established, deformation primitives defined on the reference can be transferred to subject-specific geometries. This step represents the core mechanism enabling geometry-based data augmentation in the proposed framework.

Let $\delta_j : S^{\text{ref}} \to \mathbb{R}^3$ denote a deformation primitive defined on the reference surface, associated with a semantic label Y_j. Given the pointwise correspondence $\mathcal{M}_i : S^{\text{ref}} \to S_i$ introduced in Section 4.4.2, the deformation can be transported to the target surface by reparameterizing δ_j through the inverse mapping $\mathcal{M}_i^{-1}$. For a vertex $v \in S_i$, the transferred deformation is defined as

$$\delta_{i,j}(v) = \delta_j\Big(\mathcal{M}_i^{-1}(v)\Big)\,. \tag{4.2}$$

The synthetic pathological geometry $S_{i,j}^{\star}$ is then obtained by displacing each vertex of the target surface according to the transferred deformation field,

$$S_{i,j}^{\star} = \{v + \delta_{i,j}(v) \mid v \in S_i\}\,. \tag{4.3}$$

This formulation ensures that the deformation is applied at anatomically corresponding locations on the target geometry, independently of inter-subject variability in shape or scale.

An important property of this construction is the preservation of semantic meaning: because deformation primitives are defined on the reference surface and associated with a known label Y_j, the same label is automatically propagated to the synthesized geometry $S_{i,j}^{\star}$. The correspondence-based transfer guarantees that the deformation remains localised to the intended anatomical region, thereby maintaining consistency between geometry and semantic annotation across subjects.

Also, this framework naturally supports the composition of multiple deformation primitives. Given a subset of primitives $\{\delta_k\}_{k=1}^{K}$, a composite deformation can be expressed as

$$\delta_i^{\text{comp}}(v) = \sum_{k=1}^{K} \alpha_k\, \delta_k\Big(\mathcal{M}_i^{-1}(v)\Big)\,, \tag{4.4}$$

where $\alpha_k \in \mathbb{R}$ are scalar coefficients controlling the severity of each deformation. This linear composition allows the synthesis of complex pathological scenarios while retaining explicit control over their semantic interpretation.

From a numerical standpoint, the deformation transfer process must preserve the quality and validity of the underlying surface mesh. This is ensured by two key factors: the smoothness of the deformation fields δ_k and the regularity of the correspondence $\mathcal{M}_i$. Each deformation δ_k is defined as a smooth

displacement field on the reference surface and is transferred to S_i through a consistent point-to-point mapping. As a result, neighbouring vertices undergo similar displacements, preventing abrupt geometric distortions or element inversions. Importantly, the deformation is applied directly to an existing and validated surface mesh S_i, without altering its connectivity. This guarantees topological consistency and avoids the need for costly remeshing operations of the surface.

The deformation transfer process described above is purely geometric. Once the synthetic surface $S^\star_{i,j}$ has been generated, it is used as a boundary condition for subsequent CFD simulations, which are automatically annotated without additional manual intervention. Overall, this approach enables the generation of large sets of anatomically consistent and semantically labelled geometries starting from a limited number of healthy baselines. This step completes the geometry-based data augmentation introduced in this Chapter and enables the construction of an enriched dataset of geometries with respect to the starting one.

4.4.4 From Augmented Geometry to CFD Data

The pipeline described in Section 4.3 so far produces a set of synthetic, anatomically consistent surfaces $\{S^\star_{i,j}\}$ obtained by transferring deformation primitives to healthy baseline geometries. The purpose of this section is to briefly describe how these augmented geometries are converted into CFD data suitable for subsequent learning tasks.

Each synthetic surface $S^\star_{i,j}$ defines the boundary of a three-dimensional flow domain. Starting from this surface representation, a volumetric computational mesh is generated using standard meshing procedures, ensuring adequate resolution near the walls and in regions of expected high flow gradients. Because the deformation transfer preserves the topology and overall regularity of the original surface mesh, volumetric mesh generation can be performed robustly without requiring manual intervention or ad hoc corrections.

Once the volumetric mesh has been generated, high-fidelity CFD simulations are performed by solving the governing flow equations under fixed boundary and operating conditions. In the biomedical scenario considered in this thesis, simulations are carried out within a large-eddy simulation (LES) framework. Details on the numerical setup, solver configuration, and validation procedures are provided in Chapter 6.

The output of this stage is the augmented dataset of paired samples

$$\mathcal{D}^\star = \{(S^\star_{i,j}, \mathbf{F}_{i,j}, Y_j)\}_{i=1,\dots,N;j=1,\dots,V}$$

where $\mathbf{F}_{i,j}$ denotes the simulated flow field associated with the augmented geometry $S^\star_{i,j}$, and Y_i its label. This dataset forms the basis for the learning-based

analyses presented in the next Sections, where the impact of geometry-based data augmentation on model performance and generalisation is systematically evaluated.

4.5 Experiments

This section presents the experimental evaluation of the proposed geometry-based data augmentation framework, with the objective of assessing its effectiveness in supporting supervised ML from CFD-derived data under severe data scarcity. All experiments are conducted by training ML models exclusively on flow fields $\{\mathbf{F}_{i,j}\}$ computed on synthetically generated pathological geometries $\{S^{\star}_{i,j}\}$ obtained from CT scans of healthy subjects through the augmentation strategy proposed in Section 4.3.

In line with the study presented in [97], but extending it to anatomically realistic and non-parametrised nasal geometries, flow fields are not used directly as model inputs. Instead, compact feature vectors are extracted by computing regional averages of CFD quantities on a set of anatomically meaningful transversal sections, defined consistently across all geometries. The experimental analysis is structured around two complementary evaluation settings. First, model performance is assessed on synthetic data only, using a Leave-One-Patient-Out cross-validation (LOPO-CV) scheme that enforces subject-level separation between training and testing samples. Second, classifiers trained exclusively on synthetic data are evaluated on an independent set $\mathcal{D}_{\text{real}}$ of real pathological patients, never seen during training. This second setting represents the most challenging and clinically relevant scenario, as it directly probes the ability of the proposed augmentation strategy to bridge the gap between synthetic CFD-derived data and real patient-specific anatomies.

In addition to these primary evaluations, further experiments are designed to analyse the robustness and consistency of the learned models. In particular, we investigate how training on synthetic data with increasing pathology severity and more diverse deformation combinations affect generalisation to real patients. Moreover, we assess whether classifier predictions remain consistent when pathologies in real patient geometries (set $\mathcal{D}_{\text{real}}$) are artificially strengthened (set $\mathcal{D}_{\text{boost}}$) via deformation transfer, providing an additional sanity check on the physical plausibility of the learned decision rules.

Finally, model interpretability is addressed through a SHAP-based analysis [64], which is employed to identify the most informative CFD features and anatomical regions contributing to pathology classification. This analysis complements quantitative performance metrics by providing insights into how complex flow-derived patterns are exploited by the models, thereby supporting an interpretation of the experimental results.

4.5.1 Training Set Generation

The synthetic training set is generated starting from a small set of healthy subjects. Specifically, surfaces $\{S_i\}_{i=1}^{N}$ are extracted from CT scans of $N = 7$ healthy individuals, all diagnosed as pathology-free by ENT specialists. Each surface is preprocessed as described in Section 4.4.2, yielding geometries that are topologically consistent with the reference surface S^{ref}.

A set of deformation primitives

$$\Delta = \{(\delta_j, Y_j)\}_{j=1}^{V}$$

is defined on S^{ref} (Section 4.4.1), covering two clinically relevant nasal pathologies: septal deviation and turbinate hypertrophy, described in Section 3.2. For each pathology, deformation primitives are designed by ENT experts by varying both the anatomical location and the severity of the deformation: for septal deviation, 6 deformation functions are defined, corresponding to different deviation locations and two severity levels. For turbinate hypertrophy, 9 deformation functions are defined, accounting for different turbinate locations and severity levels. In addition to single-pathology cases, deformation primitives corresponding to the same pathology but different locations or severity levels are linearly combined, yielding 12 distinct configurations for septal deviation and 17 for turbinate hypertrophy. This design allows the generation of a diverse yet controlled set of pathological configurations, while preserving a clear semantic association between geometry and label Y_i.

As outlined in Section 4.3, each deformation δ_j is transferred to all healthy surfaces S_i via the correspondence map $\mathcal{M}_i$, producing a set of synthetic pathological geometries

$$\{S_{i,j}^{\star}\}_{i=1,\dots,N;j=1,\dots,V},$$

with unambiguous labels Y_j. By applying all deformation instances and their combinations to the original $N = 7$ healthy anatomies, the initial set is expanded into a dataset of 308 synthetic pathological geometries. Healthy geometries are not included in the classification task, as the focus of the experiments is on discriminating among pathological conditions.

To explicitly organise the datasets used for training and evaluating the classifiers, we introduce a compact notation for the subsets derived from $\{S_{i,j}^{\star}\}$. We denote by S_1 and S_2 the subsets containing geometries affected by a single pathology with severity level 1 and 2, respectively, and define their union as $S_{12} = S_1 \cup S_2$. For notational simplicity, we further denote by S the complete dataset, which extends S_{12} by including synthetic samples obtained from combinations of different deformation instances and severity levels. Table 4.1 summarises all datasets considered in the experiments, distinguishing between synthetically generated samples and those directly derived from CT scans, and further categorising them by pathology type and severity level.

			Dataset Name					
			$S1$	$S2$	$S12$	S	$\mathcal{D}_{\text{real}}$	$\mathcal{D}_{\text{boost}}$
Samples	Hyp	Lev 1	35	-	35	182	5	5
		Lev 2	-	28	28			
	Dev	Lev 1	21	-	21	126	5	5
		Lev 2	-	21	21			
	Tot	Lev 1	56	-	56	308	10	10
		Lev 2	-	49	49			
		Tot	56	49	105			

Table 4.1: Summary of the sets we use in the experiments, differentiating between hypertrophies (Hyp) and septal deviations (Dev), as well as the severity level of pathologies, indicating severity 1 as Lev 1 and severity 2 as Lev 2. At the bottom, the table reports the total number of samples of each set, discriminating also between Lev 1 and Lev 2 for the synthetically generated sets. We do not highlight this distinction for S as it extends $S12$ with synthetic samples carrying combinations of different severity levels.

4.5.2 CFD Simulations and Feature Extraction

For each augmented geometry $S^{\star}_{i,j}$, a high-fidelity CFD simulation is performed following the numerical setup described in Chapter 6. All simulations are carried out under identical boundary and operating conditions, so that variations in the resulting flow fields are solely attributable to geometry-induced effects. The resulting CFD solutions are denoted as

$$\mathbf{F}_{i,j} : \Omega_{i,j} \to \mathbb{R}^d,$$

where $\Omega_{i,j}$ denotes the volumetric flow domain and d the number of simulated physical quantities, as described in Section 1.3.

Due to the extremely high dimensionality of CFD data, compact and interpretable feature representations are extracted using a section-based strategy inspired by previous work on inverse aerodynamic inference and biomedical CFD analysis [97]. Specifically, a set of 6 anatomically meaningful transversal sections $\{\Sigma_s\}_{s=1}^{6}$ is defined along the nasal cavity. Following [97], section locations are obtained by uniformly sampling the centerline of the nasal airway between the beginning and the end of the olfactory region. Each section $\Sigma_s \subset \Omega_{i,j}$ is obtained by intersecting the flow domain with a plane orthogonal to the local centerline direction, yielding a two-dimensional cross-section of the airway. To capture left–right asymmetries induced by nasal pathologies, each section Σ_s is

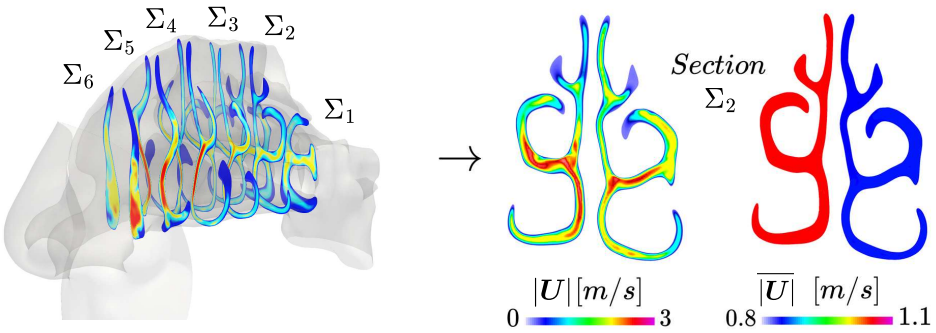


Figure 4.6: Transversal sections used for feature extraction. The left panel shows the spatial locations of the $N_s = 6$ two-dimensional sections $\{\Sigma_s\}$ defined along the nasal cavity. Each section is partitioned into left and right regions $(R_s^{\mathrm{L}}, R_s^{\mathrm{R}})$. The right panel illustrates the computation of regional average features by aggregating the velocity magnitude over the cells intersecting each region.

further partitioned into two disjoint regions,

$$\Sigma_s = R_s^{\mathrm{L}} \cup R_s^{\mathrm{R}}, \qquad R_s^{\mathrm{L}} \cap R_s^{\mathrm{R}} = \emptyset,$$

corresponding to the left and right portions of the nasal cavity.

Let $q(\mathbf{x})$ denote a scalar flow quantity of interest (e.g., velocity magnitude, velocity components or pressure). For each section Σ_s and region R_s^r, with $r \in \{\mathrm{L}, \mathrm{R}\}$, the regional feature is computed as a discrete area-weighted average over the set of CFD cells intersecting the region. Denoting by $\mathcal{C}(R_s^r)$ the set of mesh cells whose centroids lie in R_s^r, the feature is defined as

$$\bar{q}_{i,j}^{s,r} = \frac{\sum\limits_{c \in \mathcal{C}(R_s^r)} q_c \, A_c}{\sum\limits_{c \in \mathcal{C}(R_s^r)} A_c}, \tag{4.5}$$

where q_c denotes the cell-averaged value of the quantity q in cell c weighted by the area of the cells A_c to account for their uneven dimension. This formulation corresponds to a discrete approximation of a sectional average and is consistent with the numerical nature of the CFD data.

By repeating this aggregation for a set of m flow quantities $\mathcal{Q} = \{q^{(1)}, \ldots, q^{(m)}\}$ across all sections and regions, a fixed-length feature vector is obtained,

$$\mathbf{P}_{i,j} = \left[\bar{q}_{i,j}^{1,\mathrm{L}}, \bar{q}_{i,j}^{1,\mathrm{R}}, \ldots, \bar{q}_{i,j}^{N_s,\mathrm{L}}, \bar{q}_{i,j}^{N_s,\mathrm{R}}\right] \in \mathbb{R}^{12 \times m}. \tag{4.6}$$

This representation preserves the spatial ordering of anatomical sections and explicitly encodes left–right asymmetries, which are central to the characterisation of nasal pathologies. At the same time, discrete regional averaging improves

robustness with respect to mesh resolution and numerical noise, while retaining sensitivity to clinically relevant airflow alterations. The same feature extraction pipeline is applied consistently to both synthetic samples and real patient data, ensuring a fair evaluation of generalisation performance. It is worth noting, however, that for the 7 healthy subjects and for patients in $\mathcal{D}_{\text{real}}$, the identification of the transversal sections Σ_s requires the manual annotation of the beginning and end of the olfactory region by domain experts. In contrast, for synthetic samples the section locations are inherited directly from the corresponding original healthy anatomies, since synthetic geometries are generated through deformation transfer and preserve the anatomical parametrisation of their source surfaces.

Considered Flow Quantities The regional averages introduced in Eqs. (4.5)–(4.6) are computed for a set of flow quantities

$$\mathcal{Q} = \{|\boldsymbol{U}|,\ |\nabla p|,\ E,\ k_t\}\,,$$

where $|\boldsymbol{U}|$ denotes the magnitude of the velocity field, $|\nabla p|$ the pressure gradient magnitude, E the resolved enstrophy, and k_t the resolved turbulent kinetic energy. These quantities summarise complementary aspects of the flow field and therefore encode diverse information about the underlying airway anatomy and its impact on airflow dynamics [97].

The velocity magnitude $|\boldsymbol{U}|$ is closely related to geometric properties of the airway. For an incompressible flow within a duct, the volumetric flow rate Q is conserved along the streamwise direction. As a consequence, the mean velocity over a transversal section is inversely proportional to the cross-sectional area. Denoting by A_s^r the area of region R_s^r, this quantity is computed as

$$\overline{|\boldsymbol{U}|}_{s,r} = \frac{1}{A_s^r} \sum_{c \in R_s^r} |\boldsymbol{U}_c|\, A_c = \frac{Q}{A_s^r} \tag{4.7}$$

where the sum runs over the CFD cells intersecting the region and A_c denotes the cell area. For a fixed volumetric flow rate Q, variations in $\overline{|\boldsymbol{U}|}_{s,r}$ therefore directly reflect local changes in the effective cross-sectional area, making this quantity sensitive to geometric constrictions such as those induced by septal deviation or turbinate hypertrophy.

The magnitude of the pressure gradient $|\nabla p|$ provides information related to flow acceleration and local resistance. Regions affected by pathological narrowing typically exhibit steep pressure gradients, reflecting abrupt changes in flow direction and cross-sectional area.

The resolved enstrophy E is defined as the squared magnitude of the vorticity associated with the resolved velocity field,

$$E = |\boldsymbol{\Omega}|^2\,, \qquad \boldsymbol{\Omega} = \nabla \times \boldsymbol{U}. \tag{4.8}$$

This quantity captures information related to resolved velocity gradients and rotational structures in the flow. *Resolved* refers to the fact that the CFD simulations are performed using a LES approach, in which the smallest turbulent scales are filtered out. As a consequence, E captures the contribution of velocity gradients associated with flow structures that are explicitly resolved at the numerical scale of the simulation. Regions characterised by strong shear layers or vortical motion typically exhibit high values of E. A detailed description of the LES formulation and filtering strategy is provided in Chapter 6.

The resolved turbulent kinetic energy k_t is defined as half the sum of the variances of the velocity fluctuations with respect to the mean velocity field,

$$k_t = \frac{1}{2}\left[(u_x - \overline{u}_x)^2 + (u_y - \overline{u}_y)^2 + (u_z - \overline{u}_z)^2\right], \tag{4.9}$$

where $\overline{\boldsymbol{U}}$ denotes the mean velocity field. The quantity k_t represents the resolved part of the turbulent kinetic energy and provides information on the intensity of velocity fluctuations associated with resolved-scale flow structures. High values of k_t are typically induced by flow separation, recirculation, and geometric irregularities. By aggregating these quantities over anatomically defined regions and transversal sections, the resulting feature vector $\mathbf{P}_{i,j}$ combines geometric sensitivity with physically meaningful flow descriptors. This design yields a compact yet expressive representation of the airflow that is robust to mesh resolution and numerical noise, while remaining interpretable from both a fluid-dynamics and a clinical perspective.

4.5.3 Evaluation Protocol and Experimental Objectives

The effectiveness of the proposed data augmentation strategy is evaluated through two complementary experimental settings, designed to assess both the generalisation on synthetic data and the generalisation to real patient-specific anatomies.

1. **Synthetic validation.** In the first setting, classifiers are trained and tested exclusively on synthetic samples generated through the proposed augmentation pipeline. A LOPO-CV protocol is adopted, in which, at each fold, all augmented samples derived from one of the healthy subjects are excluded from the training set and used for testing. This evaluation enforces subject-level separation between training and testing data, as all synthetic samples generated from a given healthy anatomy are never observed during training when that anatomy is used for testing. Consequently, the classifier is required to generalise across different underlying anatomical shapes and can only rely on flow alterations induced by the imposed pathological deformations. This setting, therefore, verifies that the

model does not exploit subject-specific anatomical cues implicitly shared among synthetic samples originating from the same healthy geometry.

2. **Generalisation to real patients.** In the second setting, classifiers trained solely on synthetic data are evaluated on an independent set $\mathcal{D}_{\text{real}}$ composed of 10 real pathological patients, never seen during training. These subjects present clinically diagnosed nasal pathologies and are extracted from CT scans following the same preprocessing pipeline adopted for healthy subjects. No synthetic deformation or augmentation is applied to these geometries. This evaluation directly assesses the ability of the proposed augmentation strategy to bridge the gap between synthetic CFD-derived data and real patient-specific anatomies, which represent the most challenging and clinically relevant scenario.

In particular, the experiments are designed to answer the following key questions:

(i) Can a classifier trained exclusively on augmented CFD data correctly generalise to synthetic samples derived from anatomical geometries not observed during training?

(ii) Can a classifier trained exclusively on augmented CFD data generalise to previously unseen *real* pathological geometries?

(iii) Does increasing deformation diversity and severity improve generalisation performance?

(iv) Are the learned decision rules consistent with clinical intuition and known flow-physics principles?

(v) Which CFD-derived features and anatomical regions are most influential in the classification process?

Quantitative results and qualitative analyses addressing these questions are presented and discussed in the following section.

4.5.4 ML Model Training

Supervised classification models are trained on feature vectors obtained from regional averages of CFD-derived quantities, as described in Section 4.5.2. Five different classifiers are considered in order to assess the contribution of individual flow quantities and their combination for pathology discrimination.

Four classifiers take as input feature vectors extracted from a single flow quantity. Specifically, each of these models processes a vector of 12 features corresponding to the regional averages computed on 6 transversal sections, each

Model	Input features	Architecture
$K_{\lvert \boldsymbol{U} \rvert}$, $K_{\lvert \nabla p \rvert}$, K_E, K_{k_t}	12 features (6 sections × left/right regions)	MLP with four hidden layers: 60, 40, 20, 10 neurons ReLU activations 4261 trainable parameters
K_{Full}	48 features (4 quantities × 6 sections × left/right regions)	MLP with four hidden layers: 120, 60, 30, 10 neurons ReLU activations 15291 trainable parameters

Table 4.2: Summary of the classification models considered in this work. Four single-quantity classifiers operate on regional averages of individual CFD-derived quantities, while the full model exploits the concatenated feature representation. All architectures are implemented as multilayer perceptrons with ReLU activations.

divided into left and right regions. These classifiers are denoted as $K_{\lvert \boldsymbol{U} \rvert}$, $K_{\lvert \nabla p \rvert}$, K_E, and K_{k_t}, according to the CFD quantity used for feature extraction. All four models are implemented as multilayer perceptrons (MLPs). After preliminary hyperparameter tuning, a common architecture is adopted, consisting of four hidden layers with 60, 40, 20, and 10 neurons, respectively, for a total of 4261 trainable parameters. This choice reflects a trade-off between model expressiveness and robustness in the small-data regime considered in this work.

In addition to single-quantity classifiers, a fifth model, denoted as K_{Full}, is trained using the complete feature vector obtained by concatenating all regional averages from the four CFD quantities. This model takes as input 48 features and is implemented as an MLP with four hidden layers containing 120, 60, 30, and 10 neurons, respectively, for a total of 15291 trainable parameters.

For all architectures, the output layer consists of a single neuron producing a scalar value in the interval [0,1], corresponding to the predicted class label, with 0 indicating turbinate hypertrophy and 1 indicating septal deviation. All hidden layers use the ReLU activation function.

Prior to training, feature vectors are standardised to have zero mean and unit variance, with statistics computed on the training set only. Models are trained by minimising the binary cross-entropy loss on the augmented training set using the Adam optimiser with a learning rate of 10^{-3}, for a maximum of 1000 epochs. An early stopping criterion based on the validation loss is adopted to prevent overfitting. At each training epoch, 85% of the available samples are used for training and the remaining 15% for validation. The same training protocol is applied consistently across all experimental settings.

	Training set	LOPO-CV	S	
	Test set	LOPO-CV	$\mathcal{D}_{real}$	
Classifier	$K_{\|\boldsymbol{U}\|}$	Acc: 89.48%, AUROC: 0.923	7/10	Results
	$K_{\|\nabla p\|}$	Acc: 84.23%, AUROC: 0.911	8/10	
	K_E	Acc: 84.55%, AUROC: 0.851	8/10	
	K_{k_t}	Acc: 66.26%, AUROC: 0.672	6/10	
	K_{Full}	Acc: 86.35%, AUROC: 0.917	8/10	

Table 4.3: Classification results obtained during Leave-One-Patient-Out Cross-Validation on the synthetic dataset, reported in terms of accuracy and AUROC on the LOPO-CV test folds, together with test performance on the real patient set $\mathcal{D}_{\text{real}}$.

4.6 Results

This section reports the experimental results obtained using the proposed geometry-based data augmentation framework. The goal of the analysis is to evaluate whether classifiers trained exclusively on synthetic CFD-derived data are able to generalise across unseen anatomies despite the severe data scarcity characterising this problem. Results are presented following the evaluation protocol introduced in Section 4.5.3, and are organised to progressively assess generalisation, robustness, and consistency of the learned models.

4.6.1 Synthetic Validation via Leave-One-Patient-Out Cross-Validation

In this experiment, at each fold, the classifier is trained on all synthetic samples derived from 6 healthy anatomies and tested on the synthetic samples generated from the held-out one. After iterating over all healthy subjects, performance metrics are averaged across folds. This evaluation assesses the ability of the classifiers to generalise across different anatomical geometries while being exposed to the same set of synthetic pathological deformations.

Classification results obtained during LOPO-CV are reported in Table 4.3. In addition to accuracy, we also report the Area Under the Receiver Operating Characteristic curve (AUROC), which provides a threshold-independent assessment of the separability between the two pathology classes. The corresponding ROC curves are shown in Figure 4.7, while the confusion matrices are reported in Figure 4.8. Overall, all classifiers achieve good performance on

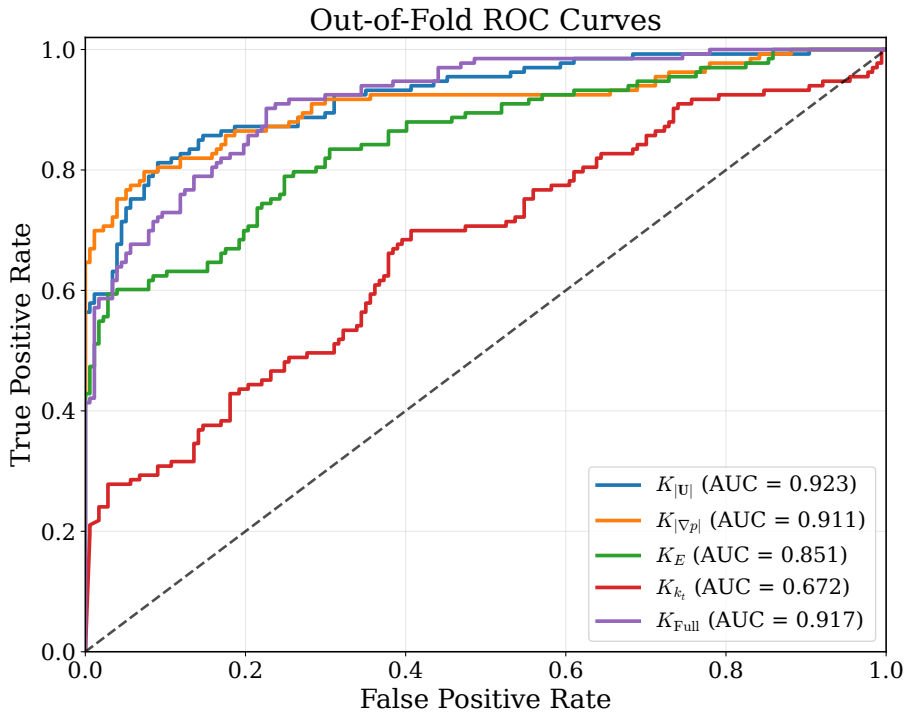


Figure 4.7: Receiver Operating Characteristic (ROC) curves obtained from the out-of-fold predictions of the LOPO-CV procedure for all classifiers. The figure highlights the different separability achieved by the considered CFD-derived feature sets.

never-seen-before synthetic anatomies, confirming that the proposed geometry-based augmentation strategy produces CFD patterns that are sufficiently consistent across subjects to enable cross-patient generalisation. The best performance is obtained by $K_{|\boldsymbol{U}|}$, the classifier trained on the velocity magnitude, which achieves the highest LOPO-CV accuracy (89.48%) and a high AUROC (0.923). Comparable results are obtained by $K_{|\nabla p|}$ and K_{Full}, with AUROC values above 0.91, indicating excellent discriminative ability even when classification thresholds are varied. The classifier based on enstrophy, K_E, remains reasonably effective (AUROC = 0.851), whereas K_{k_t} exhibits a markedly lower performance both in terms of accuracy and AUROC (Acc: 66.26%, AUROC = 0.672), suggesting a weaker separation between the two pathology classes.

The ROC curves in Figure 4.7 further support these findings. In particular, the curves associated with $K_{|\boldsymbol{U}|}$, $K_{|\nabla p|}$, and K_{Full} remain consistently close to the top-left corner of the ROC plane, confirming that these representations provide a strong separation between hypertrophy and septal deviation across a broad range of classification thresholds. This is especially relevant in a clinical setting, where the optimal operating point may vary depending on whether

sensitivity or specificity is prioritised. By contrast, the flatter ROC curve of K_{k_t} indicates substantially weaker discriminative power, consistently with its lower out-of-fold AUROC. Overall, the AUROC analysis confirms that velocity- and pressure-related quantities are the most informative CFD-derived markers for pathology classification in the proposed framework.

This behaviour reflects the different sensitivity of the considered flow quantities to geometry-induced variations. Velocity magnitude and pressure gradient are directly influenced by local changes in cross-sectional area and flow acceleration, which are strongly affected by the geometric deformations associated with nasal pathologies. Enstrophy captures resolved velocity gradients and rotational structures, which are also impacted by geometric irregularities. Conversely, the resolved turbulent kinetic energy k_t encodes fluctuations at the resolved scales of the simulation and appears to be less directly correlated with the specific geometric modifications induced by the considered pathologies.

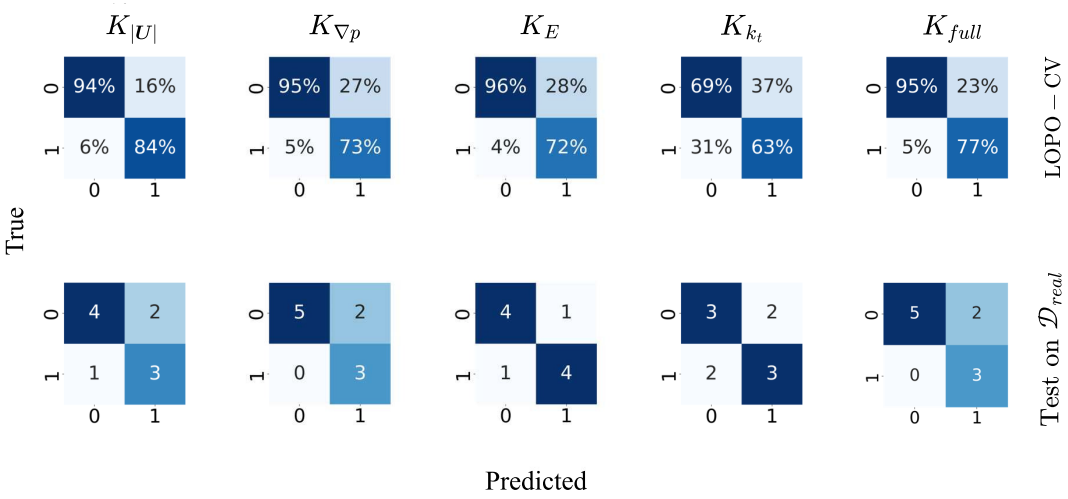


Figure 4.8: Confusion matrices illustrating classifier performance. Top row: LOPO-CV scores on synthetic data. Bottom row: test performance on real patient data. Labels 0 and 1 correspond to hypertrophy and septal deviation, respectively.

Beyond overall accuracy, the confusion matrices reported in Figure 4.8 show that turbinate hypertrophies are generally classified more accurately than septal deviations. We associate this behaviour with the different ways in which the two pathologies perturb the airway geometry and, consequently, the resulting flow field. Hypertrophies typically induce pronounced and localised geometric constrictions, leading to strong alterations in velocity and pressure distributions that are readily captured by the proposed CFD-derived features. Septal deviations, instead, may in some clinical cases correspond to mild or smoothly bent septal geometries that do not necessarily generate distinct flow signatures.

As a result, the associated fluid dynamic features can partially overlap with those of healthy anatomies, increasing ambiguity and leading to a higher rate of misclassification.

To further investigate the impact of pathology severity on classification performance, we analyse LOPO-CV results by restricting the test samples to synthetic cases exhibiting a single pathology at different severity levels. Figure 4.9 reports the confusion matrices obtained when testing on mild ($S1$) and severe ($S2$) pathologies. Across all models, classification performance improves when testing on more severe cases, indicating that stronger geometric deformations induce more pronounced and easily identifiable flow patterns. This trend confirms the internal consistency of the synthetic dataset generated by the proposed augmentation method and highlights the role of deformation severity in shaping discriminative CFD features.

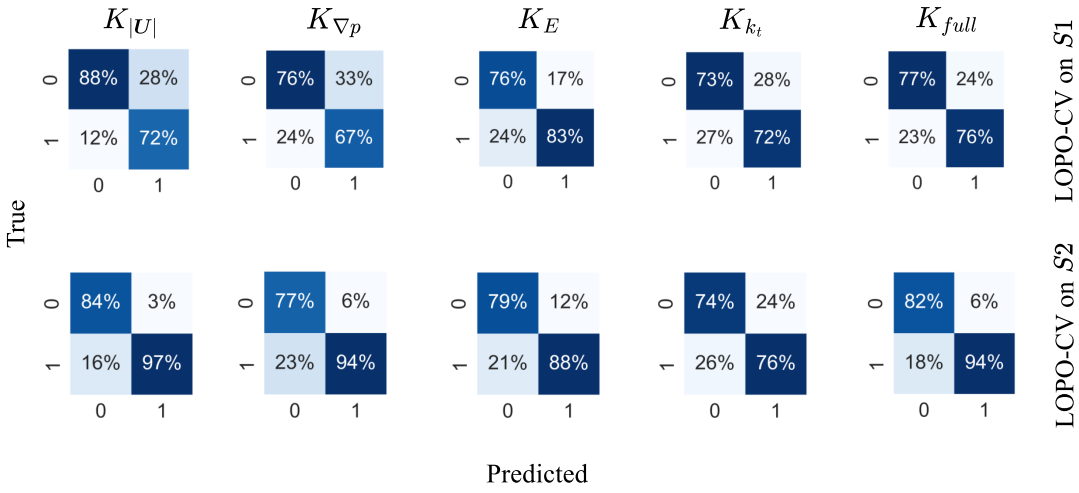


Figure 4.9: LOPO-CV performance on test patients with a single pathology. Top row: accuracy on patients in $S1$. Bottom row: accuracy on patients in $S2$.

4.6.2 Generalisation to Real Patients

In this section, we discuss the ability of classifiers trained exclusively on synthetic data to generalise to real patient-specific anatomies. All models are trained on the complete synthetic dataset S and tested on an independent set $\mathcal{D}_{\text{real}}$ consisting of 10 real pathological patients, never observed during training. These subjects present clinically diagnosed nasal pathologies and are evenly distributed between turbinate hypertrophy and septal deviation. The limited size of $\mathcal{D}_{\text{real}}$ reflects the substantial cost and complexity associated with acquiring new patient data, including CT acquisition, expert-driven geometry extraction

and cleaning, and the execution of high-fidelity CFD simulations. All real geometries are processed using the same surface extraction, cleaning, and CFD simulation pipeline adopted for healthy subjects, as described in Section 4.3.

Classification accuracy on real patients is reported in Table 4.3, while the corresponding confusion matrices are shown in Figure 4.8. The best-performing classifiers correctly identify 8 out of 10 pathological cases. Considering the significant inter-patient anatomical variability and the fact that the augmentation strategy is applied to only 7 healthy anatomies, these results highlight the effectiveness of the proposed approach in transferring knowledge learned from synthetic data to real, previously unseen patients. This capability is particularly relevant in clinical settings, where ENT doctors are required to diagnose new patients without access to prior patient-specific information.

The variability in classification accuracy across different classifiers confirms that the discriminative power of the models strongly depends on the physical meaning of the adopted features. The high performance of $K_{|\boldsymbol{U}|}$ is expected, as the velocity magnitude is directly linked to geometric properties of the airway, in particular to variations in the cross-sectional area affected by the considered deformations, as discussed in Section 4.5.2. In contrast, $|\nabla p|$, E, and k_t are purely fluid-dynamic quantities, whose ability to discriminate pathologies relies on how strongly geometric alterations are reflected in the resulting flow field. Within this group, the satisfactory performance of K_E and $K_{|\nabla p|}$ indicates that the considered pathologies induce sufficiently distinct flow signatures, whereas the lower accuracy of k_t suggests a weaker and less consistent coupling with the underlying geometric deformations.

Overall, these results show that, through the proposed augmentation strategy, ML models trained exclusively on synthetic CFD data can generalise effectively to real patient-specific anatomies, enabling the identification of clinically relevant nasal pathologies despite substantial inter-patient variability.

4.6.3 Effect of Pathology Severity and Consistency under Deformation Strengthening

In addition to the primary evaluation on synthetic and real data, we extrapolate further results to assess the robustness and internal consistency of the learned classifiers. These analyses are particularly important given the limited size of the real patient test set and aim to verify whether model predictions are stable with respect to controlled variations in pathology severity.

Impact of pathology severity and deformation diversity. We first investigate how the composition of the synthetic training set affects generalisation to real patient-specific anatomies. To this end, classifiers are trained using progressively richer subsets of the augmented dataset. Specifically, we consider

	Training set	$S1$	$S12$	S	S	
	Test set	$\mathcal{D}_{\text{real}}$	$\mathcal{D}_{\text{real}}$	$\mathcal{D}_{\text{real}}$	$\mathcal{D}_{\text{boost}}$	
Classifier	$K_{\lvert \boldsymbol{U} \rvert}$	4/10	5/10	7/10	8/10	Accuracy
	$K_{\lvert \nabla p \rvert}$	4/10	7/10	8/10	9/10	
	K_E	4/10	5/10	8/10	8/10	
	K_{k_t}	5/10	5/10	6/10	6/10	
	K_{Full}	5/10	7/10	8/10	9/10	

Table 4.4: Classification accuracy for classifiers trained on progressively enriched synthetic datasets ($S1$, $S12$, and S) and tested on the real patient set $\mathcal{D}_{\text{real}}$. Results for classifiers trained on the complete synthetic dataset S and tested on boosted patient geometries $\mathcal{D}_{\text{boost}}$ are also reported.

three synthetic training sets: $S1$, containing only samples exhibiting a single pathology with low severity; $S12$, extending $S1$ with higher severity levels; and S, the complete synthetic dataset including multiple severity levels and combinations of pathological deformations.

All classifiers are evaluated on the same real patient set $\mathcal{D}_{\text{real}}$. Classification scores are reported in Table 4.4. Results show a clear and consistent improvement in performance as the training set is enriched with an additional severity level and deformation combinations. This trend indicates that increasing anatomical and pathological diversity in the synthetic data is a key factor in improving generalisation to real patients, and proves that training a classifier using more and more data generated from our augmentation method can enhance the generalisation of our classifiers to unseen patients.

Consistency under pathology strengthening. We further assess the stability and consistency of classifier predictions by evaluating models on real patient geometries with artificially strengthened pathological features. For each patient in $\mathcal{D}_{\text{real}}$, the deformation primitive δ_j corresponding to the expert-provided clinical diagnosis Y_j of that patient is transferred from the reference surface S^{ref} to the patient-specific geometry using the correspondence map described in Section 4.4.2. As a result, we obtain a geometrically consistent variant of each real patient in which the clinically observed pathology is intentionally amplified, yielding the boosted dataset $\mathcal{D}_{\text{boost}}$. CFD simulations and feature extraction are performed on these boosted geometries using the same pipeline adopted for both synthetic and real samples. Classifiers trained on the complete synthetic dataset S are then evaluated on $\mathcal{D}_{\text{boost}}$.

As reported in Table 4.4, classification performance on boosted patients either improves or remains unchanged across all models. This behaviour shows

that classifier predictions are consistent under controlled increased pathology severity and supports the conclusion that the learned decision rules are grounded in meaningful flow-related patterns rather than in spurious correlations.

Overall, the combined analysis of pathology severity and deformation strengthening shows that the proposed augmentation framework enables the training of classifiers that are both robust to variations in pathological expression and stable under controlled geometric perturbations. These results further substantiate the ability of the proposed approach to support pathology identification in real and previously unseen anatomies.

4.6.4 Explainability

The complexity and high dimensionality of CFD-derived data make direct interpretation of flow fields challenging, particularly in the presence of nonlinear interactions between geometry and fluid dynamics. While ML models can effectively capture such interactions, their use in a clinical context requires a clear understanding of the mechanisms driving their predictions. An explainability analysis based on SHAP (SHapley Additive exPlanations) [64] is therefore performed with the aim of identifying which CFD-derived features and anatomical regions contribute most significantly to pathology classification.

SHAP analysis provides a principled framework to explain the output of machine learning models by quantifying the contribution of each input feature to the model prediction through Shapley values. Shapley values originate from cooperative game theory [77], where a set of players collaborates to achieve a common outcome and the total payoff is distributed among them according to their individual contributions. In the present context, the players correspond to CFD-derived features, while the payoff is given by the output of K.

Formally, The Shapley value ϕ_i associated with the i-th feature measures its average marginal contribution to the prediction, computed by averaging over all possible subsets of features, and is defined as follows:

$$\phi_i = \sum_{Z \subseteq N_f \setminus \{i\}} \frac{|Z|!\,(|N_f| - |Z| - 1)!}{|N_f|!} \Big(K(Z \cup \{i\}) - K(Z) \Big), \tag{4.10}$$

where N_f denotes the set of all input features provided to the classifier, and Z is any subset of N_f that does not contain feature i. Moreover, $K(Z)$ indicates the model output when the input is restricted to the features in Z, while $K(Z \cup \{i\})$ denotes the output obtained by including also feature i. The notation $|\cdot|$ denotes set cardinality; hence, $|Z|!$ is the factorial of the number of elements in Z. Intuitively, ϕ_i quantifies the gain (or loss) in the model output induced by adding feature i to a subset Z of the remaining features, and then averaging this marginal contribution over all possible subsets (equivalently, over all feature

orderings through the Shapley weighting)[1].

A key property of Shapley values is additivity, which allows each prediction to be decomposed as the sum of a baseline prediction (the average model output over the dataset) and the individual feature contributions. This property makes SHAP particularly suitable for interpreting complex models trained on structured feature representations.

We apply SHAP analysis to the classifier K_{Full}, which takes as input the complete set of CFD-derived features. As the task is binary classification, Shapley values are expected to lie in a bounded interval, with positive values biasing predictions toward septal deviation and negative values biasing predictions toward turbinate hypertrophy. In light of the performance observed in the LOPO-CV and real patient experiments (Sections 4.6.1 and 4.6.2), we expect features associated with more informative CFD quantities to exhibit larger absolute Shapley values.

Figure 4.10 reports the Shapley values of the most influential features, ranked according to the mean absolute value of their Shapley contributions across all LOPO-CV test samples. Feature importance is therefore quantified as the mean absolute Shapley value associated with each feature, providing a global measure of its influence on the classification outcome. To do that, for each iteration of the LOPO-CV, Shapley values are computed on the test samples associated with the excluded patient. The results are subsequently aggregated over all LOPO-CV iterations, yielding a single Shapley value distribution per feature that summarises its behaviour across the entire synthetic test set. Each point in Figure 4.10 corresponds to the Shapley value of a single feature for one test sample, while the colour scale encodes the feature magnitude, ranging from low values (blue) to high values (red). Positive Shapley values bias the classifier prediction toward septal deviation (label 1), whereas negative values bias the prediction toward turbinate hypertrophy (label 0).

We observe that the results of the SHAP analysis are fully consistent with the classification performance reported in Figure 4.8. In particular, the most influential features according to their Shapley values, namely the regional averages of velocity magnitude and pressure gradient ($|\boldsymbol{U}|$ and $|\nabla p|$), correspond to the CFD quantities that achieved the highest classification accuracy when considered individually in Section 4.6.1.

A closer inspection reveals that features associated with the left and right regions of the same transversal section often exhibit an antisymmetric behavior. For instance, the regional averages $|\boldsymbol{U}|_{1,\text{L}}$ and $|\boldsymbol{U}|_{1,\text{R}}$, computed on the left and right portions of Section 1, show Shapley value distributions with opposite signs for comparable feature magnitudes. Specifically, low values of $|\boldsymbol{U}|_{1,\text{R}}$ tend to be

[1]In practice, "excluding" a feature i corresponds to replacing its value with a baseline value in the model input vector, typically the mean value computed on the training distribution.

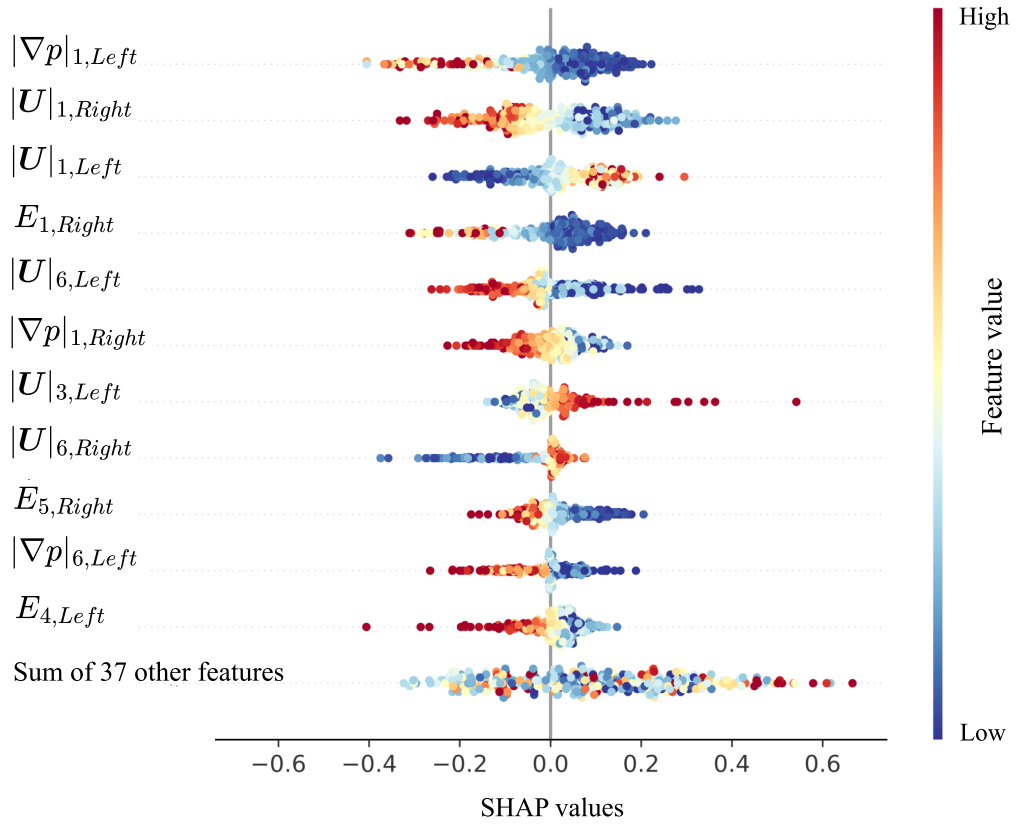


Figure 4.10: Shapley values of the top-ranked CFD-derived features for the classifier K_{Full}. Each row corresponds to a feature associated with a specific section and left/right region. Each point represents the Shapley value computed for one synthetic test sample during the LOPO-CV procedure. Positive values bias the prediction toward septal deviation, while negative values bias it toward turbinate hypertrophy. Colors indicate feature magnitude, with red denoting high values and blue low values.

associated with positive Shapley values, whereas low values of $|\boldsymbol{U}|_{1,\text{L}}$ correspond to negative Shapley values. As a consequence, when the left and right regional features assume similar values, their Shapley contributions tend to compensate each other, resulting in a limited net influence on the final prediction. This behavior is consistent with healthy or quasi-symmetric anatomical configurations, in which balanced airflow between the left and right cavities induces feature contributions that naturally offset one another. At the same time, the fact that Shapley value distributions are not perfectly symmetric with respect to

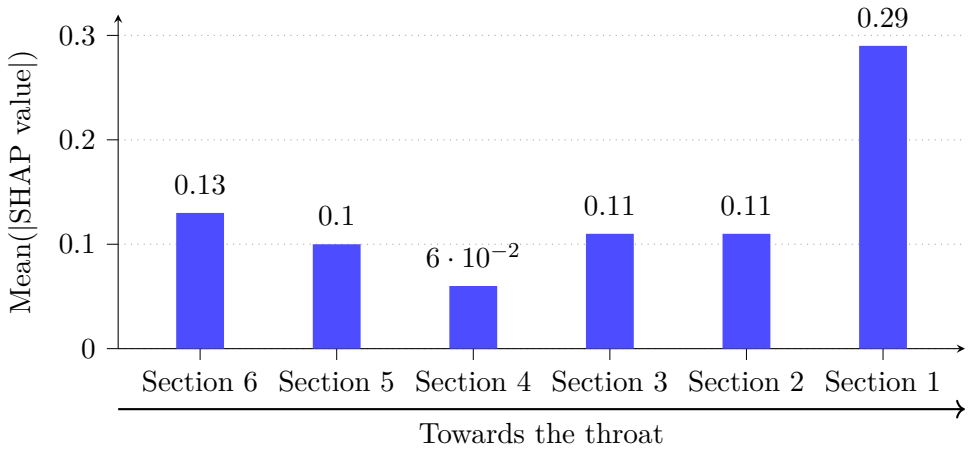


Figure 4.11: Mean absolute Shapley values aggregated at the level of transversal sections. For each section, Shapley values of all associated features are aggregated and averaged across LOPO-CV test samples, providing a measure of the relative importance of each section in the classification process.

zero indicates that even approximately symmetric flow patterns may still carry diagnostic information. This observation is in line with the presence of pathological conditions, such as certain forms of turbinate hypertrophy, that can alter the flow while preserving a largely symmetric left–right distribution.

Similar considerations apply to the pressure-gradient features $|\nabla p|_{1,\mathrm{L}}$ and $|\nabla p|_{1,\mathrm{R}}$. In this case, however, the dependence between feature values and Shapley contributions exhibits the same qualitative trend on both sides: high values of $|\nabla p|$ are associated with negative Shapley values, whereas lower values bias the prediction toward septal deviation. This indicates that, unlike velocity magnitude, pressure-gradient features convey diagnostic information primarily through their absolute intensity rather than through left–right asymmetry.

Influence of the Transversal Sections on the Predictions In addition to identifying the most influential CFD-derived features, SHAP analysis is employed to investigate the contribution of each of the six transversal sections introduced in Section 4.5.2 to the final classification outcome. From the SHAP results obtained for the classifier K_{Full} and reported in Figure 4.10, it can be observed that the majority of the most influential features is associated with either Section 1 or Section 6.

To further substantiate this observation, we exploit the additivity property of Shapley values to quantify the contribution of each section to the model predictions. Specifically, for each iteration of the LOPO-CV and for each test sample, Shapley values corresponding to the features extracted from the same transversal section are summed, yielding a section-wise Shapley contribution.

The absolute values of these aggregated section-wise contributions are then averaged across all test samples, resulting in a mean Shapley value that quantifies the overall influence of each section on the classification task.

The resulting section-level analysis is reported in Figure 4.11, which shows the mean absolute Shapley values for each of the six transversal sections. The results confirm the trend already suggested by Figure 4.10, indicating that Sections 1 and 6 provide the largest contribution to the classifier decisions. While the prominence of Section 6 is expected, as this region is most directly affected by the considered pathologies, the strong influence of Section 1 is less intuitive. We attribute this behaviour to the fact that Section 1, located at the end of the olfactory region (Figure 4.6), captures flow patterns that reflect the cumulative effects of upstream geometric and aerodynamic alterations along the nasal cavity. This interpretation is further supported by the ordering of sections in Figure 4.11, where, excluding Section 6, the remaining sections exhibit a progressively decreasing influence as they are positioned farther upstream. Despite these differences, it is important to note that all six sections contribute non-negligibly to the predictions, indicating that the impact of nasal pathologies on airflow is distributed across the entire computational domain rather than being localised to a single region.

Overall, the SHAP-based analysis confirms that the classifiers rely on complex yet physically meaningful patterns embedded in the CFD-derived features. It provides a spatially resolved interpretation of the learned decision rules and is fully consistent with the performance trends discussed in Section 4.6.1, further supporting the validity of the proposed data augmentation strategy in this clinical setting.

4.7 Conclusions

Previous work by Schillaci et al. [97] demonstrated that neural networks can successfully classify CFD flow fields in controlled and simplified scenarios. In that setting, the high dimensionality of CFD data and the substantial computational cost associated with flow simulations imposed severe constraints on the complexity of the considered geometries, which were necessarily parametrised and idealised. As a consequence, although the study highlighted the potential of data-driven approaches for flow-based classification, its applicability to realistic and patient-specific anatomical configurations remained limited.

Here, we extend that paradigm by addressing the problem of pathology classification in anatomically realistic and non-parametric upper airway geometries. The central contribution of this chapter is the introduction of a geometry-based data augmentation framework that enables the generation of a large, labeled set

of CFD simulations starting from a small number of CT scans of healthy individuals. By defining deformation primitives on a reference anatomy and transferring them to subject-specific geometries through intrinsic correspondence, the proposed approach allows the systematic synthesis of pathological configurations while preserving anatomical plausibility and semantic consistency. Importantly, this strategy significantly reduces the need for expert supervision, as pathological labels are automatically inherited from the applied deformation functions.

The experimental results presented in this chapter show that classifiers trained exclusively on synthetic and augmented CFD data are able to generalise beyond the training distribution. In particular, high classification accuracy is achieved on synthetic test samples generated from unseen healthy anatomies, indicating that the models do not simply memorise subject-specific features. More importantly, the same classifiers demonstrate robust performance when evaluated on real pathological patients that were never observed during training. This result is especially relevant in a clinical context, as it shows that learning from synthetically generated CFD data can be sufficient to identify pathological flow patterns in real patient-specific anatomies.

The robustness of the proposed framework is further supported by additional analyses investigating the effect of training set composition and pathological severity. Classification performance consistently improves as the diversity and severity of synthetic deformations included in the training set increase, mirroring well-known trends observed when learning from real clinical data. Moreover, classifier predictions remain stable when pathological conditions are synthetically strengthened on test patients, suggesting that the learned decision rules are not sensitive to small perturbations but instead capture coherent and physically meaningful flow alterations associated with the considered pathologies.

Complementing the quantitative performance evaluation, the SHAP-based explainability analysis provides further insight into the behaviour of the trained models. The analysis confirms that the most influential features correspond to CFD quantities that are known to be strongly affected by geometric modifications, such as velocity magnitude and pressure gradients. Furthermore, the spatial distribution of influential features across transversal sections highlights that the classifiers exploit both local and cumulative flow effects, rather than relying on isolated regions. These observations support the conclusion that the models base their predictions on physically interpretable patterns embedded in the flow field, rather than on spurious correlations.

Scalability of the Dataset A notable strength of the proposed data augmentation framework is its inherent scalability. Once a reference surface and a set of deformation primitives are defined, synthetic pathological samples can be generated in a largely automated manner. Increasing the size of the training

dataset primarily requires access to additional CT scans of healthy individuals, while the definition of new pathologies only demands limited expert intervention on the reference anatomy. From a methodological standpoint, no fundamental limitation prevents the extension of the dataset to a substantially larger number of samples.

At the same time, the scalability of the framework is practically constrained by the computational cost of CFD simulations. High-fidelity simulations remain the most computationally demanding component of the pipeline and represent the primary bottleneck in dataset generation. Nevertheless, this limitation is external to the augmentation methodology itself and reflects the intrinsic cost of simulating complex physiological flows rather than a conceptual shortcoming of the proposed approach.

Dataset Representativeness The relatively small number of real patient scans naturally raises questions regarding dataset representativeness. In the considered application, however, this limitation is difficult to avoid, as nasal pathologies exhibit large geometric variability and reliable clinical labels require expert assessment. Despite these constraints, the experimental results suggest that the functional impact of a given pathology on the airflow is sufficiently consistent across patients to be captured by the augmented dataset.

The ability of classifiers trained on synthetic data to generalize to real, never-seen patients indicates that the proposed augmentation strategy produces CFD samples that are representative of clinically relevant flow alterations. This observation supports the underlying assumption that, although anatomical shapes may differ substantially, pathologies belonging to the same class induce characteristic flow patterns that can be learned from a limited but carefully constructed dataset.

Automatic Segmentation and Noise Reduction An additional practical aspect of the proposed framework concerns surface extraction and preprocessing from CT data. In practice, the impact of this automated preparation step is substantial. Preparing the test set composed of 10 real pathological patients required a significantly larger amount of manual effort compared to the generation of the synthetic dataset, which consists of 308 pathological geometries obtained by augmenting only 7 healthy subjects. This comparison highlights how the proposed augmentation framework shifts the most demanding expert-driven operations to a limited number of baseline cases, while enabling the efficient creation of large labeled datasets.

At the same time, noise and segmentation inaccuracies in CT-derived geometries remain a relevant source of uncertainty. In the current study, segmentation errors are corrected manually by experts, which is feasible due to the limited number of healthy scans involved. While this issue does not critically affect the

validity of the present results, it highlights an important practical consideration when applying the framework to larger datasets.

Class Imbalance and Rare Pathologies Although the dataset considered in this Chapter is intentionally balanced, class imbalance represents a common challenge in clinical applications, particularly for rare pathologies. The proposed augmentation framework inherently mitigates this issue by enabling the controlled generation of synthetic samples for under-represented classes. This capability is especially valuable in medical settings, where collecting sufficient data for uncommon conditions is often impractical. While class imbalance does not pose a limitation in the present study, it represents an important aspect of the problem that is naturally addressed by the augmentation strategy introduced in this work.

To conclude, the results presented in this chapter show that geometry-based data augmentation provides an effective strategy to mitigate data scarcity in learning-based inference from CFD simulations. By leveraging anatomically realistic deformations and physically consistent flow simulations, the proposed framework enables the construction of labelled datasets starting from a limited number of reference geometries, while reducing the dependence on extensive expert intervention. The focus now shifts from the problem of limited sample availability (*Small-n*) to the complementary and equally critical challenge posed by the extreme dimensionality of CFD data (*Large-p*). Accordingly, Chapter 5 investigates how compact, informative, and physically meaningful feature representations can be extracted from high-dimensional flow fields to support efficient and robust learning.

Chapter 5

Feature Extraction from CFD Data via Clustering and Morphing

5.1 Motivation and Overview

In Chapter 4, we addressed the problem of data scarcity in inference from CFD simulations. In this chapter, we turn to the complementary and equally critical challenge: the high dimensionality of CFD data (Large-p), which persists even when the number of available simulations is sufficiently large. As detailed in Section 1.3, each flow field $\mathbf{F}_i$ typically consists of millions of spatial degrees of freedom and multiple scalar and vector quantities per cell, resulting in data representations that are extremely high-dimensional and strongly correlated. Directly using raw CFD fields as input to learning models is therefore computationally infeasible and conceptually ill-posed, especially when the objective is to infer quantities that are not explicitly encoded in the flow variables themselves.

In this context, we recall that the classical way to feed CFD data into an ML model relies on handcrafted features designed by domain experts. The dimensionality of CFD flow fields is reduced by manually defining regions of interest within the computational domain and extracting selected physical quantities as regional averages, based on prior knowledge. As already introduced in Chapter 4, a representative example of this paradigm is the framework proposed by Schillaci et al. [97], where flow features are extracted from *a priori* defined locations to enable inference tasks in both aerodynamic and biomedical settings, including airfoil characterisation and respiratory pathology classification in simplified nasal geometries. Within the same paradigm, and consistently with the setting described in Chapter 4, this chapter adopts as baseline the extension of this approach to real anatomies of the human upper airways reconstructed from

CT scans and simulated via CFD. In this case, features are extracted from regions associated with fixed anatomical sections defined by ENT experts, which must be manually specified and adapted on a patient-specific basis to account for anatomical variability.

Handcrafted feature extraction strategies of this kind present important strengths. They explicitly leverage expert knowledge, either medical or fluid-dynamical, to define interpretable regions tied to anatomical landmarks or aerodynamic structures, and they provide descriptors that are directly accessible to domain experts. For this reason, handcrafted features remain a valid strategy. However, their main limitation lies in scalability: because regions must be redefined and adapted on a case-by-case basis, particularly in datasets characterised by strong geometric variability, the process becomes time-intensive, costly, and prone to inconsistencies across samples.

To overcome these limitations, in this Chapter, we introduce two alternative strategies for defining flow regions while eliminating the need for repeated manual intervention while preserving physical consistency. The first is a *Clustering-based* method, in which regions are automatically identified directly from the flow field. The second is a *Morphing-based* method, where regions are defined once by an expert on a reference geometry and then consistently transferred across simulations by mapping each flow field onto the reference.

In the *Clustering-based* approach, the CFD domain is segmented into regions by clustering the terms of the governing equations. Cells characterised by a similar local balance of physical contributions, such as advection, diffusion, and pressure gradient, are grouped together, leading to regions that reflect underlying flow phenomena rather than predefined geometric partitions. This strategy removes the need for *a priori* region specification, enabling an adaptive, simulation-specific partitioning driven by the underlying flow physics. At the same time, it introduces new challenges: since clustering is performed independently for each simulation, the resulting regions may differ across samples, and areas capturing similar physical phenomena may not correspond spatially. As a consequence, the extracted features form unordered and potentially inconsistent sets. To address this issue, two complementary solutions are considered: either explicit correspondences between clusters across samples are established, enabling the use of standard ML models, or permutation-invariant models are adopted to directly process unordered sets of features. Despite their effectiveness from a data-driven perspective, features extracted from automatically identified regions lack explicit anatomical or geometric grounding, which hinders their interpretability in application domains, such as clinical settings, where understanding of the decision process is required.

The *Morphing-based* approach is motivated precisely by the need to preserve interpretability while achieving scalability. Inspired by [16], this method ensures spatial alignment across simulations by mapping each flow field onto

a unique reference geometry. Once a reference configuration is selected, each CFD simulation is morphed onto it by fitting a smooth deformation model based on Radial Basis Functions (RBFs) [9]. This procedure aligns mesh topologies and flow fields across samples, allowing features to be extracted from regions defined only once on the reference geometry and consistently reused across the dataset. By embedding expert knowledge into the reference and propagating it automatically, this strategy eliminates repetitive manual operations and enables scalable learning in the presence of geometric variability, surface defects, or anatomical differences.

The two proposed methods thus address complementary requirements: adaptability, achieved through clustering-based region identification that conforms to the specific flow field, and transferability, achieved through morphing-based alignment that enforces consistency across geometries. We evaluate these feature extraction strategies across the scenarios described in Chapter 3, which increase in complexity and variability. The first set of experiments focuses on two large datasets of two-dimensional CFD simulations around NACA airfoils, including a publicly available dataset [96] and an extended version generated to include controlled geometric defects (see Section 6.1). In this data-rich setting, we address two regression tasks: *i*) prediction of the NACA four-digit airfoil code, and *ii*) estimation of geometric defects such as bumps, cavities, and cut trailing edges. The second experimental scenario and task coincides with that introduced in Chapter 4, namely the classification of nasal pathologies in the human upper airways. Together with the aerospace case, this scenario allows the proposed methodology to be evaluated under markedly different data regimes and geometric complexities.

The main contributions of this chapter can be summarised as follows:

- We address a fundamental limitation in applying ML to CFD data, namely the lack of standardised and scalable feature extraction procedures that remain valid across simulations with heterogeneous geometries.
- We introduce two complementary strategies for feature extraction from CFD data: a *Clustering-based* method that adaptively identifies flow-consistent regions, and a *Morphing-based* method that transfers expert-defined regions across heterogeneous geometries.
- We validate the proposed methods on real-world CFD applications of increasing complexity and demonstrate improved scalability and performance compared to established handcrafted approaches [97].

The chapter is organised as follows. Section 5.2 reviews the relevant background and related work on dimensionality reduction and feature extraction from CFD data, with particular emphasis on clustering-based representations

and geometry-alignment techniques. Section 5.3 introduces the problem formulation and notation. Section 5.4 presents the proposed methodologies for region definition and feature extraction, covering both the clustering-based approach (Section 5.4.1) and the morphing-based method (Section 5.4.2). Finally, Section 5.5 and Section 5.6 report the experimental evaluation, including the considered datasets, training protocols, and results.

5.2 Related Works and Background

As discussed in Section 1.1, CFD simulations generate spatial fields of extremely high dimensionality, typically on the order of $p \sim 10^6$–10^8 degrees of freedom per snapshot, depending on mesh resolution and the number of physical quantities stored per cell. This scale renders raw CFD fields impractical for direct use in ML pipelines, both from a computational and a statistical perspective. As a result, dimensionality reduction has long been recognised as a central research direction in CFD, with increasing relevance for data-driven analysis and learning-based methods.

Classical approaches to dimensionality reduction in fluid mechanics are primarily based on modal decompositions, most notably Proper Orthogonal Decomposition and Dynamic Mode Decomposition [92, 102, 103, 73]. These methods reduce dimensionality by exploiting dominant correlation structures in ensembles of flow snapshots, and have proven particularly effective in time-resolved or dynamically evolving flows. In applications dominated by steady configurations or time-averaged fields, however, temporal variability is absent or plays a marginal role. In such settings, the dominant source of dimensionality stems from the high spatial resolution of the discretised flow fields and from geometric variability across samples, which limits the effectiveness and interpretability of classical modal decompositions.

A distinct line of work addresses spatial dimensionality reduction through clustering and flow segmentation. By grouping computational cells or flow snapshots according to similarity in appropriately chosen feature spaces, these approaches identify latent spatial or dynamical structures and partition the flow into a limited number of representative regions or states. When clustering is performed in physics-informed feature spaces, the resulting partitions reflect local balances of the Navier–Stokes equations or dominant physical mechanisms, rather than relying solely on geometric proximity or pointwise similarity. A seminal contribution in this direction is Cluster-Based Reduced-Order Modelling (CROM) [51], which applies k-means clustering to time-resolved flow snapshots to identify a finite set of representative flow states, and models their temporal evolution through a Markov chain, enabling the unsupervised identification of regime transitions. Related works have employed unsupervised segmentation to reveal physically meaningful regions or states by clustering local force

balances, invariant-based descriptors, or turbulence-related quantities. For instance, Callaham et al. [14] combined Gaussian Mixture Models with Sparse PCA to partition CFD fields according to the local balance of Navier–Stokes terms, with the goal of visualizing dominant physical regimes rather than extracting features for downstream inference. Similarly, Saetta et al. [93] used GMM-based clustering to automatically identify homogeneous regions such as boundary layers, shocks, and inviscid zones, avoiding heuristic thresholds and manual intervention. Other studies have focused on the identification of flow regimes and transitions by clustering high-dimensional observations in turbulent or transitional flows. For example, Foroozan et al. [36] applied clustering to DNS data of a transitional boundary layer to embed flow states and analyse their probabilistic transitions, while related approaches exploit vortex-identification criteria [47], turbulence statistics, or anisotropy measures [6, 76] to construct compact and physically meaningful flow representations. More recently, Tran et al. [106] integrated optimal transport distances into autoencoder-based latent embeddings to obtain interpretable variables associated with flow separation and control performance, albeit under the assumption of abundant data and fixed geometries. Together, these works demonstrate that unsupervised approaches become significantly more informative when guided by physical structure. At the same time, they primarily target data exploration, regime identification, or reduced-order modelling, typically relying on large numbers of snapshots and limited geometric variability, rather than addressing inference or prediction tasks across heterogeneous geometries.

In parallel, classical feature extraction pipelines in CFD have traditionally relied on handcrafted descriptors, in which regions of interest and aggregated flow quantities are defined *a priori* based on expert knowledge. Such approaches exploit physical intuition and application-specific insights to construct interpretable features, as discussed in Section 5.5.2. However, they also introduce strong inductive biases and require substantial manual effort to adapt feature definitions across different geometries, flow regimes, or problem settings. As a consequence, handcrafted pipelines exhibit limited scalability in scenarios characterised by pronounced geometric variability or restricted expert availability.

Alongside advances in CFD-specific representations, recent developments in ML architectures have expanded the class of data structures that can be directly processed by learning models. In particular, set-based learning architectures have been proposed to operate on unordered and variable-size collections of elements, enforcing permutation invariance without assuming a predefined ordering or topology. Models such as PointNet [19] and its extensions, including attention-based architectures like the Point Transformer [114], were originally developed for point-cloud analysis but exemplify a broader class of models capable of learning from structured, non-Euclidean data. These architectural

properties are potentially relevant in CFD, where data are often naturally organised as collections of spatial regions or entities rather than fixed grids with a canonical ordering.

Finally, geometric alignment and mesh morphing techniques constitute a long-standing line of research for handling geometric variability in computational simulations. In CFD, mesh morphing has traditionally been employed in shape optimisation and aerodynamic design, where controlled geometric deformations are used to explore parametric variations while preserving mesh quality, reflecting the strong sensitivity of aerodynamic performance to shape changes [43, 18, 15]. Beyond aerodynamic optimisation, morphing techniques are widely adopted in moving boundary problems, where solid–fluid interfaces undergo prescribed or induced displacements, such as in valves, pistons, or turbomachinery blades [44]. Another major application area is fluid–structure interaction, where computational meshes deform in response to structural displacements, ranging from weakly coupled rigid-body motions to strongly coupled phenomena such as vortex-induced vibrations and aeroelastic effects [101]. Beyond shape modification, morphing has also been employed as a normalisation mechanism, mapping heterogeneous simulations onto a common reference geometry to facilitate comparison and analysis, as demonstrated for instance in the MMGP framework [16]. Such techniques typically rely on the availability of meaningful geometric correspondences, for which Functional Maps [80] and subsequent extensions incorporating descriptor-aware regularisation and multiresolution refinement [91, 70] can provide a robust and flexible formulation, as already discussed in Section 4.

Within this Chapter, several of the approaches and concepts reviewed above are leveraged as methodological building blocks for feature extraction from high-dimensional CFD data, providing the background for the techniques discussed in the following Sections.

5.3 Region Definition and Feature Extraction: Problem Formulation

Following the inference problem introduced in Section 1.3 and the augmentation problem described in Section 4.3, we now focus on the problem of feature extraction from high-dimensional CFD data. Given that each simulation produces a large and heterogeneous flow field, learning a model K that predicts a target quantity $Y_i \in \mathcal{Y}$, either continuous (regression) or discrete (classification), requires mapping $\mathbf{F}_i$ to a compact and structured representation that can be effectively processed by K.

Let us consider the dataset

$$\mathcal{D} = \{(S_i, M_i, \mathbf{F}_i, Y_i)\}_{i=1}^{N},$$

where S_i denotes the surface bounding the computational domain $\Omega_i \subset \mathbb{R}^3$, M_i the corresponding volumetric mesh discretizing Ω_i, $\mathbf{F}_i \in \mathbb{R}^{n_i \times H}$ the CFD data matrix containing the H quantities computed at each cell of M_i (e.g., velocity components, pressure, or turbulence terms), and Y_i the associated target label. Each $\mathbf{F}_i$ is typically high-dimensional ($n_i \sim 10^7$), making it impractical to use directly as input to the learning model K.

To derive a compact and structured representation from $\mathbf{F}_i$, we introduce a region extraction operator Φ that segments the volumetric domain Ω_i into a collection of r_i flow regions:

$$\Phi : (S_i, M_i, \mathbf{F}_i) \longrightarrow \{R_{i,j}\}_{j=1}^{r_i}, \quad \text{with} \quad R_{i,j} \subset M_i,\ r_i \ll n_i.$$

The j-th region $R_{i,j}$ identifies a spatial subset of the i-th flow field over which aggregated descriptors are computed (e.g., regional averages and geometric properties). The concatenation of the features extracted from all regions $\{R_{i,j}\}_{j=1}^{r_i}$ defines the feature representation $\mathbf{P}_i$. Regions can be identified either automatically, through unsupervised criteria, or manually defined by an expert.

The overall feature extraction process can be compactly summarised as:

$$(S_i, M_i, \mathbf{F}_i) \xrightarrow{\Phi} \{R_{i,j}\}_{j=1}^{r_i} \xrightarrow[Features]{Regional} \mathbf{P}_i,$$

and the resulting dataset is defined by the pairs $\{(\mathbf{P}_i, Y_i)\}_{i=1}^{N}$, where each $\mathbf{P}_i$ encodes the relevant physical and geometrical information of $\mathbf{F}_i$ in a low-dimensional, structured form suitable for learning.

Our goal is to design a region extraction operator Φ and a corresponding set of extracted features such that

$$\begin{cases} \mathbf{P}_i \text{ is compact and physically meaningful,} \\ \mathbf{P}_i \text{ is consistent and comparable across varying geometries } S_i, \\ \mathbf{P}_i \text{ preserves the flow structures relevant to the target } Y_i, \end{cases}$$

This formulation aims to produce a compact, consistent, and physics-aware representation $\mathbf{P}_i$ for each simulation, ensuring that $K(\mathbf{F}_i) \approx K'(\mathbf{P}_i)$ for all $i = 1, \ldots, N$.

5.4 Methodology

Following the formulation introduced in Section 5.3, the core component of the proposed framework is a region extraction operator Φ that segments each volumetric domain Ω_i into a set of regions $\{R_{i,j}\}_{j=1}^{r_i}$. Within each region, representative flow descriptors are computed and aggregated into a feature set $\mathbf{P}_i$, which is subsequently used as input to K.

We propose two alternative implementations of the region extraction operator Φ, namely a *Clustering-based* method and a *Morphing-based* method, illustrated in the right and left parts of Figure 5.1 respectively. The two approaches differ fundamentally in how regions are defined. In the *Clustering-based* method, regions are identified in a data-driven manner by exploiting the physics encoded in the governing equations of the flow. In contrast, in the *Morphing-based* method, regions are expert-defined on a single reference geometry and consistently transferred across simulations by means of a geometric alignment procedure based on RBFs [9].

In the *Clustering-based* method [68], the operator Φ is constructed as follows. Starting from the CFD data matrix $\mathbf{F}_i$, the raw flow variables are inserted into the governing equations of the CFD solver [83], yielding a set of N derived quantities corresponding to the individual terms of the equations (top part of the right box in Figure 5.1). Collecting these quantities for all cells of the mesh M_i results in the matrix

$$\mathbf{C}_i \in \mathbb{R}^{n_i \times D},$$

where each row represents a CFD cell and encodes its local physical balance. A clustering algorithm is then applied to the rows of $\mathbf{C}_i$ (central part of the right box in Figure 5.1), partitioning the domain into regions $\{R_{i,j}\}_{j=1}^{r_i}$ characterised by coherent physical behaviour. Within each region, aggregated flow descriptors are computed, leading to the feature set $\mathbf{P}_i$ (bottom part of the right box in Figure 5.1), which is then processed by the learning model K.

In the *Morphing-based* method, the region extraction operator Φ relies on an explicit geometric alignment of each simulation onto a common reference domain Ω^*. Given a CFD simulation $\mathbf{F}_i$ defined on the mesh M_i, a morphing operator

$$\Pi_i : (\mathbf{F}_i, M_i, M^*) \longrightarrow \mathbf{F}_i^*$$

is used to map the flow field onto the reference mesh M^* (top and central parts of the left box in Figure 5.1). This procedure partially follows the approach presented in [16] and requires a point-to-point correspondence $\mathcal{M}_i$ between the boundaries $\partial\Omega_i$ and $\partial\Omega^*$. Boundary displacements are computed from these correspondences and smoothly propagated throughout the domain using RBFs. All flow quantities are then interpolated from Ω_i to Ω^* using a finite element basis. As a result, each morphed flow field $\mathbf{F}_i^* \in \mathbb{R}^{n^* \times H}$ is defined on the same reference discretisation M^*. On this reference domain, a fixed set of regions $\{R_j^*\}_{j=1}^{r^*}$ is defined once by a domain expert, and consistently used to extract features from all morphed flow fields $\mathbf{F}_i^*$, yielding feature sets $\mathbf{P}_i$ that are directly comparable across simulations (bottom part of the left box in Figure 5.1). The resulting representations are finally processed by a standard ML model K.

In the following sections, we describe the two proposed methods in detail and analyse their respective roles within the overall feature extraction framework.

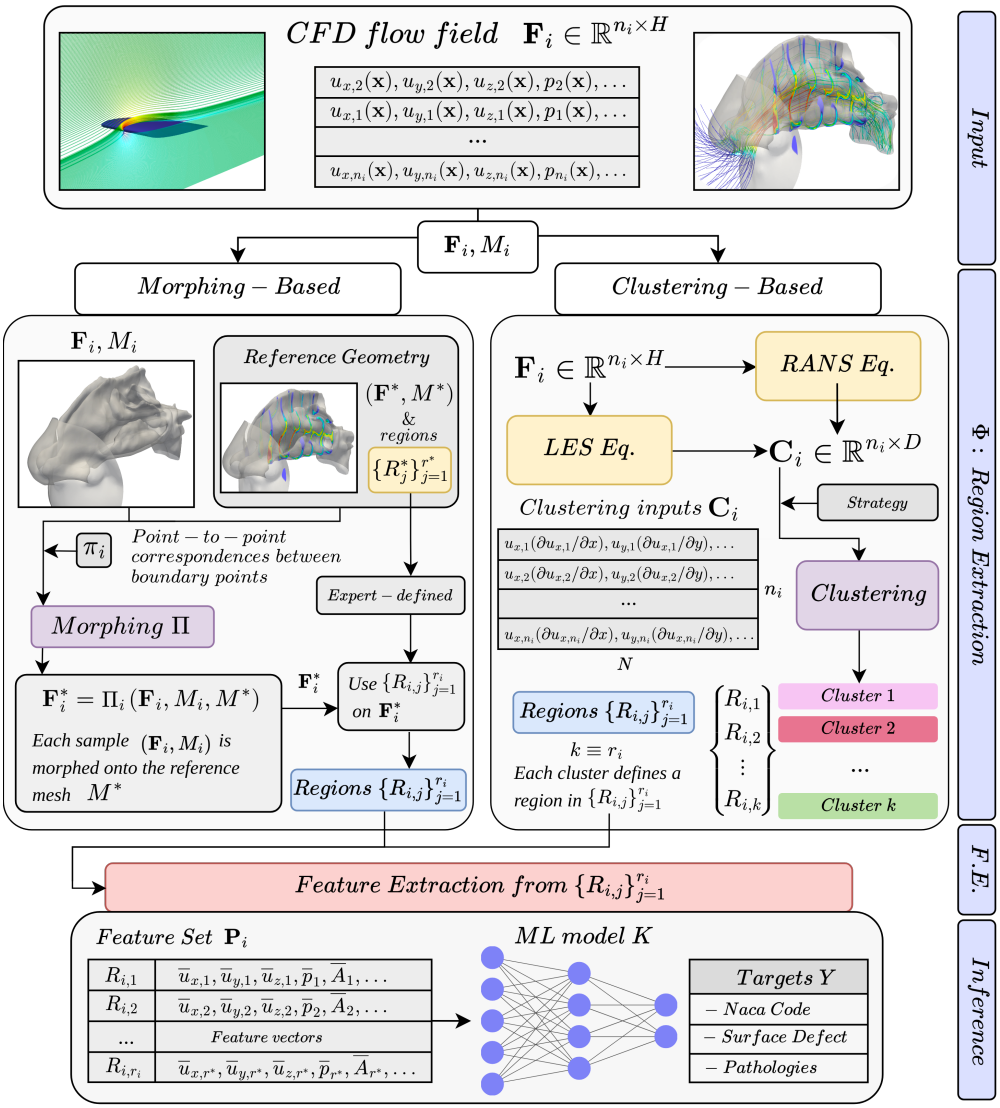


Figure 5.1: Overview of the two region extraction operators Φ. In the *Clustering-based* method (right), regions $\{R_{i,j}\}_{j=1}^{r_i}$ are identified by clustering a matrix $\mathbf{C}_i$ that collects physically meaningful quantities derived from the governing equations, resulting in coherent flow regions characterized by similar local physical behavior. In the *Morphing-based* method (left), each flow field $\mathbf{F}_i$ is mapped onto a common reference mesh M^* through a morphing operator Π. Regions $\{R_j^*\}_{j=1}^{r^*}$ are defined once on the reference geometry and then consistently applied to all morphed flow fields. In both cases, regional descriptors are computed within each region and aggregated into the feature set $\mathbf{P}_i$, which is used as input to a learning model K.

5.4.1 Clustering-Based Method

To automatically extract flow regions $\{R_{i,j}\}_{j=1}^{r_i}$ through the operator Φ, without relying on geometry-specific landmarks or manually defined anatomical sections, we adopt a physics-based clustering strategy. The key idea is to derive, at each cell of the computational domain, quantities directly related to the governing equations of the CFD model, and to use these quantities to identify coherent flow regions through unsupervised learning.

As introduced in Chapter 6, in this thesis, either Reynolds-averaged Navier–Stokes (RANS) or Large Eddy Simulation (LES) formulations of the incompressible Navier–Stokes equations [83] are adopted for CFD simulations. In both cases, the governing equations provide a natural decomposition of the flow dynamics into physically meaningful contributions, such as advection, diffusion, pressure gradients, and turbulence-related terms. In the following, these equation terms are exploited to construct a physics-based representation of each flow field, which serves as input to the clustering algorithm.

Clustering Inputs Rather than applying clustering directly to the raw CFD data matrix $\mathbf{F}_i \in \mathbb{R}^{n_i \times H}$, where each row corresponds to a CFD cell and stores primitive flow variables, we operate on a transformed representation $\mathbf{C}_i \in \mathbb{R}^{n_i \times D}$. This representation is obtained by mapping the raw flow variables through the governing equations of the CFD model, following a strategy inspired by [14].

More precisely, each row of $\mathbf{F}_i$ corresponds to a vector $f_j^i \in \mathbb{R}^H$, containing quantities such as velocity components, pressure, and turbulence-related variables at the j-th cell of the mesh M_i. From these variables, we compute a vector $c_j^i \in \mathbb{R}^D$ whose components correspond to the individual terms appearing in the governing equations, evaluated locally at that cell. Collecting these vectors for all cells yields the matrix $\mathbf{C}_i$, which provides a compact, physics-based description of the flow field. Typically, $D < H$, and the resulting feature space is both lower-dimensional and more structured than the original CFD data.

Clustering is then applied to the rows of $\mathbf{C}_i$, partitioning the computational domain into regions $\{R_{i,j}\}_{j=1}^{r_i}$ that group together cells characterised by a similar local balance of physical contributions. In this way, the segmentation of the domain emerges directly from the governing equations, ensuring that the extracted regions reflect meaningful and physically consistent flow structures. In the following, we describe how the matrices $\mathbf{C}_i$ are constructed for RANS and LES simulations, respectively.

Construction of the matrix $\mathbf{C}_i$ The RANS and LES formulations adopted in this work are detailed in Chapter 6. Here, we briefly recall the governing equations to clarify how the clustering inputs $\mathbf{C}_i$ are constructed.

In the RANS setting, all flow variables are decomposed into mean and fluctuating components through time averaging. For instance, the velocity field is written as $\boldsymbol{U} = \overline{\boldsymbol{U}} + \boldsymbol{u}'$, with analogous decompositions holding for pressure and related quantities. The incompressible RANS momentum equations read

$$\underbrace{\nabla \cdot \left(\overline{\boldsymbol{U}} \otimes \overline{\boldsymbol{U}}\right)}_{\text{Advection}} = - \underbrace{\frac{1}{\rho}\nabla \overline{p}}_{\text{Pressure gradient}} + \underbrace{\nu \nabla^2 \overline{\boldsymbol{U}}}_{\text{Laminar diffusion}} + \underbrace{\nabla \cdot \left(\nu_t \nabla \overline{\boldsymbol{U}}\right)}_{\text{Turbulent diffusion}} - \underbrace{\nabla \left(\tfrac{2}{3}\overline{k}_t\right)}_{\text{TKE}}, \tag{5.1}$$

where ν is the kinematic viscosity, ρ is the density (constant for incompressible flows), ν_t denotes the turbulent viscosity provided by the Spalart–Allmaras model [100], and $\overline{k}_t$ is the Turbulent Kinetic Energy (TKE). Accordingly, for RANS simulations, $\mathbf{C}_i$ collects the Cartesian components of the advection, pressure gradient, laminar diffusion, turbulent diffusion, and TKE terms.

In the LES setting, the incompressible Navier–Stokes equations are instead filtered in space, rather than averaged in time ($\widetilde{\cdot}$ operator). Spatial filtering separates the flow field into resolved large-scale components and unresolved subgrid fluctuations by applying a low-pass filter with a characteristic length scale associated with the local grid resolution. As a result, the governing equations describe the dynamics of the resolved-scale variables. The filtered incompressible Navier–Stokes equations read

$$\underbrace{\frac{\partial \widetilde{\boldsymbol{U}}}{\partial t}}_{\text{Unsteadiness}} + \underbrace{\nabla \cdot \left(\widetilde{\boldsymbol{U}} \otimes \widetilde{\boldsymbol{U}}\right)}_{\text{Advection}} = \underbrace{\nu \nabla^2 \widetilde{\boldsymbol{U}}}_{\text{Laminar diffusion}} - \underbrace{\frac{1}{\rho}\nabla \widetilde{p}}_{\text{Pressure gradient}} - \underbrace{\nabla \cdot \boldsymbol{\tau}^{\text{SGS}}}_{\text{SGS stress}}, \tag{5.2}$$

where $\widetilde{\boldsymbol{U}}$ and $\widetilde{p}$ denote the filtered velocity and pressure fields, and $\boldsymbol{\tau}^{\text{SGS}}$ is the subgrid-scale stress tensor, here modelled using the WALE closure [27]. Although LES are inherently unsteady, all quantities used for feature extraction are time-averaged after the transient phase, so that $\mathbf{C}_i$ represents statistically stationary flow behavior. For LES simulations, $\mathbf{C}_i$ includes the Cartesian components of the advection, pressure gradient, laminar diffusion, and SGS stress divergence. Once $\mathbf{C}_i$ has been assembled, the next step is to cluster its rows in order to identify regions characterised by coherent physical behaviour.

Clustering Algorithm To partition the rows of $\mathbf{C}_i$, we adopt a Bayesian Gaussian Mixture Model (BGMM) [34]. Unlike standard GMMs, the BGMM employs a variational Bayesian inference framework that automatically infers the effective number of clusters from the data. Each cluster is modelled using a full covariance matrix, allowing the method to capture correlations among the physical terms appearing in (5.1) or (5.2). This enables the identification of regions associated with distinct flow phenomena, such as boundary layers, shear layers, separated regions, and wakes [14].

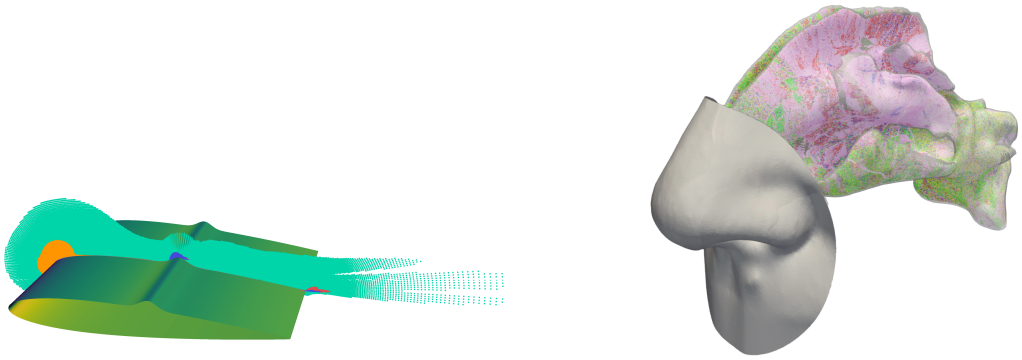
For numerical stability, an upper bound $k_{\max}$ on the number of clusters is prescribed, and the BGMM is initialised using the centroids obtained from a k-means clustering with $k_{\max}$ components. During variational inference, redundant components are automatically pruned, resulting in an effective number of clusters $k \leq k_{\max}$ for each simulation.

Clustering Strategies Clustering $\mathbf{C}_i$ yields the set of regions $\{R_{i,j}\}_{j=1}^{r_i}$, where $r_i = k_i$ denotes the number of clusters inferred for the i-th simulation (for notational simplicity, we refer to this quantity as k when no ambiguity arises). However, neither the number nor the ordering of the clusters is guaranteed to be consistent across simulations. As a consequence, the corresponding feature representations $\mathbf{P}_i$ form unordered sets with potentially varying cardinality, which cannot be directly processed by standard supervised learning models requiring fixed-size inputs. To address this issue, we adopt two alternative clustering strategies:

- *Free clustering* (*C-FREE*). In this approach, clustering is performed independently for each simulation. The resulting feature representation $\mathbf{P}_i$ is an unordered set whose cardinality may vary across cases. To process such representations, we employ permutation-invariant set-learning models [114]. This strategy maximises flexibility and allows the clustering to adapt to different flow regimes and geometrical configurations. However, the aggregation operations used by set-based models may smooth out fine-grained information, potentially affecting predictive performance.

- *Clustering propagation* (*C-PROP*). In this approach, a reference simulation $\mathbf{F}^*$ is selected and clustered using the *C-FREE* strategy. The resulting reference clusters are then propagated to all other simulations by assigning each row of $\mathbf{C}_i$ to the nearest reference cluster in the feature space. This distance-based matching is efficiently implemented using a k-d tree [7]. Since the rows of $\mathbf{C}_i$ encode local physical balances, the propagation preserves the physical meaning of the clusters while enforcing consistent ordering and cardinality across simulations. This enables the use of standard supervised learning models, such as MLPs, at the cost of reduced adaptability to flow features not present in the reference case.

In both cases, the clustering-based approach yields a physics-guided segmentation of the flow field that is applicable in the same manner across different geometries, as it is driven by the governing equations rather than by geometry-specific constraints. A visual example of segmented flow fields is shown in Figure 5.2. We now introduce the *Morphing-based* method, which reintroduces expert knowledge and enforces spatial consistency across geometries.



(a) Clustering-based segmentation for a defected airfoil. The resulting regions correspond to distinct flow regimes, including the leading edge (LE) stagnation zone, the perturbation induced by the surface defect (bump), the trailing edge (TE), and the downstream wake.

(b) Clustering-based segmentation in the nasal airway. Due to the complex and highly irregular geometry, the resulting regions are less directly interpretable, reflecting distributed flow patterns rather than clearly separable canonical structures.

Figure 5.2: Examples of clustering-based region definition across different application domains. While the method consistently partitions the flow field based on local physical behaviour, the interpretability of the resulting regions depends on the geometric complexity of the domain.

5.4.2 Morphing-based Method

In the *Morphing-based* method, the feature extraction operator Φ restores the use of regions defined by domain experts, as originally adopted in Chapter 4. Instead of specifying the regions $\{R_{i,j}\}_{j=1}^{r_i}$ independently for each geometry, the regions are defined once on a reference mesh M^*, yielding a fixed set $\{R_j^*\}_{j=1}^{r^*}$ (see the top part of the left box in Figure 5.1). To consistently apply this partition across different geometries, each CFD flow field $\mathbf{F}_i$, computed on its original mesh M_i, is mapped onto the reference mesh M^* through a morphing operator, denoted by Π_i, resulting in a morphed field $\mathbf{F}_i^*$.

More specifically, we transfer $\mathbf{F}_i \in \mathbb{R}^{n_i \times H}$, originally defined on M_i, onto a common reference domain Ω^* discretised into n^* cells forming the mesh M^*. Without such an alignment, the CFD fields $\{\mathbf{F}_i\}_{i=1}^{N}$ would not be directly comparable, preventing the consistent application of the same expert-defined regions across simulations. By mapping both the geometry M_i and the associated CFD data $\mathbf{F}_i$ onto the reference mesh M^*, we obtain a consistent CFD data matrix $\mathbf{F}_i^* \in \mathbb{R}^{n^* \times H}$, defined on the same spatial support, namely the same boundary $\partial\Omega^*$ and computational mesh M^* on which the regions $\{R_j^*\}_{j=1}^{r^*}$ are defined.

The morphing operator is defined as

$$\Pi_i : (\mathbf{F}_i, M_i, M^*) \to \mathbf{F}_i^*,$$

and is decomposed as

$$\Pi_i = \mathcal{I}_i \circ \mathcal{T}_i,$$

where $\mathcal{T}_i$ performs a non-rigid geometric deformation of the mesh, and $\mathcal{I}_i$ transfers the CFD fields onto the reference mesh. In particular,

$$\mathcal{T}_i : (M_i, M^*, \mathcal{M}_i) \to M_i'$$

is a geometric transformation that deforms M_i so that its boundary nodes match those of the reference mesh M^*, yielding a deformed mesh M_i' whose boundary coincides with $\partial\Omega^*$, while its internal discretisation is different from that of M^*. The operator $\mathcal{T}_i$ requires as input a point-to-point correspondence defined on the boundaries of the two domains, which we denote by

$$\pi_i : \partial\Omega_i \to \partial\Omega^*.$$

This correspondence defines boundary displacements that are propagated to the interior via RBF interpolation, as illustrated in Figure 5.3. The mapping π_i is constructed independently of the deformation step and is determined by the specific application considered. In the biomedical scenario discussed in Chapter 4, π_i is instantiated by the functional correspondence $\mathcal{M}_i$ obtained via functional maps, as detailed in Section 4.4.2 and Appendix A. Other strategies can be adopted in different settings, such as the airfoil case considered in the following.

From π_i, boundary displacements are computed and smoothly propagated into the interior of the mesh using RBF interpolation [9]. Subsequently, a projection operator

$$\mathcal{I}_i : (\mathbf{F}_i, M_i') \to \mathbf{F}_i^*$$

transfers the CFD field $\mathbf{F}_i$ from the morphed mesh M_i' to the common reference mesh M^*, ensuring that the CFD data are defined on the same spatial support. This final step enables the consistent application of the expert-defined regions $\{R_j^*\}_{j=1}^{r^*}$ across all samples and allows for coherent feature extraction on a common spatial domain. In the following, we detail how the two operators $\mathcal{T}_i$ and $\mathcal{I}_i$ are defined and implemented.

$\mathcal{T}_i$: mesh deformation to a common shape Let $\mathbf{x}_j$ denote the coordinates of the j-th node of the mesh M_i. The geometric transformation $\mathcal{T}_i$ applies a displacement field $\mathbf{d}_i(\mathbf{x})$ such that

$$\mathbf{x}_j' = \mathbf{x}_j + \mathbf{d}_i(\mathbf{x}_j),$$

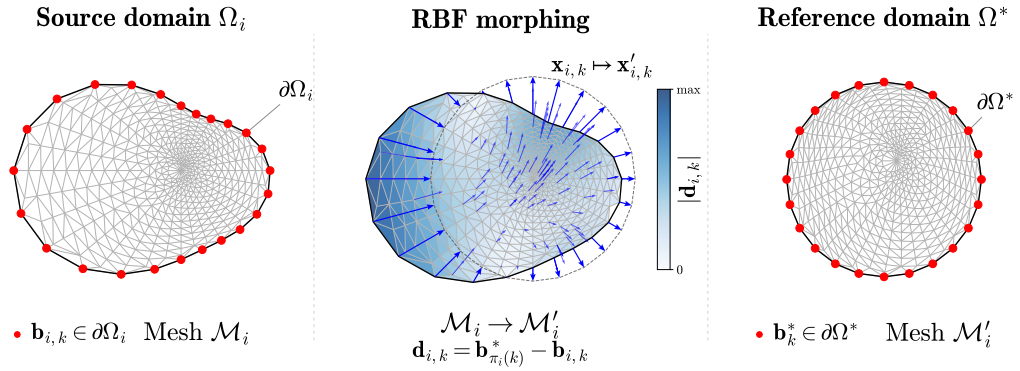


Figure 5.3: Illustration of the boundary-driven morphing operator $\mathcal{T}_i$ based on radial basis function (RBF) interpolation. Left: source domain Ω_i with mesh $\mathcal{M}_i$ and boundary nodes $\mathbf{b}_{i,k} \in \partial\Omega_i$. Center: a point-to-point correspondence $\pi_i : \partial\Omega_i \to \partial\Omega^*$ defines boundary displacements $\mathbf{d}_{i,k} = \mathbf{b}^*_{\pi_i(k)} - \mathbf{b}_{i,k}$, which are smoothly propagated to the interior through RBF interpolation, yielding a non-rigid deformation of the mesh (visualised via the magnitude $|\mathbf{d}_{i,k}|$). Right: reference domain Ω^* with mesh $\mathcal{M}^*$ and boundary nodes $\mathbf{b}^*_k \in \partial\Omega^*$. The resulting transformation aligns the source boundary $\partial\Omega_i$ to the reference boundary $\partial\Omega^*$ while preserving a smooth volumetric deformation.

yielding a deformed mesh M'_i whose boundary coincides with the boundary $\partial\Omega^*$ of the reference mesh M^*. The internal discretisation of M'_i is obtained by interpolating the boundary displacements into the volume and therefore does not, in general, coincide with the internal discretisation of M^*. In particular, following the mesh deformation method proposed by De Boer et al. [9], the displacement field $\mathbf{d}_i(\mathbf{x})$ is interpolated from the boundary to the interior of the domain using RBFs. Let $\{\mathbf{b}_{i,k}\}_{k=1}^{n_b}$ be the set of boundary nodes discretising $\partial\Omega_i$, and let $\{\mathbf{b}^*_{k'}\}_{k'=1}^{n^*_b}$ be the boundary nodes of the reference mesh M^*. The correspondence between the two sets is defined through the mapping

$$\pi_i : \partial\Omega_i \to \partial\Omega^*,$$

such that $k' = \pi_i(k)$ for all $k = 1, \ldots, n_b$. The prescribed displacement of the k-th boundary node is therefore defined as

$$\mathbf{d}^b_{i,k} = \mathbf{b}^*_{\pi_i(k)} - \mathbf{b}_{i,k},$$

indicating how each boundary node must be shifted to match its corresponding node on the reference boundary $\partial\Omega^*$.

In practice, the correspondence operator π_i is obtained through domain-specific knowledge. In aerodynamic applications, correspondences are defined using a set of unambiguous geometric landmarks, including the leading edge,

trailing edge, uniformly spaced points along the pressure and suction sides, and the farfield boundary. For biomedical applications involving the human upper airways, π_i is corresponds with the functional correspondence $\mathcal{M}_i$, computed following the method proposed by Cosmo et al. [22] and refined via the ZoomOut procedure [70], as detailed in Chapter 4.

Given the prescribed boundary displacements $\mathbf{d}_i^b$, the displacement field inside the domain is obtained by RBF interpolation (for readability, the subscript i is omitted):

$$\mathbf{d}(\mathbf{x}) = \sum_{k=1}^{n_b} \alpha_k \, \psi(\|\mathbf{x} - \mathbf{b}_k\|) + \boldsymbol{W}(\mathbf{x}). \tag{5.3}$$

Here, α_k are interpolation coefficients, $\boldsymbol{W}(\mathbf{x}) = [1,\, x,\, y,\, z]\boldsymbol{\beta}$ is a low-order polynomial term ensuring the exact reproduction of rigid transformations (translations and rotations), and $\psi(\cdot)$ is a compactly supported C^2 radial basis function defined as

$$\psi(r) = \begin{cases} (1-r)^4(4r+1), & r \le 1, \\ 0, & r > 1, \end{cases} \qquad r = \frac{\|\mathbf{x} - \mathbf{b}\|}{r_s}, \tag{5.4}$$

where $r_s > 0$ denotes the support radius. Rather than being fixed, r_s is adapted to a characteristic length scale of the problem, allowing the interpolation to remain robust across applications characterised by different spatial scales.

The interpolation coefficients $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are obtained by enforcing the interpolation conditions at the boundary nodes together with orthogonality constraints between the radial and polynomial terms, leading to the linear system

$$\begin{bmatrix} \boldsymbol{\Psi} & \boldsymbol{W}_b \\ \boldsymbol{W}_b^T & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \mathbf{d}^b \\ \mathbf{0} \end{bmatrix}, \tag{5.5}$$

where $\boldsymbol{\Psi}_{ij} = \psi(\|\mathbf{x}_i^b - \mathbf{x}_j^b\|)$ and $\boldsymbol{W}_b$ contains the polynomial basis evaluated at the boundary nodes. The orthogonality condition $\boldsymbol{W}_b^T\boldsymbol{\alpha} = \mathbf{0}$ ensures the uniqueness of the interpolant and the exact recovery of rigid transformations. Since the system only involves boundary nodes, it can be efficiently solved using standard direct or iterative solvers [13].

$\mathcal{I}_i$: projection on common mesh After the geometry has been deformed by $\mathcal{T}_i$, the CFD fields $\mathbf{F}_i$, defined on the morphed mesh M_i', must be transferred onto the reference mesh M^*. Since M_i' and M^* do not share nodes or connectivity, a projection operator is required to ensure that all CFD data are represented on the same spatial support. An example can be seen in Figure 5.4

Following [16], we adopt a finite element (FEM) projection formulated in a variational setting, which guarantees both stability and consistency when

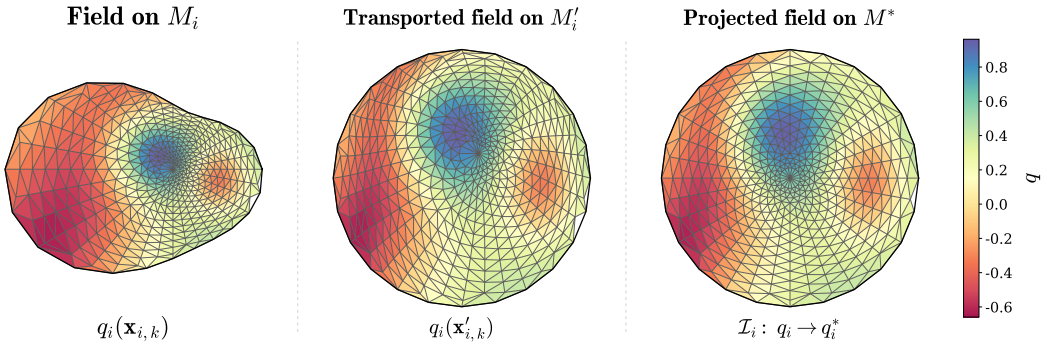


Figure 5.4: Transfer of a scalar field q_i across non-conforming meshes. Left: the field $q_i(\mathbf{x}_{i,k})$ is defined on the source mesh M_i. Center: after the geometric morphing, the same field is transported onto the deformed mesh M_i', yielding $q_i(\mathbf{x}'_{i,k})$. Right: the transported field is projected onto the reference mesh M^* through the operator $\mathcal{I}_i$, resulting in the aligned field q_i^*.

transferring solutions between non-conforming meshes. A Lagrange $\mathbb{P}_1$ finite element basis is employed. For a generic scalar field q_i defined on M_i', we write

$$q_i(\mathbf{x}) = \sum_{I=1}^{n_i} Q_I^i \, \lambda_I^i(\mathbf{x}),$$

where Q_I^i are the nodal values and $\{\lambda_I^i\}$ denotes the finite element basis associated with M_i'. The projection onto the reference mesh M^* is defined as

$$q_i^*(\mathbf{x}) = \sum_{J=1}^{n^*} q_i(\mathbf{x}_J^*) \, \lambda_J^*(\mathbf{x}) = \sum_{I=1}^{n_i} \sum_{J=1}^{n^*} Q_I^i \, \lambda_I^i(\mathbf{x}_J^*) \, \lambda_J^*(\mathbf{x}),$$

where $\{\lambda_J^*\}$ denotes the finite element basis associated with M^* and $\mathbf{x}_J^*$ the coordinates of its nodes. Applying this projection to all the CFD fields of $\mathbf{F}_i$ defines the operator $\mathcal{I}_i$ and yields the aligned CFD matrix

$$\mathbf{F}_i^* = \Pi_i(\mathbf{F}_i, M_i, M^*) = \mathcal{I}_i\Big(\mathbf{F}_i, \mathcal{T}_i(M_i, M^*, \mathcal{M}_i)\Big).$$

The combined action of $\mathcal{T}_i$ and $\mathcal{I}_i$ defines the morphing operator Π_i and ensures that both geometry and CFD fields are consistently represented on the same reference mesh M^*, resulting in a unified dataset $\{\mathbf{F}_i^*\}_{i=1}^N$ on which the expert-defined regions $\{R_j^*\}_{j=1}^{r^*}$ can be directly and consistently applied.

5.4.3 Extraction of Features from Regions $\{R_{i,j}\}_{j=1}^{r_i}$

Once the flow regions $\{R_{i,j}\}_{j=1}^{r_i}$ have been identified through the region extraction operator Φ, either by means of clustering or via expert-defined regions on

morphed data, the next step consists in constructing the feature set $\mathbf{P}_i$ (bottom part of Figure 5.1), which serves as input to the ML model K. The purpose of this stage is to compress the high-dimensional CFD data into compact, informative, and comparable descriptors, while preserving the physical and geometric information relevant to the inference task.

For both the *Clustering-based* and the *Morphing-based* methods, features are computed independently within each region by aggregating flow-related and geometrical quantities. In particular, we extract regional features by computing weighted averages of selected flow quantities (in the same fashion as in 4.5), together with geometric descriptors, namely the total area (or volume in 3D) of the region and the coordinates of its centroid. These quantities are introduced to preserve the spatial information, which is essential in CFD-based representations where the location of flow structures plays a key role.

Collecting all regional descriptors, the feature extraction process maps the CFD data $\mathbf{F}_i$ to a compact set of feature vectors

$$\mathbf{P}_i = \left\{ \mathbf{p}_j^i \in \mathbb{R}^l \;\middle|\; j = 1, \ldots, r_i \right\}, \tag{5.6}$$

where $r_i \ll n_i$ is the number of regions and $l = \mathcal{O}(D)$ is the dimensionality of each regional feature vector. The complete list of features composing $\mathbf{P}_i$ for RANS and LES simulations is reported in Table 5.1.

In the *Clustering-based* method, regions correspond to the k clusters estimated by the BGMM described in Section 5.4.1. This generalises the region-based feature extraction strategy adopted in [97], replacing manually defined spatial regions with data-driven clusters that adapt to the local flow structures of each simulation.

In the *Morphing-based* method, expert knowledge is explicitly reintroduced into the feature extraction process. In this case, the clusters are replaced by regions defined once on the reference mesh M^* and then consistently applied to all simulations through the morphing operator Π_i. Depending on the application, these regions may correspond to probe-like measurements in aerodynamic configurations or to anatomically meaningful cross-sections in biomedical flows [66]. Compared to clustering, which prioritises adaptivity to local flow structures, expert-defined regions provide standardised and interpretable descriptors at the cost of reduced sensitivity to localised or case-specific flow phenomena.

5.4.4 Inference Models

The choice of the inference model K depends on the structural properties of the feature sets $\mathbf{P}_i$, which in turn are determined by how the regions $\{R_{i,j}\}_{j=1}^{r_i}$ are defined. In particular, clustering-based and morphing-based region extraction lead to feature representations with different degrees of ordering, cardinality, and spatial consistency, which directly guide the selection of the learning

Features in $\mathbf{P}_i$		
Category	**RANS (2D)**	**LES (3D)**
Avg. Flow variables	$\bar{u}_x$, $\bar{u}_y$, $\bar{p}$, ν_t	$\tilde{u}_x$, $\tilde{u}_y$, $\tilde{u}_z$, $\tilde{p}$, ν_{SGS}
Avg. Advection	$\nabla\cdot(\bar{\boldsymbol{U}}\otimes\bar{\boldsymbol{U}})\big\vert_{x,y}$	$\nabla\cdot(\tilde{\boldsymbol{U}}\otimes\tilde{\boldsymbol{U}})\big\vert_{x,y,z}$
Avg. Laminar diffusion	$\nu\,\nabla^2\bar{\boldsymbol{U}}\big\vert_{x,y}$	$\nu\,\nabla^2\tilde{\boldsymbol{U}}\big\vert_{x,y,z}$
Avg. Turbulent diffusion	$\nabla\cdot(\nu_t\nabla\bar{\boldsymbol{U}})\big\vert_{x,y}$, $\nabla\left(\frac{2}{3}\bar{k}_t\right)\big\vert_{x,y}$	$\nabla\cdot\boldsymbol{\tau}_{\text{SGS}}\big\vert_{x,y,z}$
Avg. Pressure gradient	$\frac{1}{\rho}\nabla\bar{p}\big\vert_{x,y}$	$\frac{1}{\rho}\nabla\tilde{p}\big\vert_{x,y,z}$
Geometrical	Centroids (x, y), Area	Centroids (x, y, z), Volume

Table 5.1: Summary of the features composing $\mathbf{P}_i$. Flow-related features are computed as regional averages (*Avg.*) according to (4.5), while geometric features encode the spatial extent and location of each region. On the left, RANS (2D, time-averaged); on the right, LES (3D, spatially filtered). The notation $\big\vert_x$, $\big\vert_{x,y}$, $\big\vert_{x,y,z}$ indicates the retained vector components. In the LES case, the subgrid-scale stress term $\boldsymbol{\tau}_{\text{SGS}}$ is reported explicitly, although it is modelled through an eddy viscosity ν_{SGS} [27].

architecture of K.

***Clustering-based* method** When regions are obtained through clustering, the inference model depends on the clustering strategy described in Section 5.4.1. In the *C-PROP* setting, clusters are propagated from a reference simulation, ensuring that the resulting regional feature vectors $\mathbf{p}_j^i$ in (5.6) have a fixed ordering and cardinality across simulations. In this case, the feature set $\mathbf{P}_i$ can be represented as a single vector by concatenating all regional descriptors, and a standard Multi-Layer Perceptron (MLP) is employed as the inference model K.

In contrast, the *C-FREE* strategy produces feature sets with variable cardinality and no predefined ordering, as clustering is performed independently for each simulation. As a result, the feature sets $\mathbf{P}_i$ cannot be aligned across samples and must be processed as unordered sets. To handle this setting, we adopt a Point Transformer (PT) [114], a permutation-invariant architecture specifically designed for learning from point sets. Since each regional feature vector $\mathbf{p}_j^i$ includes the centroid coordinates of the corresponding region (see Section 5.4.3), the remaining components can be interpreted as point-wise features. This formulation allows the PT to exploit self-attention mechanisms to construct spatially aware representations of the flow regions, which is particularly

relevant in CFD applications where spatial relationships play a fundamental role in governing the flow dynamics.

***Morphing-based* method** When features are extracted from morphed flow fields $\{\mathbf{F}_i^*\}_{i=1}^N$, all simulations are aligned to the same reference geometry M^* through the morphing operator Π_i, as described in Section 5.4.2. In this setting, regions $\{R_j^*\}_{j=1}^{r^*}$ are defined once on the reference mesh and consistently applied to all simulations. As a consequence, the resulting feature sets $\mathbf{P}_i$ have fixed size and consistent ordering across cases.

This regular structure makes standard feed-forward architectures a natural choice. In particular, we employ an MLP as the inference model K, which can directly exploit the standardized input representation without requiring permutation invariance or set-based processing.

5.5 Experiments

This section presents the experimental evaluation of the feature extraction and inference framework introduced in the previous sections. The experiments are designed to assess how different region extraction strategies Φ and the resulting feature representations $\mathbf{P}_i$ affect the performance of supervised learning models in CFD-based inference tasks.

We consider three inference problems of increasing complexity, corresponding to the scenarios introduced in Chapter 3: *i*) airfoil shape identification, *ii*) surface defect detection on airfoils, and *iii*) pathology classification in real human upper airways. For each task, we compare alternative feature extraction strategies based on different definitions of flow regions, including physics-based clustering and expert-defined regions transferred through morphing. As a reference, we also include as baseline the feature extraction strategies based on manually defined regions introduced in Section 2.2, which represent the classical approach for learning from CFD data.

We first validate the proposed methods on two-dimensional airfoil datasets, including a publicly available benchmark [96] and an extended version described in Section 6.1. We then consider a more challenging scenario, namely pathology identification in patient-specific upper airways reconstructed from CT scans, using the dataset introduced in Chapter 4 and Section 6.2. This progression from controlled two-dimensional configurations to complex three-dimensional anatomies enables a systematic evaluation of the proposed methods across increasingly demanding CFD regimes.

5.5.1 Tasks and Employed Datasets

Here, we formally introduce the inference tasks considered in the experimental evaluation and fix the notation used throughout the remainder of the Chapter. The computational domains, mesh generation procedures, and CFD simulation setups are described in detail in Chapter 6 and are not repeated here.

Airfoil Shape Identification (*AirNACA*) The *AirNACA* task addresses the problem of identifying the shape of a two-dimensional airfoil directly from its CFD solution. Specifically, we consider the family of NACA four-digit airfoils (see Chapter 3) and train a regressor K to predict the corresponding NACA code from the extracted feature representation $\mathbf{P}_i$. Since the last two digits of the NACA code jointly define the airfoil thickness, this task reduces to a regression problem over three integer parameters encoding camber magnitude, camber position, and thickness.

The dataset consists of steady RANS simulations around airfoils at fixed operating conditions, as described in Section 6.1.1. We use a publicly available dataset of NACA airfoil simulations [96], which provides a large and well-structured benchmark for evaluating feature extraction methods. For reproducibility, we release the reference implementation of the clustering-based feature extraction pipeline used for the *AirNACA* task at 10.5281/zenodo.15637850.

Surface Defect Detection (*AirDEF*) The *AirDEF* task extends the airfoil benchmark by introducing controlled surface deformations that mimic manufacturing defects, structural damage, or ice accretion. While the underlying airfoil geometries are still defined by NACA four-digit profiles, each configuration is augmented with synthetic surface defects applied to an otherwise nominal shape. The computational setup and mesh generation are described in Section 6.1.

To compactly encode surface modifications in a form suitable for learning, we adopt a three-digit defect code (i, j, k) derived from the five-digit representation introduced in Chapter 6. Specifically, the original five-digit code is mapped to a signed, lower-dimensional parametrisation by merging symmetric deformations and introducing an explicit sign to distinguish bumps from cavities. The first two digits $i, j \in [-2,2]$ encode surface deformations on the suction and pressure sides, respectively, where positive values correspond to bumps and negative values to cavities, and the absolute value denotes deformation intensity. The third digit $k \in [0,2]$ controls the depth of a trailing-edge cut. Table 5.2 reports the training defect codes together with representative geometries for a reference NACA0012 airfoil. This parametrisation allows a unified treatment of different defect types and intensities while reducing the dimensionality of the regression and classification problems. We generate 18 distinct defect configurations per airfoil, resulting in a total of 3600 CFD flow fields.

5-digit code	**3-digit code** (i, j, k)	**Interpretation**
$\begin{pmatrix} 1 & 0 & 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} +1, & 0 , & 0 \end{pmatrix}$	Suction-side bump (low intensity)
$\begin{pmatrix} 2 & 0 & 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} +2, & 0 , & 0 \end{pmatrix}$	Suction-side bump (high intensity)
$\begin{pmatrix} 0 & 1 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 , & +1, & 0 \end{pmatrix}$	Pressure-side bump (low intensity)
$\begin{pmatrix} 0 & 2 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 , & +2, & 0 \end{pmatrix}$	Pressure-side bump (high intensity)
$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 , & 0 , & 0 \end{pmatrix}$	Suction-side cavity (low intensity)
$\begin{pmatrix} 2 & 0 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -2 , & 0 , & 0 \end{pmatrix}$	Suction-side cavity (high intensity)
$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 , & -1 , & 0 \end{pmatrix}$	Pressure-side cavity (low intensity)
$\begin{pmatrix} 0 & 2 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 , & -2 , & 0 \end{pmatrix}$	Pressure-side cavity (high intensity)
$\begin{pmatrix} 1 & 1 & 1 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} +1, +1, & 0 \end{pmatrix}$	Bumps on both sides
$\begin{pmatrix} 2 & 2 & 1 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} +2, +2, & 0 \end{pmatrix}$	Strong bumps on both sides
$\begin{pmatrix} 1 & 1 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 , & -1 , & 0 \end{pmatrix}$	Cavities on both sides
$\begin{pmatrix} 2 & 2 & 0 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -2 , & -2 , & 0 \end{pmatrix}$	Strong cavities on both sides
$\begin{pmatrix} 1 & 1 & 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} +1, & -1 , & 0 \end{pmatrix}$	Suction bump, pressure cavity
$\begin{pmatrix} 2 & 2 & 1 & 0 & 0 \end{pmatrix}$	$\begin{pmatrix} +2, & -2 , & 0 \end{pmatrix}$	Strong asymmetric deformation
$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 , & +1, & 0 \end{pmatrix}$	Suction cavity, pressure bump
$\begin{pmatrix} 2 & 2 & 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} -2 , & +2, & 0 \end{pmatrix}$	Strong asymmetric deformation
$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 , & 0 , & 1 \end{pmatrix}$	Trailing-edge cut (low depth)
$\begin{pmatrix} 0 & 0 & 0 & 0 & 2 \end{pmatrix}$	$\begin{pmatrix} 0 , & 0 , & 2 \end{pmatrix}$	Trailing-edge cut (high depth)

Table 5.2: Mapping between the original five-digit deformation codes introduced in Chapter 6 and the three-digit encoding (i, j, k) adopted in the *AirDEF* task. The first two digits encode surface deformations on the suction and pressure sides, respectively, with sign distinguishing bumps (positive) from cavities (negative) and magnitude controlling deformation intensity. The third digit encodes the depth of a trailing-edge cut.

Pathology Identification in Upper Airways (*NosePAT* and *NoseREAL*) The third task, referred to as *NoseREAL*, focuses on pathology identification from CFD simulations of airflow in human upper airways. We consider the two clinically relevant conditions introduced in Chapter 3 and visibile

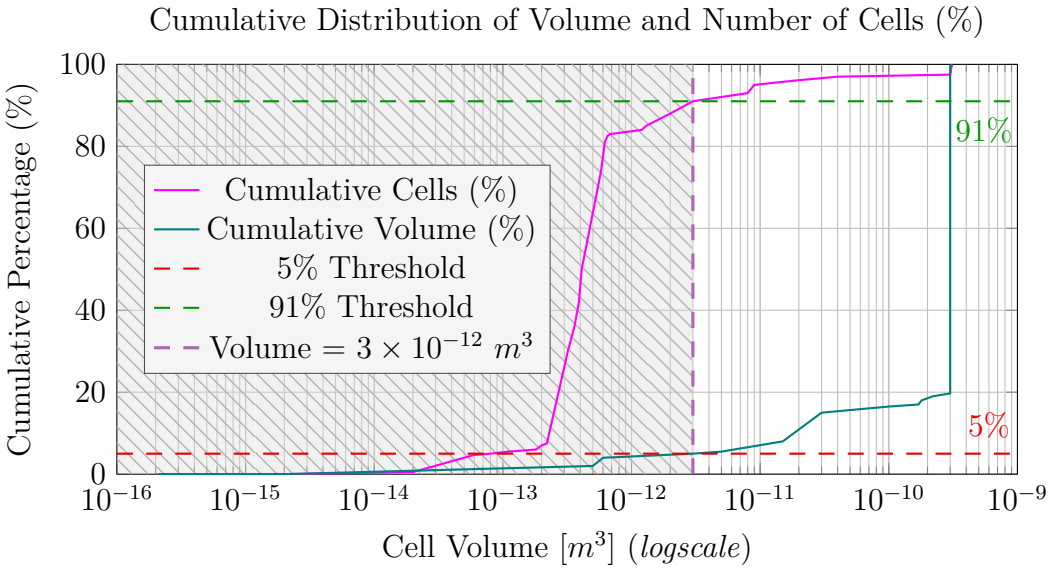


Figure 5.5: Cumulative percentage distribution of Cell Volume and Number of Cells with respect to the size of the cells. The dashed lines represent the percentage of volume we lose with the filtering (red line), the percentage of memory we save (light blue line), and the volume threshold (violet line).

in Figure 3.3: septal deviation and turbinate hypertrophy. The data consist of three-dimensional LES simulations performed on patient-specific geometries reconstructed from CT scans (see Section 6.2). Specifically, we consider the set of synthetic pathological geometries introduced in Chapter 4, generated through the data augmentation strategy described therein. This collection comprises 308 anatomically consistent upper-airway geometries with controlled septal deviations and turbinate hypertrophies, each associated with an LES simulation of steady inspiration. To assess generalisation to real clinical conditions, we further define a second evaluation task, referred to as *NoseREAL*, based on the set $\mathcal{D}_{\text{real}}$ introduced in Chapter 4. This set consists of ten CT-derived pathological anatomies (five septal deviations and five turbinate hypertrophies) diagnosed by medical experts and is used exclusively for testing. The *NoseREAL* task allows us to evaluate whether models trained on synthetic geometries can correctly identify pathologies in real, previously unseen clinical cases.

Given the extremely high resolution of the LES meshes (on the order of $\mathcal{O}(10^7)$ cells), a post-processing filtering step is applied to reduce the computational cost of feature extraction. Figure 5.5 reports the cumulative distributions of cell count and enclosed volume as a function of cell size. While the smallest cells account for the vast majority of the mesh elements, they occupy only a negligible fraction of the total domain volume and are primarily concentrated in

near-wall regions where mesh refinement is driven by numerical stability and turbulence modelling requirements rather than by the large-scale flow organisation. By discarding cells with volumes smaller than 3×10^{-12}, m^3, approximately 91% of the cells are removed while retaining about 95% of the total domain volume. Importantly, this filtering operation preserves the dominant flow structures associated with airway-scale geometry and obstruction patterns, which are the primary drivers of pathology-related flow alterations. As a result, the reduction substantially decreases data size and computational cost without affecting the physical information relevant to downstream clustering and classification tasks.

5.5.2 Baseline: Hand-Crafted Regions $\{R_{i,j}\}_{j=1}^{r_i}$

As a baseline for all experimental evaluations, we adopt the expert-driven feature extraction strategy introduced in Section 2.2, based on manually defined flow regions. Here, we specify the concrete definition of the handcrafted regions used in each experimental scenario, while referring to the previous chapters for the conceptual formulation of the baseline approach.

Aerodynamic cases (*AirNACA* and *AirDEF*). For the airfoil-related tasks, handcrafted regions are defined following the classical strategy proposed in [97]. Three vertical lines perpendicular to the airfoil chord are placed at $x = -c$, $x = c$, and $x = 10c$ (left-hand side of Figure 5.6). Each line is subdivided into eight regions symmetrically distributed around the midline $y = 0$, with boundaries defined by the coordinates

$$[-500,\, -10,\, -1,\, -0.1,\, 0,\, 0.1,\, 1,\, 10,\, 500]\, c.$$

On each region, we compute area-weighted averages of the velocity magnitude $|\boldsymbol{U}|$ and pressure p, as described in (4.5). This procedure yields a fixed and ordered set of handcrafted features that can be directly processed by standard ML models.

Biomedical cases (*NosePAT* and *NoseREAL*). For the upper-airway datasets, we adopt the same handcrafted regions defined in Chapter 6 and adopted in Chapter 4. Briefly, six cross-sectional planes are placed along the nasal cavity, spanning the olfactory region (right-hand side of Figure 5.6). Each section is split into left and right semi-sections, yielding a total of twelve handcrafted regions $\{R_1, \ldots, R_{12}\}$. On each region, we compute the average velocity magnitude $|\boldsymbol{U}|$, resulting in a 12-dimensional feature vector for each simulation.

As already discussed in Chapter 4, these regions are defined once on the healthy reference anatomies and reused for all corresponding synthetic variants. This handcrafted-region baseline therefore serves as a reference point, but would

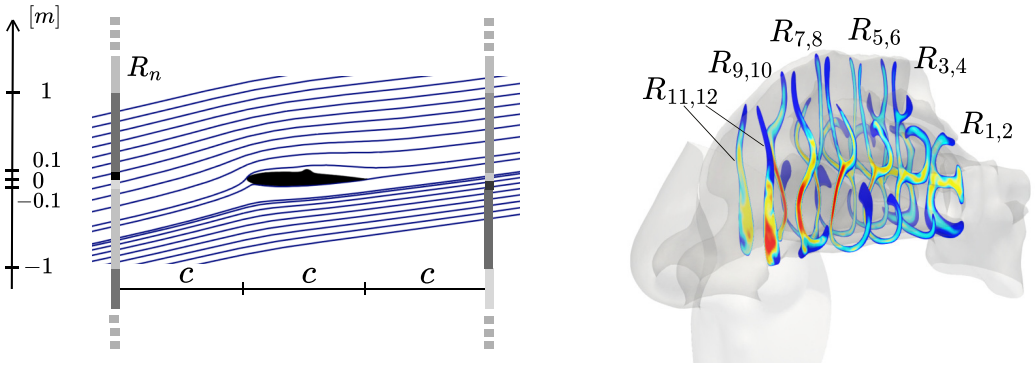


Figure 5.6: Handcrafted regions used in the baseline approach: aerodynamic cases (*AirNACA* and *AirDEF*, left) and biomedical cases (*NosePAT* and *NoseREAL*, right).

entail a substantial expert-driven annotation effort when applied independently to each patient-specific geometry, as in a realistic clinical setting.

5.5.3 Conducted Experiments

In this section, we describe the experimental settings adopted to evaluate the proposed feature extraction strategies. We consider six different approaches for constructing the feature set $\mathbf{P}_i$, which differ in the definition of the regions $\{R_{i,j}\}_{j=1}^{r_i}$, the extracted features, and the inference model K. The considered settings are summarised in Table 5.3.

I) *HC* (Hand-Crafted features). This setting serves as the baseline. Regions $\{R_{i,j}\}_{j=1}^{r_i}$ are manually defined by experts on each geometry M_i, as detailed in Section 5.5.2 and illustrated in Figure 5.6. Features are computed as regional averages of velocity magnitude and pressure. Since the resulting feature vectors have fixed size and consistent ordering across samples, inference is performed using a Multi-Layer Perceptron (MLP).

II) *CR+HC* (Clustering Regions with Hand-Crafted features). In this experiment, clustering is used to identify flow regions via the operator Φ, while retaining the same handcrafted features adopted in *HC*. This setting isolates the effect of region definition, allowing us to assess the impact of clustering without modifying the feature set. We employ the *C-PROP* strategy described in Section 5.4.1 to enforce consistent cluster correspondence across simulations. A reference configuration is selected for each dataset (a NACA0012 airfoil for *AirNACA* and *AirDEF*, and one healthy subject for *NosePAT* and *NoseREAL*).

Since the feature vectors are ordered and of fixed size, an MLP is used for inference.

III) *FREE-CR+FC* (C-FREE, Clustering Regions with Full Cluster features). Here, regions are obtained using the clustering-based method described in Section 5.4.1, and features are computed as regional averages of the quantities listed in Table 5.1. Clustering is performed independently on each simulation, so both the number and ordering of clusters may vary. The resulting unordered sets of feature vectors $\mathbf{P}_i$ are processed using a Point Transformer, which naturally handles variable-size and permutation-invariant inputs, as discussed in Section 5.4.4.

IV) *PROP-CR+FC* (C-PROP, Clustering Regions with Full Cluster features). This setting enforces consistency across simulations by propagating clusters from a reference case using the *C-PROP* strategy. The same references adopted in *CR+HC* are used. By fixing both the number and ordering of clusters, the feature vectors become directly comparable across samples, enabling the use of a standard MLP for inference.

V) *MORPH+HC* (Morphing with Hand-Crafted features). In this experiment, all simulations are first morphed onto a common reference geometry M^*, as described in Section 5.4.2. The same handcrafted regions and features adopted in *HC* are then computed on the reference mesh. Since regions are defined once on M^* and consistently reused across simulations, the resulting feature vectors have fixed size and ordering, and inference is performed using an MLP.

VI) *MORPH+FC* (Morphing with Full Cluster features). Finally, this setting combines morphing-based region alignment with the richer feature set used in *FREE-CR+FC* and *PROP-CR+FC*. After morphing all flow fields onto the reference mesh M^*, features are computed as regional averages of the quantities reported in Table 5.1 over the expert-defined regions on M^*. Since these regions are fixed and ordered, an MLP is adopted for inference.

Table 5.3 provides a concise overview of the six experimental settings, highlighting the interplay between region definition, feature extraction, and inference models.

5.5.4 Morphing onto the Reference Geometry M^*

In this section, we describe how the boundary correspondence π_i is constructed in the experimental evaluation. The general definition of the morphing operator Π_i is given in Section 5.4.2; here, we restrict the discussion to the practical

Experiment	**Regions** $\{\boldsymbol{R}_{i,j}\}_{j=1}^{r_i}$	**Model** $\boldsymbol{K}$
HC (Baseline)	Expert-defined (expert-defined regions and features described in Section 5.5.2)	MLP
CR+HC	Clustering-based (regions from clustering, same features as *HC*)	MLP
FREE-CR+FC	Clustering-based (regions from clustering, features in Table 5.1)	PT
PROP-CR+FC	Clustering-based (clusters propagated from a reference, features in Table 5.1)	MLP
MORPH+HC	Expert-defined (regions and features of *HC* transferred via morphing)	MLP
MORPH+FC	Expert-defined (regions transferred via morphing, features in Table 5.1)	MLP

Table 5.3: Summary of the conducted experiments. Each setting differs in how regions are defined, which features are extracted, and the inference model employed.

definition of π_i in the considered application scenarios. The correspondence $\pi_i : \partial\Omega_i \to \partial\Omega^*$ defines a point-to-point mapping between the boundary of the i-th geometry and that of the reference domain and provides the boundary displacements required by the morphing procedure.

Airfoil cases (*AirNACA* and *AirDEF*). For the aerodynamic datasets, boundary displacements are defined directly based on geometric and aerodynamic prior knowledge, following the approach of [16]. The boundary $\partial\Omega_i$ consists of the airfoil surface and the farfield boundary. Farfield nodes are fixed to enforce a zero-displacement condition, while leading-edge and trailing-edge points are matched explicitly to their counterparts on the reference geometry. Nodes along the suction and pressure sides are associated with the corresponding curves on the reference airfoil, preserving the relative node density along the surface. The RBF support radius r_s is set proportional to the airfoil chord length, ensuring that deformations remain localized in the vicinity of the airfoil while maintaining mesh quality in the outer region of the domain. A schematic illustration of the adopted boundary conditions is reported in Figure 5.7.

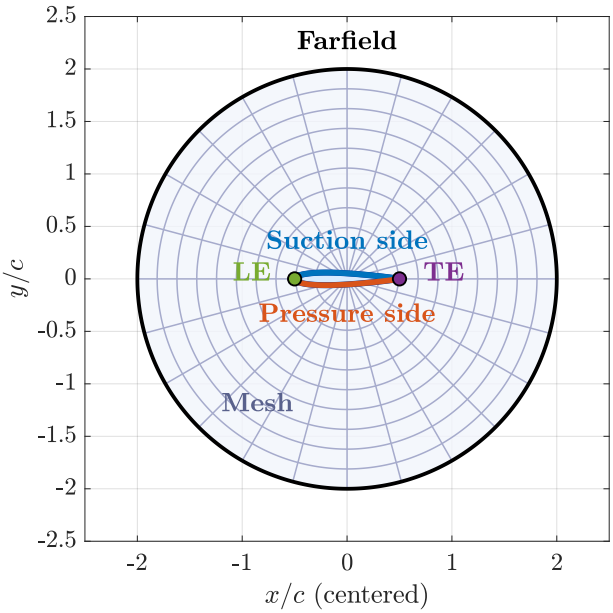


Figure 5.7: Schematic representation of a circular domain with a centred NACA0012 airfoil. The suction (upper) and pressure (lower) sides are highlighted in blue and red, respectively. Leading edge (LE) and trailing edge (TE) are marked with colored symbols, while the farfield boundary is indicated around the domain. Note that the illustrated domain is only a schematic representation: the computational domain is a much larger O-grid, ensuring sufficient distance from the boundaries to avoid spurious effects.

Upper-airway cases (*NosePAT* and *NoseREAL*). For the upper-airway datasets, boundary displacements are derived from anatomically consistent surface correspondences computed using the functional maps framework, as described in detail in Section 4.4.2. In this setting, the morphing operator Π_i leverages dense point-to-point correspondences $\mathcal{M}_i$ between the airway surfaces $\partial\Omega_i$ and the reference surface $\partial\Omega^*$ to prescribe boundary displacements $\mathbf{d}_i^b$ defined in Section 5.4.2. The resulting boundary displacements are propagated inside the volumetric mesh using RBFs, yielding a smooth deformation field that aligns the full three-dimensional geometry while preserving anatomical consistency. The support radius r_s is set proportional to the inter-nostril distance, which provides a natural global length scale for nasal geometries and allows the deformation to adapt smoothly to inter-patient variability.

5.5.5 Models Training and Evaluation

Across all experimental settings, we employ small inference models with a comparable number of trainable parameters, so that performance differences across experiments can be attributed to the feature extraction strategy rather than to variations in learning capacity. MLPs are used for the experiments *HC*, *CR+HC*, *PROP-CR+FC*, *MORPH+HC*, and *MORPH+FC*. For *FREE-CR+FC*, we adopt a PT, as this setting requires a permutation-invariant architecture capable of processing unordered feature sets. Architectural details and training hyperparameters are reported in Table 5.4.

Airfoil experiments (*AirNACA* and *AirDEF*). For both airfoil-related tasks, the datasets are standardised and randomly shuffled, then partitioned into five folds for cross-validation. At each iteration, one fold is used for testing, while the remaining folds are split into training (80%) and validation (20%) subsets. Since both tasks are formulated as regression problems, models are trained by minimising the mean squared error (MSE) loss.

Predicted values are not constrained to be integers and are therefore rounded to obtain the final discrete codes. Performance is reported as the average over the five folds and evaluated using the mean absolute error (MAE) computed separately for each digit of the code, together with the overall accuracy, defined as the fraction of samples for which the full code is correctly reconstructed after rounding.

Upper-airway experiments (*NosePAT*). For the *NosePAT* task, performance is evaluated using the LOPO-CV protocol, consistently with the evaluation strategy adopted in Section 4.5.3. Briefly, in each fold, all synthetic samples in $\mathcal{D}$ associated with one healthy reference anatomy are held out for testing, while the remaining data are split into training (85%) and validation (15%) subsets. Pathology identification is formulated as a binary classification problem, models are trained using categorical cross-entropy as the loss function, and classification accuracy is used as the primary evaluation metric.

Generalisation to real anatomies (*NoseREAL*). In the *NoseREAL* experiment, models are trained on the full set $\mathcal{D}$ of 308 synthetic pathological cases introduced in Chapter 4, using the same architectures and hyperparameters adopted in the previous experiments, and are evaluated on the set of real pathological anatomies $\mathcal{D}_{real}$. This setting simulates a realistic diagnostic scenario in which models trained on synthetic data are applied to real clinical cases. Given the limited number of available real samples, performance is summarized using the *modal score*, defined as the number of correctly classified patients most frequently obtained across multiple independent trainings of the same model.

Model	Experiments	Architecture	Training setup
MLP	*HC, CR+HC, PROP-CR+FC, MORPH+HC, MORPH+FC*	Flatten, Dense(256, 128, 64, 32, 16, 8, output dimension)	Batch size: 16 Optimizer: Adam + exponential decay (LR 0.001, decay 0.985, steps 150) Epochs: up to 1000 with early stopping (patience = 10)
PT	*FREE-CR+FC*	2×Attention(32, 64), Dense(128, 64, 32, 16, output dimension)	Batch size: 16 Optimizer: Adam (LR 0.0001) Epochs: up to 1000 with early stopping (patience = 10)

Table 5.4: Summary of the inference model architectures used for K. As in [97], we train simple MLPs for *HC*, *CR+HC*, *PROP-CR+FC*, *MORPH+HC*, and *MORPH+FC*, and adopt a Point Transformer for *FREE-CR+FC*. All models are defined with a comparable number of parameters to ensure similar learning capacity. The *output dimension* is left unspecified, as it depends on the target variable in each experimental scenario (*AirNACA*, *AirDEF*, or *NosePAT*).

This metric provides a compact and robust indication of typical model behavior in small-sample evaluation regimes.

5.5.6 Challenges in the 3D Extension and Computational Costs

Extending the proposed framework from two- to three-dimensional CFD simulations introduces several practical challenges, primarily related to computational cost, geometric correspondence, and mesh quality preservation. First, the computational burden increases substantially in three dimensions. While 2D airfoil meshes contain on the order of 10^5 cells, 3D upper-airway simulations typically involve tens of millions of cells before post-processing. In our implementation, the clustering stage was executed on the Leonardo Data-Centric General-Purpose (DCGP) partition of the CINECA HPC system, using one process per core across three compute nodes equipped with dual Intel Xeon Platinum 8480+ CPUs (56 cores per CPU). For 2D airfoil simulations (approximately 3×10^5 cells), BGMM-based clustering required about 10–12 minutes per sample. In contrast, for 3D nasal geometries, after the filtering step described in

Section 5.5, the meshes contained approximately 1.5–1.7 million cells, and clustering required around 1.5 hours per sample. This behavior is consistent with the expected linear scaling of the BGMM with respect to the number of data points. The morphing procedure was performed on the same HPC infrastructure. For the airfoil datasets, each morphing operation required approximately 30 minutes per case, including both the RBF-based deformation and the finite-element interpolation stages. For the 3D nasal geometries, where the number of mesh nodes and control points increases by roughly one order of magnitude, morphing required approximately 4–5 hours per sample.

Second, defining accurate point-to-point correspondences on complex anatomical 3D surfaces is inherently challenging, particularly in the presence of irregular boundaries or local geometric variability. To address this issue, we exploit the shared topological structure of upper-airway anatomies and employ functional-map-based techniques to recover dense and anatomically consistent correspondences from a limited set of manually identified landmarks. The computation of these correspondences is performed offline on a personal workstation following the procedure detailed in Appendix A, and requires approximately 15 minutes per geometry without relying on HPC resources.

Finally, preserving mesh quality during the morphing process is critical to ensure numerical stability and accuracy of the transferred CFD fields. Following [16], we employ compactly supported C^2 radial basis functions with an adaptive support radius (see Section 5.4.2), which allow smooth deformation fields while preventing excessive mesh distortion.

Despite the substantial increase in computational cost introduced by the three-dimensional extension, this overhead is acceptable given the level of anatomical realism and physical fidelity required by the target applications. The reported timings show that clustering and morphing remain tractable on current HPC infrastructures for patient-specific upper-airway geometries, enabling the analysis of clinically relevant cases that cannot be meaningfully addressed using two-dimensional or simplified models.

5.6 Results

The results of the experimental evaluation are summarised in Tables 5.5 and 5.6. Table 5.5 reports the test accuracy obtained by the different inference models across all considered tasks, while Table 5.6 provides the MAE and standard deviation for each digit of the regression codes in the airfoil experiments.

***AirNACA* and *AirDEF*.** The results on the airfoil datasets show that clustering-based region extraction is consistently beneficial when compared to the handcrafted baseline (*HC*). This trend is observed both when clustering is

	Test Accuracy			Score
	AirNACA	*AirDEF*	*NosePAT*	*NoseREAL*
HC (Baseline)	84.6%	65.6%	**88.8%**	**8/10**
CR+HC	85.0%	83.2%	71.5%	6/10
PROP-CR+FC	**86.5%**	**88.7%**	86.8%	**8/10**
FREE-CR+FC	<u>85.1%</u>	<u>84.2%</u>	77.5%	7/10
MORPH+HC	82.4%	64.1%	84.3%	7/10
MORPH+FC	82.5%	64.3%	<u>88.5%</u>	**8/10**

Table 5.5: Test accuracy across all tasks. On the right-hand side, we also report the modal score on the set of real pathological patients. Best results are highlighted in **bold**, while second-best results are <u>underlined</u>.

used to redefine regions while retaining handcrafted features (*CR+HC*), and when richer physics-based features are extracted from the clusters (*FREE-CR+FC* and *PROP-CR+FC*).

In *AirNACA*, *PROP-CR+FC* achieves the highest accuracy (86.5%), outperforming both the handcrafted baseline and the fully unordered clustering strategy (*FREE-CR+FC*). This indicates that enforcing a consistent number and ordering of clusters across simulations yields a more stable representation, which is particularly advantageous for regression tasks. Although *FREE-CR+FC* can adapt flexibly to each simulation and capture local flow variations, the absence of a fixed ordering introduces additional variability that slightly degrades performance. A similar but more pronounced trend is observed in *AirDEF*. The handcrafted baseline performs poorly (65.6%), while clustering-based approaches yield substantially higher accuracies, with *PROP-CR+FC* reaching 88.7%. This gap can be explained by the nature of surface defects: localised geometric perturbations such as bumps, cavities, and trailing-edge cuts induce flow modifications that are spatially confined near the airfoil surface. Handcrafted regions, being placed far from the airfoil, are largely insensitive to these effects, whereas physics-based clustering identifies regions directly associated with the governing equations, resulting in features that are more informative for defect detection.

The morphing-based experiments in the aerodynamic setting (*MORPH+HC* and *MORPH+FC*) yield performances close to the handcrafted baseline. This behaviour is expected, as the expert-defined regions are located far from the airfoil surface and are therefore minimally affected by the morphing procedure. These results primarily validate the morphing pipeline, confirming that the deformation process preserves the consistency of CFD fields across geometries without introducing artifacts that degrade performance.

Table 5.6 further supports these observations. In *AirNACA*, the second digit

of the NACA code, corresponding to the position of maximum camber, exhibits the highest MAE across all methods, indicating that it is the most challenging parameter to infer from flow data. Conversely, the thickness parameter (third digit) is predicted with the lowest error relative to its range, suggesting that thickness information is strongly encoded in the flow field. In *AirDEF*, the trailing-edge cut (third digit of the defect code) is the easiest parameter to predict, as it generates a distinctive wake structure that is consistently captured by clustering-based features. The first two digits, encoding bump or cavity intensity, remain more difficult due to their subtler aerodynamic signatures.

NosePAT. In the biomedical scenario, the observed trends partially differ. Incorporating the richer set of physics-based features improves classification performance for clustering-based methods, with *PROP-CR+FC* achieving 86.8% accuracy. However, the handcrafted baseline (*HC*) attains the highest performance (88.8%). This result reflects the strong inductive bias provided by expert-defined cross-sectional regions, which are explicitly designed to capture flow alterations associated with septal deviations and turbinate hypertrophies. Despite this, *PROP-CR+FC* achieves performance close to the handcrafted baseline without relying on any expert-driven definition of regions, demonstrating that clustering-based feature extraction can generalise effectively to anatomically complex domains.

Regarding the morphing-based experiments, *MORPH+HC* (84.3%) achieves performance comparable to the handcrafted baseline (*HC*), confirming that morphing can reliably transfer expert-defined regions onto a common reference geometry while preserving consistency across anatomies. *MORPH+FC* attains 88.5% accuracy, nearly matching the baseline and outperforming both clustering-based strategies. This result highlights the benefit of combining the spatial consistency provided by morphing with the richer physical descriptors extracted from clustering-based features: expert-defined regions ensure interpretability, while the additional physics-aware features enhance discriminative capability. Overall, the results indicate that the morphing-based strategy provides a tangible benefit in the biomedical setting. Across all configurations, it consistently achieves performance comparable to, or higher than, geometry-specific handcrafted features, while avoiding manual region definition for each anatomy. This demonstrates that morphing is an effective mechanism for retaining expert-level performance in the presence of substantial anatomical variability, which is a defining characteristic of realistic clinical applications.

NoseREAL. The evaluation on real pathological anatomies provides a more stringent validation of the proposed methods. Models trained exclusively on synthetic pathological cases are tested on previously unseen real patients, closely

	Mean Absolute Error (MAE) and Standard Deviation (σ)											
	AirNACA						*AirDEF*					
	I digit		II digit		III digit		I digit		II digit		III digit	
	MAE	$\pm\sigma$	MAE	$\pm\sigma$	MAE	$\pm\sigma$	MAE	$\pm\sigma$	MAE	$\pm\sigma$	MAE	$\pm\sigma$
Range	[0:9]		[0:9]		[05:50]		[-2:2]		[-2:2]		[0:2]	
HC (Baseline)	0.17	0.23	0.30	0.24	0.17	0.26	0.35	0.25	0.33	0.18	0.11	0.15
CR+HC	<u>0.16</u>	0.20	0.28	0.21	0.16	0.23	0.22	0.21	<u>0.21</u>	0.17	0.03	0.12
PROP-CR+FC	**0.14**	0.19	**0.24**	0.21	**0.15**	0.24	**0.18**	0.19	**0.19**	0.18	**0.01**	0.11
FREE-CR+FC	**0.14**	0.21	<u>0.26</u>	0.20	<u>0.16</u>	0.19	<u>0.21</u>	0.22	0.21	0.18	<u>0.02</u>	0.09
MORPH+HC	0.18	0.23	0.30	0.25	0.19	0.30	0.38	0.27	0.37	0.22	0.12	0.15
MORPH+FC	0.17	0.23	0.29	0.24	0.16	0.27	0.36	0.25	0.35	0.18	0.11	0.14

Table 5.6: Mean Absolute Error and standard deviation (σ) for each digit of the regression code in *AirNACA* and *AirDEF* tasks. Best results (lowest MAE) are highlighted in **bold**, while second-best results are <u>underlined</u>.

reflecting a realistic diagnostic setting. Despite the very limited number of available real samples, both *PROP-CR+FC* and *MORPH+FC* achieve a modal score of 8/10, indicating stable and consistent performance across patients. These results show that the proposed methods generalise effectively from synthetic to real CFD data, supporting their applicability even in small-sample clinical regimes.

5.7 Conclusions

In this Chapter, we addressed the problem of learning from high-dimensional CFD data in settings where each simulation produces millions of degrees of freedom, making direct use of raw flow fields impractical for ML. To overcome this limitation, we formulated feature extraction as the problem of defining physically meaningful and geometrically consistent regions within the flow domain, from which compact descriptors can be derived and used for inference.

Within this framework, we introduced and analysed two complementary but conceptually distinct strategies for region definition and feature extraction. The first strategy is a *Clustering-based* method, which automatically identifies flow regions by clustering quantities derived from the governing equations of the CFD model. By grouping cells that share a similar local balance of physical terms, this approach produces regions that reflect coherent flow phenomena without relying on any geometry-specific or expert-defined landmarks. This

method is based on the underlying physics of the problem, and it proved particularly effective in aerodynamic applications, where flow structures such as wakes, shear layers, and separation regions are strongly encoded in the governing equations. In the airfoil experiments, clustering-based feature extraction consistently outperformed the handcrafted baseline, achieving up to 86.5% accuracy in airfoil identification and 88.7% in defect detection. These results demonstrate that physics-based clustering provides informative and discriminative representations when the relevant flow structures are not tightly coupled to predefined geometric locations.

The second strategy is a *Morphing-based* method, which enforces spatial consistency across simulations by mapping heterogeneous geometries and their associated flow fields onto a common reference mesh through smooth Radial Basis Function deformations. In this setting, regions are defined once by an expert on the reference geometry and then automatically reused across all samples after morphing. This approach preserves interpretability and explicitly embeds domain knowledge into the feature extraction process, while eliminating the need for case-by-case manual region definition. In the biomedical experiments, where anatomical correspondence and clinical interpretability are essential, morphing-based feature extraction achieved performance comparable to expert-driven baselines, reaching 88.5% accuracy in pathology classification. These results confirm that morphing provides a reliable mechanism for transferring expert knowledge across anatomically variable geometries while maintaining consistency and scalability.

Overall, the experimental results show that the two proposed strategies address complementary needs in learning from CFD data. Clustering-based methods are preferable in settings where flow organisation is primarily governed by physical mechanisms and geometric variability is secondary, as in aerodynamic applications. Conversely, morphing-based methods are more appropriate when expert knowledge and anatomical consistency play a central role, as in medical CFD, where inter-subject variability would otherwise hinder learning. Together, these approaches substantially reduce reliance on handcrafted, case-specific feature definitions and establish a general and scalable framework for region-based feature extraction from high-dimensional CFD simulations.

Chapter 6

CFD Datasets Generation

This Chapter describes the datasets that supported the study presented in this thesis. The focus is on their construction, organisation, and content, including details on geometry processing, CFD simulations, and labelling. All datasets are derived from CFD simulations performed in OpenFOAM and correspond to the application scenarios introduced in Chapter 3. Here, the datasets are presented in a unified and self-contained manner, with the objective of ensuring reproducibility and supporting their publication and reuse through Zenodo open repositories.

6.1 Defected NACA Airfoils

This section reports the full mathematical formulation used to generate the airfoil geometries included in the dataset. The same definitions are adopted throughout the entire dataset generation pipeline and are recalled here to ensure completeness and reproducibility.

6.1.1 Geometry Parametrization

Baseline airfoil geometries are defined using the classical NACA four-digit formulation. Each airfoil is identified by a code `ABXX`, where:

- A denotes the maximum camber as a percentage of the chord,
- B denotes the chordwise location of maximum camber in tenths of the chord,
- XX denotes the maximum thickness as a percentage of the chord.

The chord length is normalized to $c = 1$ and the parameters are parsed as

$$a = \frac{A}{100}, \qquad b = \frac{B}{10}, \qquad t = \frac{XX}{100}. \tag{6.1}$$

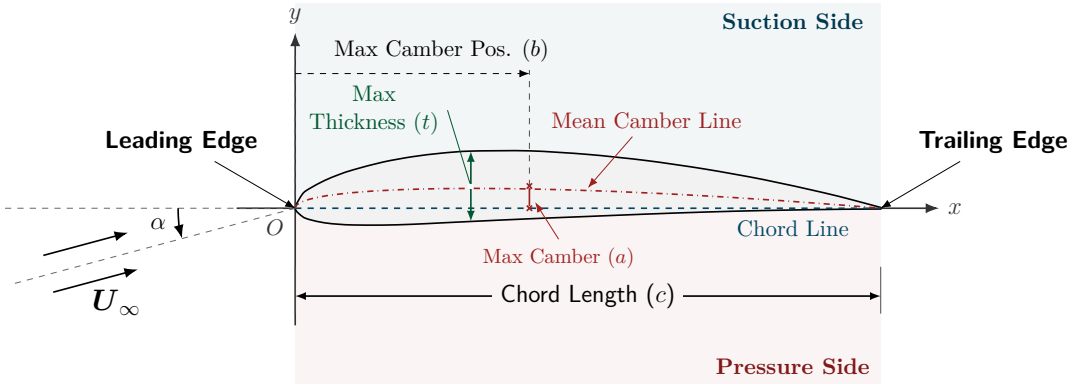


Figure 6.1: Schematic representation of a cambered NACA airfoil in the Cartesian coordinate system (x, y). The chord line (blue) and the mean camber line (red) are shown, partitioning the airfoil into the suction side (light blue) and the pressure side (light red). The geometry is parametrised by the chord length c and by the NACA digits a, b, and t. The incoming flow is characterised by the freestream velocity $\boldsymbol{U}_\infty$ and the angle of attack α.

Let $x \in [0,1]$ denote the non-dimensional chordwise coordinate. The thickness distribution $y_t(x)$ is defined as

$$y_t(x) = \frac{t}{0.2}\left(l_0\sqrt{x} + l_1 x + l_2 x^2 + l_3 x^3 + l_4 x^4\right), \tag{6.2}$$

with coefficients

$$(l_0, l_1, l_2, l_3) = (0.2969, -0.1260, -0.3516, 0.2843), \tag{6.3}$$

and

$$l_4 = \begin{cases} -0.1036, & \text{sharp trailing edge,} \\ -0.1015, & \text{finite-thickness trailing edge.} \end{cases} \tag{6.4}$$

Sampling points along the chord are distributed using half-cosine spacing, which increases resolution near the leading edge and improves numerical stability.

Mean Camber Line The mean camber line represents the locus of points that are equidistant from the upper and lower surfaces of the airfoil. It defines the geometric asymmetry of the profile with respect to the chord line and plays a central role in determining the aerodynamic characteristics of the airfoil, such as lift generation at zero angle of attack. Mathematically, it is defined as a function $y_c(x)$ of the chordwise coordinate $x \in [0,1]$, where the chord length is normalized to unity. Its shape is controlled by the parameters a and b, which represent the maximum camber and its chordwise location, respectively.

If $a = 0$, the airfoil is symmetric and the mean camber line is identically zero:

$$y_c(x) = 0. \tag{6.5}$$

Otherwise, the mean camber line $y_c(x)$ is defined piecewise as

$$y_c(x) = \begin{cases} \frac{a}{b^2}\left(2bx - x^2\right), & x \leq b, \\ \frac{a}{(1-b)^2}\left((1-2b) + 2bx - x^2\right), & x > b. \end{cases} \tag{6.6}$$

The local inclination angle $\theta(x)$ of the camber line is computed as

$$\theta(x) = \arctan\left(\frac{dy_c}{dx}\right), \tag{6.7}$$

with

$$\frac{dy_c}{dx} = \begin{cases} \frac{a}{b^2}(2b - 2x), & x \leq b, \\ \frac{a}{(1-b)^2}(2b - 2x), & x > b. \end{cases} \tag{6.8}$$

Upper and Lower Surfaces The coordinates of the baseline upper (U) and lower (L) surfaces are obtained as

$$x_{U/L}(x) = x \mp y_t(x) \sin\theta(x), \tag{6.9}$$

$$y_{U/L}(x) = y_c(x) \pm y_t(x) \cos\theta(x). \tag{6.10}$$

These equations define a smooth, closed airfoil profile parameterised along the chord.

Bumps and Cavities Localised surface defects are introduced by applying Gaussian perturbations to the upper and/or lower surfaces of the airfoil. Let x_U and x_L denote the chordwise coordinates of the upper and lower surfaces, respectively. The perturbation profile is defined by a Gaussian function

$$g(x) = \exp\left(-\frac{1}{2}\frac{(x - x_0)^2}{\sigma^2}\right), \tag{6.11}$$

where x_0 denotes the chordwise location of the defect centre. In all experiments, x_0 is fixed at mid-chord, i.e.

$$x_0 = 0.5, \tag{6.12}$$

while the width of the perturbation is controlled by

$$\sigma = 0.03. \tag{6.13}$$

The peak amplitude of the perturbation is set to

$$A = 0.02. \tag{6.14}$$

Defects can be applied independently to the upper and lower surfaces. A discrete severity selector $s \in \{0,1,2\}$ is mapped to the scaling factors

$$s \in \{0, 1, 1.5\}, \tag{6.15}$$

while a binary variable $\eta \in \{+1, -1\}$ determines whether the perturbation corresponds to a bump ($\eta = +1$) or a cavity ($\eta = -1$).

The perturbed surfaces are finally obtained as

$$y_U \leftarrow y_U + A\, s_U\, \eta_U\, g(x_U), \tag{6.16}$$

$$y_L \leftarrow y_L - A\, s_L\, \eta_L\, g(x_L). \tag{6.17}$$

The introduction of these perturbations results in either localised bumps or cavities on the airfoil surface, providing a controlled and parametric representation of surface defects for the experimental analysis.

Trailing-Edge Cut In addition to localised Gaussian surface perturbations, a further geometric modification is introduced to model non-ideal trailing-edge geometries. In practical applications, perfectly sharp trailing edges are often unfeasible due to manufacturing constraints, resulting in finite-thickness or truncated profiles.

Let D_{base} denote the original airfoil domain. The truncated domain is defined as

$$D_{\text{cut}} = \{(x, y) \in D_{\text{base}} \mid x \leq (1 - C_{\text{cut}})\}, \tag{6.18}$$

where C_{cut} denotes the fraction of chord removed from the trailing edge. The parameter C_{cut} is selected from a discrete set of predefined values, corresponding to incremental cut depths. Specifically, each admissible value of C_{cut} represents a 5% reduction of the chord length. This discretisation enables controlled and repeatable variations of the trailing-edge geometry while limiting the number of distinct configurations considered.

Defect Encoding Each airfoil geometry in the dataset is associated with a compact five-digit discrete encoding that uniquely identifies the presence, type, location, and severity of geometric defects, as well as the presence of a trailing-edge cut. This encoding is deterministic, directly derived from the geometry generation parameters, and is used as the ground-truth label for supervised learning tasks.

Let the defect code be denoted as

$$\mathcal{D} = d_1 d_2 d_3 d_4 d_5, \tag{6.19}$$

where each digit (or pair of digits) controls a specific geometric modification, as detailed below.

Digits d_1–d_2: Defect Severity on Upper and Lower Surfaces The first two digits $(d_1, d_2) \in \{0,1,2\}^2$ encode the presence and the severity of localised surface defects on the upper (suction) and lower (pressure) sides of the airfoil, respectively:

- d_1: severity level of the defect on the upper surface;
- d_2: severity level of the defect on the lower surface.

Each digit follows the same convention:

- $d_k = 0$: no defect on the corresponding surface;
- $d_k = 1$: defect with nominal severity;
- $d_k = 2$: defect with increased severity.

Digits d_3–d_4: Defect Type on Upper and Lower Surfaces The third and fourth digits $(d_3, d_4) \in \{0,1\}^2$ encode the type of surface defect applied to the upper and lower surfaces, respectively:

- d_3: defect type on the upper surface;
- d_4: defect type on the lower surface.

Each digit specifies whether the perturbation corresponds to a local thickening or thinning of the profile:

- $d_k = 1$: bump (positive displacement of the surface);
- $d_k = 0$: cavity (negative displacement of the surface).

These digits control the sign of the Gaussian perturbation, through the factors

$$\eta \in \{+1, -1\}, \tag{6.20}$$

with $\eta = +1$ corresponding to a bump and $\eta = -1$ to a cavity. The defect type is defined independently for the upper and lower surfaces.

Digit d_5: Trailing-Edge Cut The fifth digit $d_5 \in \{0,1,2\}$ encodes the presence and depth of a trailing-edge cut:

- $d_5 = 0$: no trailing-edge cut;
- $d_5 = 1$: trailing-edge cut with depth 5% of the chord;
- $d_5 = 2$: trailing-edge cut with depth 10% of the chord.

The cut is implemented by truncating both upper and lower surfaces at

$$x < 1 - 0.05\, d_5, \tag{6.21}$$

ensuring geometric consistency and preservation of the profile closure.

6.1.2 Selection of Baseline NACA Airfoils and Defect Configurations

To construct the set of baseline geometries used in the dataset, we generate a controlled subset of NACA four-digit airfoils rather than exhaustively sampling the full combinatorial space of parameters. This choice ensures physical consistency of the parametrisation, geometric diversity across samples, and numerical robustness of the resulting CFD simulations, while avoiding redundant or degenerate configurations.

Baseline airfoils are generated by enumerating the NACA four-digit parameters (a, b, t) within predefined discrete ranges. The admissible parameter ranges are defined as follows:

- $a \in \{0,1,\ldots,9\}$,
- $b \in \{0,1,\ldots,9\}$,
- $t \in \{05,06,\ldots,50\}$.

However, not all parameter combinations correspond to physically meaningful airfoils. The following selection rules are therefore enforced.

- **Purely symmetric profiles:** airfoils with zero camber ($a = 0$) are included only when the camber position parameter is also zero ($b = 0$), yielding symmetric profiles (e.g., NACA 0012). Configurations with $a = 0$ and $b \neq 0$ are discarded, as the camber position is undefined in the absence of camber.
- **Cambered profiles:** airfoils with nonzero camber ($a > 0$) are required to have a strictly positive camber position ($b > 0$). Configurations with $a > 0$ and $b = 0$ are excluded, as they correspond to degenerate geometries in which the camber line collapses at the leading edge, leading to ill-defined or numerically unstable profiles.
- **Thickness range:** the thickness parameter t spans a continuous set of admissible integer values between 5% and 50% of the chord. Extreme thickness values are intentionally excluded to avoid poor mesh quality and numerical instabilities in the CFD simulations.

After applying these physical and geometric constraints, the resulting pool of valid NACA airfoils contains several hundred distinct configurations. From this pool, a subset of 200 baseline airfoils is selected by uniform subsampling of the ordered list of admissible profiles. This strategy preserves coverage of the parameter space while preventing over-representation of specific camber or thickness combinations.

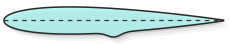
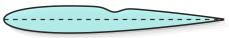
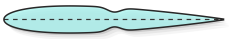
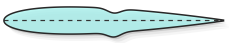
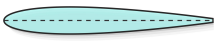
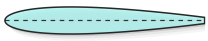
Code	Shape	Code	Shape
(1 0 1 0 0)		(2 0 1 0 0)	
(0 1 0 1 0)		(0 2 0 1 0)	
(1 0 0 0 0)		(2 0 0 0 0)	
(0 1 0 0 0)		(0 2 0 0 0)	
(1 1 1 1 0)		(2 2 1 1 0)	
(1 1 0 0 0)		(2 2 0 0 0)	
(1 1 1 0 0)		(2 2 1 0 0)	
(1 1 0 1 0)		(2 2 0 1 0)	
(0 0 0 0 1)		(0 0 0 0 2)	

Table 6.1: NACA deformation codes and corresponding shapes for the reference NACA0012 airfoil. All rows except the last represent surface modifications (bumps or cavities) with increasing intensity in the right column. The last row corresponds to trailing-edge cuts of increasing depth. The five-digit codes reported in the table match the image filenames and are the labels used in the dataset.

The resulting set of baseline airfoils provides a well-balanced and physically meaningful foundation for the dataset, upon which the surface defect configurations described in the following section are systematically applied.

Defect Configurations Although the five-digit encoding defined above allows for a large number of combinations, the dataset includes a controlled subset of 18 defect configurations. This subset was selected to cover representative deformation patterns (single-side defects, symmetric/asymmetric combinations, and trailing-edge cuts) while avoiding unnecessary combinatorial explosion.

Table 6.1 provides a visual gallery of the 18 configurations, illustrated on a reference NACA0012 airfoil. In the table, the *Code* column reports the five-digit defect identifier, which is also the label used in the dataset organisation.

6.1.3 CFD Simulation Setup

In this Section, we describe the numerical setup adopted to generate the CFD flow fields for the airfoil dataset, which is summarised in Table 6.2. All simulations are designed to produce a large number of airfoil geometries and defect configurations under identical numerical and physical conditions.

Computational Domain All simulations are performed in a two-dimensional external flow domain centred on the airfoil, with normalised chord length $c = 1$ and unitary fictitious spanwise dimension. The computational domain extends radially up to $500c$ from the airfoil, ensuring negligible influence of farfield boundaries on the near-body flow and wake development. This domain size follows standard practice for external aerodynamic simulations and is consistent with previous CFD datasets released for ML purposes [96].

Mesh Generation The computational grid is generated using the external mesh generation tool `construct2D`[1], which is specifically designed for two-dimensional aerodynamic applications. The tool employs a hyperbolic grid generation strategy to construct a structured O-Grid meshes around the airfoil geometry, ensuring smooth grid spacing and high-quality cell alignment in the near-wall region. The resulting two-dimensional mesh is subsequently extruded by a single cell in the spanwise direction to enable compatibility with the OpenFOAM solvers. Two-dimensionality is enforced by assigning the `empty` boundary condition to the spanwise patches for all solved variables.

Mesh refinement is concentrated in the vicinity of the airfoil surface to adequately resolve boundary-layer effects and capture local flow perturbations induced by surface defects such as bumps, cavities, and trailing-edge cuts. The mesh is progressively coarsened toward the farfield to reduce computational cost. All meshes are generated following the same strategy and with comparable resolution, resulting in an average cell count of $\mathcal{O}(10^6)$.

Physical Properties and Governing Equations The flow is assumed incompressible, turbulent, and characterised by constant physical properties across all simulations. The fluid density is set to $\rho = 1$ kg/m^3, while the kinematic viscosity is fixed to $\nu = 10^{-5}$ m^2/s, yielding Reynolds numbers representative of external aerodynamic flows at moderate freestream velocities.

The flow is modelled using the RANS formulation [83], where each instantaneous flow variable is decomposed into a mean component and a fluctuating one through temporal Reynolds averaging. For the velocity field, this decomposition

[1] `https://sourceforge.net/projects/construct2d/`

reads

$$\boldsymbol{U}(t) = \overline{\boldsymbol{U}} + \boldsymbol{u}'(t),$$

where the overbar $\overline{(\cdot)}$ denotes a time-averaging operator defined over a sufficiently long time interval, and $\boldsymbol{u}'(t)$ represents the zero-mean turbulent fluctuation, satisfying

$$\overline{\boldsymbol{u}'(t)} = \boldsymbol{0}.$$

An analogous decomposition applies to the pressure field.

Under the assumption of statistically steady conditions, the governing equations for the mean flow reduce to the steady RANS momentum equations

$$\nabla \cdot \left(\overline{\boldsymbol{U}} \otimes \overline{\boldsymbol{U}}\right) = -\frac{1}{\rho}\nabla \overline{p} + \nu \nabla^2 \overline{\boldsymbol{U}} - \nabla \cdot \overline{\boldsymbol{u}' \otimes \boldsymbol{u}'}, \tag{6.22}$$

where $\overline{\boldsymbol{U}}$ denotes the mean velocity, $\overline{p}$ the mean pressure, and ρ and ν the constant fluid density and kinematic viscosity, respectively.

The Reynolds stress tensor $\overline{\boldsymbol{u}' \otimes \boldsymbol{u}'}$ is closed by invoking the Boussinesq hypothesis [98] and introducing a turbulent eddy viscosity. Turbulence closure is provided by the one-equation Spalart–Allmaras model [100], activated in RAS mode in OpenFOAM. The model introduces a transported working variable $\tilde{\nu}$, from which the turbulent viscosity ν_t is derived. With this closure, the RANS momentum equations can be written as

$$\nabla \cdot \left(\overline{\boldsymbol{U}} \otimes \overline{\boldsymbol{U}}\right) = -\frac{1}{\rho}\nabla \overline{p} + \nu \nabla^2 \overline{\boldsymbol{U}} + \nabla \cdot \left(\nu_\mathrm{t} \nabla \overline{\boldsymbol{U}}\right) - \nabla \left(\tfrac{2}{3}\overline{k}_t\right), \tag{6.23}$$

where the turbulent kinetic energy is defined as

$$\overline{k}_t = \tfrac{1}{2}\overline{\boldsymbol{u}' \cdot \boldsymbol{u}'}.$$

The Spalart–Allmaras model provides a robust compromise between physical fidelity and computational efficiency, making it well-suited for the large number of simulations required for dataset construction.

Boundary and Initial Conditions At the farfield boundary, a uniform freestream velocity is imposed using freestream-type boundary conditions. The freestream magnitude is set to $|\boldsymbol{U}_\infty| = 30\,\mathrm{m/s}$ and the angle of attack is fixed to $\alpha = 10^\circ$ by prescribing the corresponding velocity components. The flow is characterised by a Reynolds number $\mathrm{Re} = 3 \times 10^6$. Pressure is assigned a freestream reference value, and turbulence quantities are initialised consistently with the inflow conditions. On the airfoil surface, no-slip boundary conditions are enforced for the velocity. Pressure is assigned a zero-gradient condition, while turbulence quantities are treated using standard wall treatments compatible with the Spalart–Allmaras formulation.

Temporal Discretisation and Time Averaging Although the quantities of interest correspond to steady flow statistics, the RANS equations are solved using a time-marching numerical scheme. In this context, the time variable represents a pseudo-time employed to drive the solution toward convergence and does not correspond to physical flow unsteadiness. Time integration is performed using a first-order implicit scheme with a constant time step $\Delta t = 1$ in solver time units. Each simulation is advanced up to $t = 2000$, corresponding to several thousand pseudo-time steps, until convergence of residuals and stabilisation of global flow quantities are observed, indicating that a steady RANS solution has been reached.

To further reduce residual numerical fluctuations, time-averaged fields are computed using the OpenFOAM `fieldAverage` function object. Averaging is activated after the initial transient (from $t = 200$), and mean values are accumulated over the remaining pseudo-time iterations. Only the averaged fields are retained and stored in the dataset, ensuring that the resulting flow fields consistently represent converged steady RANS solutions.

Numerical Discretisation and Solver Configuration The incompressible RANS equations are solved using the steady-state solver `simpleFoam`. A first-order steady-state temporal discretisation is employed, consistently with the fixed-point nature of the SIMPLE algorithm used for pressure–velocity coupling. Spatial discretisation relies on second-order accurate schemes wherever possible. Gradients are computed using a Gauss linear formulation, with an explicit cell-limited reconstruction applied to the velocity field in order to improve robustness in regions characterised by strong gradients. Convective terms are discretised using bounded upwind-biased schemes, ensuring numerical stability while limiting the onset of non-physical oscillations. Diffusive terms are treated using a corrected Laplacian formulation to properly account for mesh non-orthogonality.

Computational Resources The generation of the airfoil CFD dataset required large-scale computational resources due to the high number of simulations performed under a uniform numerical setup. All simulations were executed on the *Leonardo* supercomputer at *CINECA* (Bologna, Italy), within the Data Centric General Purpose (DCGP) partition.

Leonardo is a high-performance computing system equipped with compute nodes based on *Intel Xeon Platinum 8480+* processors. Each compute node features two CPUs, for a total of 112 physical cores per node, and is designed to support highly parallel workloads with large memory bandwidth. This architecture enables the efficient execution of large ensembles of CFD simulations.

Parameter	**Specification**
Flow configuration	External aerodynamics, 2D
Computational domain	Radial extent $500c$, normalized chord $c = 1$
Mesh	O-type topology `construct2D` , $\mathcal{O}(10^6)$ cells, near-wall refinement
Governing equations	Incompressible RANS
Turbulence model	Spalart–Allmaras (SA)
Solved variables	Velocity $\mathbf{u}$, pressure p, SA variable ν_t
Fluid properties	$\rho = 1$ kg/m^3, $\nu = 10^{-5}$ m^2/s
Boundary conditions	Farfield freestream, no-slip airfoil wall
Freestream conditions	$\lvert \boldsymbol{U}_\infty \rvert = 30$ m/s, $\alpha = 10°$, Re $= 3 \times 10^6$
Temporal treatment	Unsteady simulation, $\Delta t = 1$
Time averaging	Statistics collected for $t \geq 200$ using `fieldAverage`
Solver	`simpleFoam` (unsteady SIMPLE-based formulation)
Spatial discretisation	Second-order accurate, bounded schemes
Linear solvers	GAMG (pressure), smooth solvers with relaxation
HPC system	Leonardo (CINECA), DCGP partition
Hardware	Intel Xeon Platinum 8480+ CPUs
Parallel execution	2 CPU cores per simulation

Table 6.2: Summary of the CFD simulation setup for the defected airfoil dataset.

Each CFD simulation was executed using two CPU cores, exploiting distributed-memory parallelism through MPI. Simulations were dispatched through job-array scheduling, allowing a large number of independent CFD cases to be run concurrently. At runtime, simulations were distributed across two compute nodes, enabling high throughput while respecting per-job resource constraints and queue policies.

6.1.4 Data Post-processing

With the term *post-processing*, we refer to all operations performed on the CFD results after the simulations and prior to their use as input for the learning model K. These steps are therefore not related to the numerical solution procedure, but are introduced to prepare CFD data for learning-based inference.

For each simulation case, the final time-averaged flow field and the corresponding wall geometry are extracted through an automated post-processing pipeline and exported in a standard VTK-based format. This stage aims to isolate the flow information relevant for the learning task and to ensure consistency across the dataset, while preserving the physical meaning of the underlying CFD quantities.

Gradients computation As part of the preprocessing stage preceding the ML model K, differential quantities are derived from the CFD solution fields. The pressure gradient ∇p is computed directly within the OpenFOAM post-processing framework, ensuring full consistency with the spatial discretisation schemes and boundary conditions adopted in the numerical simulations. This quantity is computed once and subsequently treated as an additional flow field throughout the preprocessing pipeline.

The velocity field is considered through its in-plane Cartesian components U_x and U_y. Spatial gradients of the velocity components are instead computed in a separate processing stage using VTK-based operators. First-order spatial derivatives of U_x and U_y are evaluated, and the same gradient operator is applied iteratively to obtain higher-order velocity derivative fields.

Slice extraction and cell area computation Although the CFD simulations are generated on three-dimensional meshes, the computational domain is strictly two-dimensional, with a single cell in the spanwise direction. A two-dimensional representation of the flow is therefore obtained by slicing the volumetric dataset with a plane orthogonal to the spanwise axis. The slicing plane is defined by a normal vector (0,0,1) and a fixed origin at $z = z_0$. The resulting polygonal dataset is subsequently cleaned to remove duplicated points introduced by the cutting operation.

After slice extraction, the area associated with each cell of the 2D mesh is computed. Since the dataset consists of planar polygonal cells, the cell area is evaluated geometrically and stored as an additional scalar field. This quantity provides a meaningful weighting factor which allows learning algorithms to account for non-uniform mesh resolution.

Domain clipping and data reduction To limit the dimensionality of the CFD data while retaining the flow regions most relevant for learning-based

Field	Description
(x, y)	Cell centroid coordinates
A	Cell area
p	Pressure
(U_x, U_y)	Velocity components
ν_t	Eddy viscosity
∇p	Pressure gradient
$\nabla \boldsymbol{U}$	Velocity gradient tensor
$\nabla^2 \boldsymbol{U}$	Second-order velocity derivatives (Laplacian)
Format	VTK XML PolyData (`.vtp`, compressed)

Table 6.3: Per-cell quantities retained after postprocessing and used as input for the machine learning model K.

inference, the sliced flow field is spatially clipped to a fixed region surrounding the airfoil and its near-wake. The clipping operation retains only points lying within prescribed bounds in the streamwise and transverse directions,

$$x \in [-2,\ 7], \qquad y \in [-1.5,\ 1.5],$$

while leaving the spanwise direction unconstrained. This choice standardises the spatial extent of the data across all simulations and reduces the overall data volume without altering the physical content of the retained flow region. Despite this spatial restriction, the resulting flow fields remain large-scale, with the number of retained cells still being on the order of 5×10^5 per simulation. The clipping step therefore provides only a partial reduction in dimensionality and does not trivially simplify the learning problem.

Following the clipping operation, the retained flow data are further reduced by discarding auxiliary solver variables and intermediate quantities that are not used in the learning pipeline. Only a selected subset of physically meaningful fields is preserved, balancing physical expressiveness and data compactness, and ensuring consistency across the dataset.

At the level of individual mesh cells, the retained data include geometric information, primary flow variables, turbulence-related quantities, and gradient-based features derived from the CFD solution. Table 6.3 summarises the quantities retained at the per-cell level and their associated physical meaning.

6.2 Human Upper Airways from CT Scans

The second scenario considered in this thesis consists of patient-specific geometries of the human upper airways (see Section 3.2), reconstructed from CT scans and employed for LES of nasal airflow. Unlike the aeronautical dataset introduced in the previous section, this dataset exhibits strong anatomical variability, limited sample availability, and the absence of explicit parametric shape descriptions. In the following, we describe the extraction of airway geometries from CT scans, the preprocessing steps required to obtain CFD-ready domains, and the numerical setup adopted for the CFD simulations.

6.2.1 Extraction of the Surface $\tilde{S}_i$ from CT Scans

An essential step is the extraction of the patient-specific airway surfaces $\{\tilde{S}_i\}_{i=1,\dots,N}$ from a set of CT scans $\{T_i\}_{i=1,\dots,N}$ acquired from healthy subjects. The dataset of CT scans is derived from 7 scans of healthy patients provided by *ASST Santi Paolo e Carlo* in Milan (Italy), a medical institution involved in a long-standing research collaboration.

Each CT scan T_i results from a diagnostic imaging technique based on X-rays and reconstruction algorithms, and provides information on the internal structure of the body in the form of a three-dimensional volumetric (see Figure 3.2). The CT data consist of a volume discretised into a grid of voxels, each associated with a CT number, expressed in Hounsfield Units (HU), which is proportional to the local tissue density. Thanks to this representation, voxels corresponding to biological tissues or bones can be distinguished from air-filled regions, enabling the extraction of the internal airways. To obtain the surface $\tilde{S}_i$, representing the geometry of the upper airways, the CT scan T_i is segmented through the following procedure. First, a threshold is applied to the CT number in order to discriminate between high-density voxels, associated with biological tissues, and low-density voxels, corresponding to the internal air-filled cavities. The threshold value is fixed for all CT scans to -220 [HU] in accordance with the guidelines proposed by Quadrio et al. [84], which have been shown to effectively separate air and biological material in nasal airflow studies. Second, the interface between high-density and low-density voxels is extracted and converted into a triangulated surface representation. The resulting surface is described by the vertices of the triangular mesh and the corresponding face normals, embedded in a three-dimensional Cartesian coordinate system, and is stored in STL format for further processing and CFD simulations.

The segmentation process can be challenging, particularly in localised regions where nasal mucus accumulation or partial volume effects increase the apparent voxel density. These effects may introduce small surface irregularities, spurious connections, or local discontinuities, which are not acceptable for CFD

simulations. As a result, the extracted surfaces must be carefully inspected and, when necessary, manually corrected by an expert to ensure anatomical and geometric consistency. This manual intervention is inherently time-consuming; however, in the present setting, it is required only for a limited number of cases. Specifically, surface extraction and correction are performed exclusively on the 7 healthy patients that constitute the basis of the dataset: as discussed in Chapter 4 in fact, all additional geometries are generated from these cleaned surfaces through controlled geometric transformations, and therefore do not require further segmentation or manual refinement. Out of the 308 geometries considered in total, only the initial healthy samples require direct interaction with the CT data. This significantly reduces the number of surface extraction and meshing operations, which are among the most time-consuming steps in the overall pipeline, and represents a key strength of the adopted data generation strategy.

6.2.2 Data Augmentation based on Computational Geometry

As detailed in Chapter 4, the medical dataset used in this work is constructed starting from a limited set of 7 CT scans of healthy patients. From these reference cases, a total of 308 anatomically realistic airway geometries is generated through a data augmentation strategy based on geometric correspondence.

The procedure relies on a reference surface S^{ref} and on the establishment of dense correspondences between S^{ref} and the CT-derived airway surfaces $\tilde{S}_i$. These correspondences enable both the automatic cleaning of the raw geometries, by excluding anatomical regions not relevant for nasal airflow, and the transfer of expert-defined geometric deformations from the reference surface to patient-specific anatomies. Examples of the resulting cleaning and correspondence masks are reported in Figure 4.4.

As for the manual segmentation corrections discussed in the previous section, all operations that require direct interaction with the CT-derived surfaces are performed exclusively on the 7 healthy subjects that form the basis of the dataset. Once these reference geometries are processed, all additional samples are generated through controlled geometric transformations and do not require further interaction with the original CT data. As a result, the full set of geometries $\{S^{\star}_{i,j}\}_{i=1,\dots,N;\, j=1,\dots,V}$ is obtained. The complete formulation of the shape correspondence problem, the deformation transfer procedure, and the associated refinement strategies are described in detail in Chapter 4 and Appendix A.

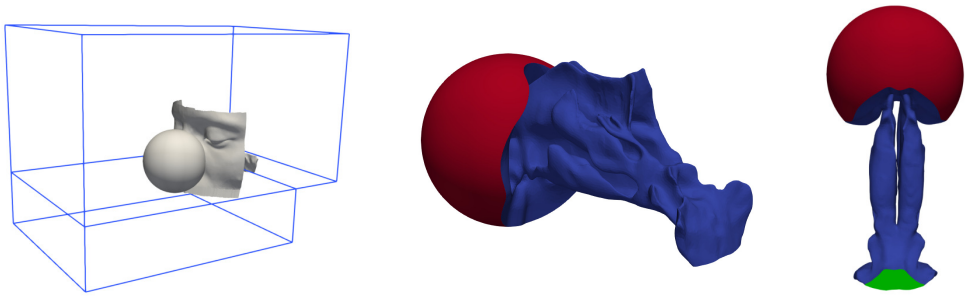


Figure 6.2: Visualisation of the computational domain and setup. On the left-hand side, the sphere closing the nostrils is visible, together with the bounding box of the simulation domain. In the centre and on the right-hand side, boundary regions are highlighted: the spherical inlet (red), the airway walls $S^{\star}_{i,j}$ (blue), and the outlet at the throat (green).

6.2.3 CFD Simulation Setup

This section describes the numerical setup adopted to generate the CFD flow fields of the human upper-airway dataset summarised in Table 6.4. The simulations are designed to produce physically consistent and mutually comparable flow fields across patient-specific anatomies and synthetically deformed geometries, under identical numerical and physical conditions. Although the quantities of interest employed in the subsequent experiments correspond to steady statistics, all simulations are performed in an unsteady framework and flow quantities are extracted through temporal averaging after the initial transient.

Computational Domain The computational domain consists of the full three-dimensional internal volume of the upper airways, as illustrated in Fig. 6.2. To mimic an open ambient environment at the nostrils, the domain is externally bounded by a spherical surface enclosing the inlet region, visible on the left-hand side of Fig. 6.2. Together with the airway surface $S^{\star}_{i,j}$, this spherical boundary defines a closed computational domain for the internal airflow while avoiding artificial confinement effects at the inlet.

The different boundary regions composing the domain are highlighted in Fig. 6.2: the spherical inlet surface, the airway walls corresponding to $S^{\star}_{i,j}$, and the outlet section located at the throat. This configuration allows the internal flow to develop under realistic inflow conditions while maintaining a well-posed computational setup.

Mesh Generation The CFD simulations are performed on unstructured three-dimensional meshes conforming to the patient-specific airway geometries

$S^{\star}_{i,j}$. Mesh generation is carried out in OpenFOAM using a two-stage procedure based on `blockMesh` and `snappyHexMesh`. In the first stage, `blockMesh` is used to generate a structured hexahedral background grid composed of multiple blocks, providing a uniform initial discretisation of the computational domain. The background mesh is isotropic, with unit grading in all directions, and serves exclusively as a support for subsequent refinement.

In the second stage, `snappyHexMesh` is employed to adapt the mesh to the airway geometry. The castellated refinement and snapping steps are enabled, while boundary-layer cell extrusion is deliberately disabled (`addLayers = false`). Surface-based refinement is applied uniformly along the airway walls with a fixed refinement level of 3, while a lower refinement level of 1 is assigned to the spherical inlet surface. Transitions between refinement levels are smoothed using `nCellsBetweenLevels = 2`.

Additional local refinement is triggered by geometric features through a curvature-based criterion with `resolveFeatureAngle = 80`. The internal flow region is selected using a point-based criterion (`locationInMesh`), ensuring that only the volume enclosed by the airway surface and the inlet boundary is retained.

Mesh quality is controlled by enforcing bounds on non-orthogonality and skewness, with a maximum non-orthogonality of 60. All meshes are generated using identical parameters across the dataset, resulting in comparable resolution and on the order of $\mathcal{O}(10^7)$ computational cells per simulation.

Physical Properties and Governing Equations The flow is assumed incompressible and turbulent, and is characterised by constant physical properties across all simulations. The fluid density is set to $\rho = 1$ kg/m^3, while the kinematic viscosity is fixed to $\nu = 10^{-5}$ m^2/s.

We adopt the LES formulation, in which the large-scale turbulent structures are explicitly resolved, while the effects of the smaller, unresolved scales are modelled [83]. Within this framework, a spatial filtering operation, denoted by the tilde $\widetilde{(\cdot)}$, is applied to the Navier–Stokes equations in order to separate the resolved scales from the subgrid-scale (SGS) fluctuations. The momentum equations read

$$\frac{\partial \widetilde{\boldsymbol{U}}}{\partial t} + \nabla \cdot \left(\widetilde{\boldsymbol{U}} \otimes \widetilde{\boldsymbol{U}} \right) = -\frac{1}{\rho} \nabla \widetilde{p} + \nu \nabla^2 \widetilde{\boldsymbol{U}} - \nabla \cdot \boldsymbol{\tau}^{\mathrm{SGS}}, \tag{6.24}$$

where $\widetilde{\boldsymbol{U}}$ and $\widetilde{p}$ denote the filtered velocity and pressure fields, respectively. The subgrid-scale stress tensor

$$\boldsymbol{\tau}^{\mathrm{SGS}} = \widetilde{\boldsymbol{U}\boldsymbol{U}} - \widetilde{\boldsymbol{U}} \otimes \widetilde{\boldsymbol{U}}$$

accounts for the effect of unresolved turbulent scales. Subgrid-scale closure is provided by introducing an eddy viscosity ν_{SGS}. The modelling of this term is

provided by the Wall-Adapting Local Eddy-viscosity (WALE) model [27], which provides the correct near-wall scaling behaviour without requiring additional damping functions.

Boundary and Initial Conditions At the spherical inlet surrounding the nostrils (in red in Figure 6.2), an inspiratory volumetric flow rate of 16 l/min is imposed through a flow-rate-based velocity condition. This corresponds to steady breathing at rest and is kept fixed across all simulations. On the airway walls $S^{\star}_{i,j}$ (in blue in Figure 6.2, no-slip and no-penetration conditions are enforced for the velocity field. The pressure is assigned a zero normal gradient at the walls, while the eddy viscosity is set to zero, consistently with a wall-bounded LES formulation. At the outlet located at the throat (green in Figure 6.2), the velocity field is assigned a zero normal gradient condition. The pressure is prescribed through a reference total pressure condition with zero gauge value. Initial and boundary conditions are kept identical for all simulations to ensure consistency and comparability of the resulting flow fields across the dataset.

Temporal Discretisation and Time Averaging Although LES is inherently unsteady, only time-averaged flow quantities are considered in the present work. Instantaneous fluctuations are therefore discarded, and the extracted features represent statistically stationary conditions after the decay of initial transients.

Each simulation reproduces an inspiration phase of duration $T_{\text{sim}} = 0.65\,\text{s}$, corresponding to steady breathing at rest. Time integration is performed using a nominal time step $\Delta t = 5 \times 10^{-5}\,\text{s}$, and the time step is dynamically adjusted during the simulation to enforce a maximum Courant number of $\text{Co}_{\text{max}} = 1$. After an initial transient phase, velocity and pressure fields are time-averaged starting from $t = 0.05\,\text{s}$ until the end of the simulation. Only these time-averaged fields are retained in the dataset and used in the subsequent stages of the machine learning pipeline.

Numerical Discretisation and Solver Configuration The incompressible LES equations are solved using the transient solver `pimpleFoam`. Time advancement is performed in a fully unsteady framework, consistently with the LES formulation, by combining pressure–velocity coupling and iterative corrections within the PIMPLE algorithm.

Temporal discretisation is second-order accurate and is based on a backward differencing scheme, while spatial discretisation relies on second-order accurate schemes throughout the computational domain. Gradients are computed using a Gauss linear formulation, and convective terms are discretised using central

differencing schemes. Diffusive terms are treated with a corrected Laplacian formulation to properly account for mesh non-orthogonality.

Pressure–velocity coupling is handled through a PIMPLE loop with a single outer iteration and two inner correctors. Linear systems arising from the discretisation are solved using iterative methods. Pressure is solved using a geometric–algebraic multigrid approach, while the velocity field is handled using a smooth iterative solver. Convergence within each time step is controlled through a combination of absolute and relative residual tolerances.

Computational Resources Similarly to the airfoil CFD dataset, the generation of the airway LES dataset required large-scale computational resources due to the high computational cost of each simulation and the uniform numerical setup adopted across all cases. All simulations were executed on the *Galileo* supercomputing system at *CINECA*. Galileo is a high-performance computing platform composed of 528 computing nodes equipped with dual *Intel Cascade Lake 8260* CPUs (24 cores per CPU at 2.4 GHz), for a total of 48 physical cores per node and 384 GB of RAM.

Each LES simulation was executed using 96 CPU cores and required approximately 160 GB of RAM. Simulations were launched in parallel across the system, allowing multiple cases to be processed concurrently while respecting per-job resource constraints and queue policies. Each run involved on the order of $\mathcal{O}(10^7)$ computational cells and produced approximately 40 GB of output data, including time-averaged flow fields and statistical quantities.

6.2.4 Data Post-processing

Similarly to the airfoil case, a postprocessing stage is introduced to transform the raw CFD output into a representation suitable for the learning-based analysis described in this thesis. Starting from the three-dimensional LES solution fields, a set of physically meaningful derived quantities is computed and retained for further processing. Quantities that depend directly on spatial derivatives of the resolved velocity field, such as resolved enstrophy E, the pressure gradient ∇p and related turbulence indicators, are computed during post-processing using the `foamPostProcess` utility, in order to ensure full consistency with the adopted numerical discretisation. In addition, spatial gradients of the resolved velocity field are computed following the same strategy adopted for the airfoil dataset and treated as derived quantities in the preprocessing pipeline.

Turbulent kinetic energy k_t is instead evaluated *a posteriori* from the resolved velocity fluctuations. In particular, the time-averaged velocity field is subtracted from instantaneous snapshots, and fluctuation statistics are accumulated over the averaging window to obtain the resolved contribution to the turbulent kinetic energy.

Parameter	Specification
Flow configuration	Internal nasal airflow, 3D
Computational domain	Full upper-airway volume with spherical inlet boundary
Mesh	Unstructured 3D mesh, $\mathcal{O}(10^7)$ cells
Governing equations	Incompressible LES
Turbulence model	WALE (subgrid-scale eddy-viscosity)
Solved variables	Filtered velocity $\widetilde{\boldsymbol{U}}$, filtered pressure $\widetilde{p}$
Fluid properties	$\rho = 1$ kg/m^3, $\nu = 10^{-5}$ m^2/s
Boundary conditions	Spherical inlet, no-slip airway walls, throat outlet
Inlet condition	Volumetric flow rate 16 l/min
Temporal treatment	Unsteady simulation, $\Delta t = 5 \times 10^{-5}$ s
Time averaging	Statistics collected after initial transient ($t \geq 0.05$ s)
Solver	`pimpleFoam` (PIMPLE-based formulation)
Spatial discretisation	Second-order accurate central schemes
Linear solvers	GAMG (pressure), smooth iterative solvers (velocity)
HPC system	Galileo (CINECA)
Hardware	Intel Cascade Lake 8260 CPUs
Parallel execution	96 CPU cores per simulation
Memory usage	$\sim$ 160 GB per simulation

Table 6.4: Summary of the CFD simulation setup for the human upper-airway dataset.

Section extraction For each of the 7 healthy patients, the set Σ of two-dimensional cross-sectional slices is extracted from the three-dimensional preprocessed flow fields. The first and last slices are manually positioned to correspond to the anatomically identified start and end of the olfactory region, respectively, while the remaining slices are placed evenly between these two locations. The same slice configuration is then retained for all synthetic geometries generated from each healthy patient, ensuring consistent section placement

across derived samples. Section extraction is performed using solver-consistent surface sampling via `foamPostProcess`, with planar cutting surfaces orthogonal to the main streamwise direction defined in a dedicated configuration file (`surfaces.cfg`), as illustrated in Figure 5.6. The resulting planar polygonal meshes constitute the two-dimensional representations of the flow fields employed throughout the experimental analyses presented in this thesis.

Chapter 7

Concluding Remarks

This thesis addressed a fundamental and often underestimated challenge at the intersection of machine learning and computational fluid dynamics: enabling learning-based inference from CFD simulations in realistic settings. CFD simulations generate data that are extremely high-dimensional, strongly coupled to geometry, and highly sensitive to small perturbations of the computational domain. Each simulation is the outcome of a long and expensive numerical pipeline involving geometry processing, mesh generation, solver configuration, and substantial computational resources. As a result, the primary difficulty in applying ML to CFD does not lie in model selection or optimisation, but in the construction of datasets that are exhaustive, consistent, and informative across samples.

These challenges manifest in the coupled *Large-p, Small-n* regime that characterises most CFD-based inference tasks: the number of available simulations is severely limited by computational cost, while each individual flow field contains millions of degrees of freedom. This imbalance fundamentally constrains the applicability of conventional ML paradigms and makes data efficiency, interpretability, and inductive bias essential requirements.

This thesis explicitly tackled these difficulties. Before any learning step can take place, CFD data must be processed, aligned, reduced, and structured in a way that preserves physical meaning while enabling statistical inference. This process is inherently time-consuming, highly domain-specific, and cannot be delegated to generic end-to-end learning pipelines. In this sense, applying ML to CFD is not a straightforward extension of standard ML workflows, but requires a reformulation of the learning problem itself, starting from data generation and representation.

The contributions of this thesis are primarily methodological. Rather than focusing on increasingly complex learning architectures, the proposed framework addresses the problem at its root by introducing physics-based and geometry-consistent strategies for data augmentation, feature extraction, and

representation alignment. These components enable the extraction of compact and consistent representations from CFD data, allowing ML models to infer quantities that CFD alone cannot compute, such as the presence of defects, pathologies, or other high-level semantic attributes of the physical system.

It is important to stress that achieving this required a substantial investment in data generation and processing. The effort devoted to preparing CFD datasets, working with real CT scans, and ensuring physical and geometric consistency far exceeds the computational cost of training the learning models themselves. This aspect, often invisible in final performance metrics, represents one of the main obstacles to practical ML applications in CFD and one of the key contributions of this work.

From a practical standpoint, this thesis demonstrates that inference from flow fields is not only conceptually appealing but also technically feasible in realistic scenarios. By treating CFD simulations as informative signatures of the underlying physical system, non-computable properties can be characterised through their impact on the flow response. This enables concrete applications in which relevant attributes, such as geometric defects in engineering components or pathological conditions in biomedical contexts, are inferred from simulated flows. In particular, the results obtained on patient-specific airway simulations show that CFD-based inference can support diagnostic tasks by extracting clinically meaningful information from airflow patterns, thereby complementing conventional image-based diagnosis.

Beyond the specific applications considered, this thesis conveys a broader methodological message: progress in ML for CFD-based inference depends less on model sophistication and more on making explicit the physical, geometric, and statistical structure of CFD data. Accordingly, this work advocates a shift in perspective, from learning better models to constructing better learning problems.

Ultimately, the long-term objective of this research direction is to pave the way for end-to-end deep learning pipelines operating directly on CFD data. However, in light of the limitations discussed throughout this thesis, such pipelines are currently not viable in realistic settings and cannot be meaningfully pursued without first addressing the foundational challenges of data construction, alignment, and representation. By explicitly tackling these challenges, the present work establishes the necessary conditions for future fully data-driven approaches that operate on CFD simulations while remaining physically consistent and statistically sound.

7.1 Research Directions

The framework developed in this thesis naturally opens several research directions, which stem from the intrinsic difficulty of applying ML to CFD data

rather than from limitations of specific models. These directions primarily concern data representation, feature design, sensitivity analysis, and computational scalability, and aim at progressively reducing the gap between proof-of-concept studies and deployable inference pipelines.

A first and central research direction concerns the design of more informative and task-adaptive features. As discussed throughout the thesis, interpretability analyses such as SHAP [64] indicates that only a limited subset of features and spatial regions of the flow field effectively drive the model's decisions. These tools can be exploited constructively to guide feature refinement. In particular, combining interpretability with sensitivity analysis would allow the identification of regions where the flow is most responsive to geometric defects or pathological alterations, thereby localising where discriminative information is physically concentrated.

In this context, future work could explore the use of adjoint-based sensitivity information [54] to refine feature definitions, redefine informative regions, or validate the relevance of physics-based partitions. A systematic comparison between regions identified through clustering and those highlighted by adjoint sensitivities could be used to assess whether data-driven region partitions align with the flow structures that are most influential for a given inference task. Such a comparison would make it possible to quantify the consistency between statistically identified regions and physically optimal perturbation directions, and to identify cases in which clustering fails to capture task-relevant sensitivities.

A particularly promising research direction concerns the integration of ML models with adjoint-based concepts in scenarios where the objective functional or loss cannot be explicitly formulated [81]. In many inference problems, the quantity of interest is implicitly defined through high-level semantic labels, making the direct construction of an adjoint problem infeasible. In this setting, ML models could be used to learn surrogate losses or adjoint-like operators, enabling data-driven sensitivity analyses that bridge physics-based methods and learning-based inference. Such hybrid approaches could play a key role in feature selection, region identification, and interpretability when classical adjoint methods are not directly applicable.

Another research avenue concerns computational scalability. Despite the proposed strategies to mitigate data scarcity, CFD simulations remain the dominant computational bottleneck. Future work should investigate the adaptation of the framework to reduced-order or simplified CFD models, provided that the resulting representations preserve the information necessary for inference. Understanding the trade-off between physical fidelity and diagnostic capability is essential to extend ML-based inference to large-scale or time-constrained scenarios.

An additional direction concerns the expansion of the available datasets, both in terms of patient-specific anatomies and pathological conditions. While

the framework proposed in this thesis is explicitly designed to operate under severe data scarcity, extending the dataset with new patients would increase the statistical coverage of anatomical variability and further improve robustness and generalisation. Importantly, such an expansion would not merely increase the number of samples, but would enrich the diversity of geometric configurations on which inference is performed.

Beyond increasing the number of subjects, future work could also address the inclusion of additional pathologies and more complex combinations of conditions. The proposed augmentation strategy naturally supports this extension, as new pathologies can be introduced through expert-defined deformation functions and consistently mapped across anatomies. This would enable the construction of richer multi-label datasets and allow the study of more challenging diagnostic scenarios in which multiple defects or pathological conditions coexist and interact.

From an application perspective, the implications of the proposed future research extend beyond the specific case studies addressed in the thesis. In biomedical settings, the proposed methodology could be further developed into decision-support tools for clinicians, providing access to flow-derived information that is not available through standard imaging alone, even in the presence of limited annotated data. In engineering applications, the same principles can be applied to the detection of defects, damage, ice accretion, or anomalous operating conditions, particularly in scenarios characterised by severe data scarcity and controlled geometric perturbations.

Overall, these research directions reinforce the central outcome of this thesis: applying ML to CFD for inference is a demanding and resource-intensive endeavour, whose difficulty lies primarily in data construction, representation, and interpretation rather than in model design. Continued progress in this area will depend on developing principled methods that make CFD data more tractable and computationally manageable for learning-based tasks.

Appendix A

Functional Maps and Spectral Shape Correspondence

This appendix provides the mathematical background underlying the spectral shape correspondence framework introduced in Chapter 4. The objective is to formalise the Functional Maps paradigm and the associated multi-scale refinement strategies used to estimate dense correspondences between non-rigid surfaces. The presentation is self-contained and focuses on the components that are directly relevant to the deformation transfer pipeline, closely following the methodological foundations introduced in [80, 91, 70, 22, 21].

The deformation transfer strategy adopted in this thesis in Chapter 4 relies on the ability to establish reliable correspondences between surface geometries. This requirement naturally leads to the shape registration problem, which consists of establishing a correspondence $\mathcal{M}$ between points on two shapes. The problem becomes particularly challenging in the presence of non-rigid deformations, which are nevertheless the only transformations capable of describing correspondences between different anatomical instances. Within the scope of this thesis, non-rigid registration is required to relate a reference anatomy S^{ref} to patient-specific surfaces S_i, where geometric variability and partial overlap are unavoidable.

A.1 Functional Maps Formulation

We address the non-rigid shape-registration problem using the Functional Maps framework introduced by Ovsjanikov et al. [80]. Rather than estimating the pointwise correspondence $\mathcal{M}$ directly, this approach represents correspondence as a linear operator acting between spaces of scalar functions defined on the two shapes. To this end, each surface is described through its own functional basis. Specifically, a basis spans a finite-dimensional subspace of $L^2(S)$, the space of square-integrable functions defined on a given 3D surface. In the

following, we denote by $\Phi = \{\phi_i\}$ the functional basis associated with the reference surface S^{ref}, and by $\Psi = \{\psi_j\}$ the corresponding basis defined on a target surface S_i (see Chapter 4). With respect to such a basis, any function $f \in L^2(S^{\text{ref}})$ can be approximated as

$$f = \sum_{i=1}^{l} a_i \phi_i,$$

where the coefficients a_i are obtained by projecting f onto the chosen basis. An analogous expansion holds for functions defined on S_i with respect to the basis Ψ. Also, given two corresponding functions f on S^{ref} and g on S_i, related by an unknown pointwise map $\mathcal{M} : S^{\text{ref}} \to S_i$, the Functional Maps paradigm seeks a correspondence between the associated functional spaces. This correspondence is represented by a linear operator $\mathcal{C} \in \mathbb{R}^{l \times l}$ such that

$$\mathbf{b} = \mathcal{C}\,\mathbf{a}, \tag{A.1}$$

where $\mathbf{a}$ and $\mathbf{b}$ denote the spectral coefficient vectors of f and g in the bases Φ and Ψ, respectively. Estimating correspondence in the functional domain is computationally efficient and more robust to noise and discretisation effects than direct pointwise matching [41]. Eventually, once the functional map $\mathcal{C}$ has been estimated, it can be converted into a dense point-to-point correspondence $\mathcal{M} : S^{\text{ref}} \to S_i$ using standard recovery procedures [21].

A.2 Choice of the Basis and Functional Map Estimation

The choice of basis functions plays a central role in the effectiveness of the Functional Maps framework. Most approaches rely on the eigenfunctions of the Laplace–Beltrami operator, which provide an intrinsic and isometry-invariant representation of surface geometry. Formally, given a Riemannian manifold $\mathcal{M}$, the Laplace–Beltrami operator $\Delta_{\mathcal{M}}$ is defined as

$$\Delta_{\mathcal{M}} f = \nabla_{\mathcal{M}} \cdot \nabla_{\mathcal{M}} f,$$

where $\nabla_{\mathcal{M}}$ denotes the intrinsic gradient on the manifold. The eigenfunctions $\{\phi_i\}_{i \geq 1}$ and associated eigenvalues $\{\lambda_i\}_{i \geq 1}$ are obtained by solving

$$\Delta_{\mathcal{M}} \phi_i = \lambda_i \phi_i,$$

subject to appropriate boundary conditions. These eigenfunctions form an orthonormal basis of $L^2(\mathcal{M})$ and provide a spectral decomposition of functions defined on the surface, ordered from low to high frequencies according to the magnitude of the corresponding eigenvalues.

Correspondence estimation is formulated as an optimisation problem that enforces consistency between pairs of corresponding descriptor functions defined on the two surfaces. A commonly used formulation is

$$\mathcal{C}_i = \arg\min_X \|XA - B\|^2 + \alpha\|\Lambda^{S_i}X - X\Lambda^{S^{\text{ref}}}\|^2, \tag{A.2}$$

where the columns of A and B contain the spectral coefficients of corresponding descriptors on S^{ref} and S_i, and $\Lambda^{S^{\text{ref}}}$, Λ^{S_i} are diagonal matrices of Laplace–Beltrami eigenvalues. The second term promotes approximate commutativity with the Laplacian operator and acts as a regulariser enforcing intrinsic consistency. In practice, the bases are truncated to the first l eigenfunctions, reducing the estimation of correspondence to a compact $l \times l$ matrix.

A.3 Partial Correspondence and Registration in Clutter

The Functional Maps formulation introduced in the previous section assumes that the two shapes involved in the registration process are globally comparable, namely that a large portion of their geometry admits a meaningful correspondence. In the application discussed in Chapter 4, this assumption is violated, as the reference surface S^{ref} corresponds only to a subset of the target surface $\tilde{S}_i$, which may contain significant geometric clutter. In particular, $\tilde{S}_i$ includes anatomically valid but CFD-irrelevant structures, most notably the paranasal sinuses (Figure 4.3a). To explicitly handle this setting, we adopt the partial shape-registration framework proposed by Cosmo et al. [22], which extends the standard Functional Maps paradigm to non-rigid correspondence in cluttered scenes and can be interpreted as a direct generalisation of the optimisation problem introduced in Section A.2, augmented with variables and priors accounting for partial overlap.

In addition to the functional map $\mathcal{C}$, the method jointly estimates two segmentation functions $u : S^{\text{ref}} \to [0,1]$ and $v : \tilde{S}_i \to [0,1]$, which act as soft indicator functions identifying the regions of the two shapes that participate in the correspondence. Regions of the target surface assigned low values by v are interpreted as clutter and progressively excluded from the matching process. Under this formulation, correspondence is obtained by solving

$$\min_{\mathcal{C},\theta,u,v} \|\mathcal{C}A(\eta(u)) - B(\eta(v))\|_{2,1} + \|\mathcal{C}\Phi^T\eta(u) - \Psi^T\eta(v)\|_2^2 + \rho_{\text{corr}}(\mathcal{C},\theta) + \rho_{\text{part}}(u,v), \tag{A.3}$$

which extends the descriptor-preserving formulation (A.2) by explicitly introducing the segmentation variables and their regularisation terms. While the first two terms enforce descriptor consistency and spectral compatibility, ρ_{corr} encodes structural priors on $\mathcal{C}$ (including the slanted-diagonal structure expected

in partial matching), and ρ_{part} promotes spatial coherence and controlled size of the selected regions through Mumford–Shah-type regularisation [55].

Problem (A.3) is non-convex and is solved in practice by alternating optimisation. Given fixed segmentation functions, $\mathcal{C}$ (and the slope parameter θ) is updated by minimising the descriptor mismatch together with the structural priors encoded in ρ_{corr}. Conversely, given a fixed functional map, the segmentation functions (u, v) are updated by minimising ρ_{part} combined with a transport consistency term that encourages agreement between the mapped reference region and the selected target region. The segmentation variables are optimised in their relaxed (continuous) form and subsequently thresholded to obtain binary masks defining the estimated corresponding subsets.

As in the standard Functional Maps pipeline, once the correspondence $\mathcal{C}$ has been estimated in the spectral domain, a dense point-to-point map $\mathcal{M}$ is recovered via nearest-neighbour search in the space of spectral embeddings. In the present implementation, this step relies on the FLANN (Fast Library for Approximate Nearest Neighbour)[1] framework to ensure computational efficiency on high-resolution meshes.

Overall, this formulation provides a principled extension of the Functional Maps framework to partial correspondence scenarios, enabling the automatic identification of the CFD-relevant subset of $\tilde{S}_i$ while preserving the spectral structure and robustness properties of the original approach.

A.4 Multi-scale Refinement via ZoomOut

When working with high-resolution meshes extracted from imaging data, direct estimation of dense correspondence becomes computationally demanding. Moreover, the clutter-aware registration step described above enforces global anatomical consistency between the reference and target surfaces, but is not designed to achieve high local accuracy at the vertex level. To refine correspondence while maintaining stability and efficiency, we therefore employ the ZoomOut algorithm introduced by Melzi et al. [70].

The ZoomOut method incrementally refines the mapping $\mathcal{M}$ between the reference surface S^{ref} and the target surface S_i by progressively increasing the spectral resolution at which the correspondence is represented. Starting from an initial low-resolution functional map $\mathcal{C}_0 \in \mathbb{R}^{l_0 \times l_0}$, or equivalently from a point-to-point map $\mathcal{M}_0$, the method introduces at each iteration additional Laplace–Beltrami eigenfunctions Φ and Ψ, effectively adding higher frequencies to the spectral representation of the correspondence.

[1] `https://github.com/flann-lib/flann`

At iteration $l \geq l_0$, the correspondence is refined through a two-step procedure. First, a point-to-point map $\mathcal{M}_l$ is recovered from the current functional map $\mathcal{C}_l$ by solving

$$\mathcal{M}_l(p) = \arg\min_q \left\| \mathcal{C}_l \, \Psi(q)^\top - \Phi(p)^\top \right\|_2 , \tag{A.4}$$

for all $p \in R$, where $\Phi(p)$ and $\Psi(q)$ denote the rows of the eigenfunction matrices evaluated at vertices p and q, respectively. This nearest-neighbour search is performed in the spectral embedding space and is, in the present implementation, efficiently handled using the FLANN framework.

Second, the recovered pointwise correspondence $\mathcal{M}_l$ is re-encoded as a functional map $\mathcal{C}_{l+1}$ of increased size $(l+1) \times (l+1)$ by projection onto the expanded Laplace–Beltrami bases of the two shapes. By alternating pointwise recovery and functional re-encoding, ZoomOut progressively improves localisation accuracy while preserving the global structure established at lower spectral resolutions.

This coarse-to-fine refinement strategy allows dense and locally accurate correspondences to be computed without directly optimising high-dimensional functional maps. In our pipeline, the final pointwise correspondence $\mathcal{M}$ is used to transfer pathologies and deformation fields from the reference surface to patient-specific geometries. Compared to purely data-driven augmentation strategies [1, 111], this approach preserves explicit control over the applied deformations and guarantees semantic consistency of the generated samples.

Nomenclature

Acronyms

BGMM Bayesian Gaussian Mixture Model

CFD Computational Fluid Dynamics

DA Data Augmentation

DL Deep Learning

DNS Direct Numerical Simulation

ENT Ear, Nose and Throat

FE Feature Extraction

FEM Finite Element Method

HU Hounsfield Units

LES Large Eddy Simulation

MAE Mean Absolute Error

ML Machine Learning

MLPs Multi-Layer Perceptrons

MSE Mean Squared Error

NS Navier-Stokes

PINNs Physics-informed Neural Networks

RBF Radial Basis Functions

SHAP SHapley Additive exPlanations

TKE Turbulent Kinetic Energy

Greek Symbols

Δ	Set of deformation functions defined by ENT experts
δ_i	i-th deformation function in Δ
ν	Kinematic viscosity
ν_t	Turbulent viscosity
Ω	Vorticity
Ω_i	i-th CFD simulation computational domain
Φ	Region extraction operator
ϕ_i	Shapley value associated with the i-th feature
Π	Morphing operator
Π_i	i-th morphing operator
π_i	i-th point-to-point correspondence between boundary vertices
$\psi(r)$	Radial basis function
ρ	Density
Σ_s	s-th transversal section identified in the human upper airways

Roman Symbols

$\boldsymbol{U}$	Velocity vector
$\mathbf{C}_i$	i-th matrix containing the terms of the governing equations
$\mathbf{F}_i$	i-th CFD simulation
$\mathbf{P}_i$	Feature set
$\mathbf{x}_{i,k}$	k-th node of the i-th mesh
$\mathcal{A}$	Data augmentation operator
$\mathcal{D}$	Dataset used to train and test K
$\mathcal{D}^\star$	Augmented Dataset
$\mathcal{F}$	Set of CFD simulations

$\mathcal{I}_i$	i-th projection operator
$\mathcal{M}_i$	Point-to-point mapping between S^{ref} and S_i
$\mathcal{Q}$	Set of flow scalar quantities
$\mathcal{R}$	Set of regions defined over the i-th simulation
$\mathcal{T}_i$	i-th non-rigid deformation based on RBFs
$\mathcal{Y}$	Set of target labels
$\tilde{S}_i$	Surface segmented from the i-th CT scan T_i
a	First NACA digit.
b	Second NACA digit.
E	Resolved part of the enstrophy
K	Machine Learning model
k_t	Resolved part of the turbulent kinetic energy
M_i	i-th discretized computational mesh
n_i	Number of nodes of the i-th mesh
p	Pressure
Q	Volumetric flow rate
q	Generic flow scalar quantity
r_i	Number of regions in $\mathcal{R}_i$
R_j^i	j-th identified regions in the i-th CFD simulation
S^{ref}	Reference geometry
S_i	i-th surface bounding the i-th computational domain
$S_j^{\text{ref},\star}$	Variant of the reference S^{ref} deformed via δ_j
t	Last two NACA digits.
T_i	i-th CT scan
Y_i	label associated with the i-th CFD simulation

N Cardinality of $\mathcal{D}$, i.e., the number of samples in the dataset

Re Reynolds Number

V V denote the number of synthetic variants generated from each S_i

Bibliography

[1] Alvaro Abucide-Armas, Koldo Portal-Porras, Unai Fernandez-Gamiz, Ekaitz Zulueta, and Adrian Teso-Fz-Betoño. "A Data Augmentation-Based Technique for Deep Learning Applied to CFD Simulations". en. In: *Mathematics* 9.16 (Jan. 2021). P. 1843. ISSN: 2227-7390. DOI: 10.3390/math9161843.

[2] N. Ashton, C. Mockett, M. Fuchs, L. Fliessbach, H. Hetmann, T. Knacke, N. Schonwald, V. Skaperdas, G. Fotiadis, A. Walle, and B. Hupertz. "DrivAerML: High-Fidelity Computational Fluid Dynamics Dataset for Road-Car External Aerodynamics". In: *arxiv.org* (2024).

[3] H. Jane Bae and Petros Koumoutsakos. "Scientific multi-agent reinforcement learning for wall-models of turbulent flows". en. In: *Nature Communications* 13.1 (Mar. 2022). P. 1443. ISSN: 2041-1723. DOI: 10.1038/s41467-022-28957-7.

[4] Yoon-Yeong Bae, Eung-Seon Kim, and Minhwan Kim. "Application of compressible Reynolds-averaged governing equations to turbulent mixed convection in supercritical fluids in heated vertical tubes". In: *International Journal of Heat and Fluid Flow* 76 (Apr. 2019), pp. 85–99. ISSN: 0142-727X. DOI: 10.1016/j.ijheatfluidflow.2018.12.003.

[5] Nathan Baker, Frank Alexander, Timo Bremer, Aric Hagberg, Yannis Kevrekidis, Habib Najm, Manish Parashar, Abani Patra, James Sethian, Stefan Wild, Karen Willcox, and Steven Lee. *Workshop Report on Basic Research Needs for Scientific Machine Learning: Core Technologies for Artificial Intelligence*. English. Tech. rep. USDOE Office of Science (SC), Washington, D.C. (United States), Feb. 2019. DOI: 10.2172/1478744.

[6] Sweta Banerjee, R. Krahl, F. Durst, and Ch Zenger. "Presentation of anisotropy properties of turbulence, invariants versus eigenvalue approaches". In: *Journal of Turbulence* 8 (Jan. 2007). DOI: 10.1080/14685240701506896.

[7] Jon Louis Bentley. "Multidimensional binary search trees used for associative searching". In: *Commun. ACM* 18.9 (1975), pp. 509–517. ISSN: 0001-0782. DOI: 10.1145/361002.361007.

[8] Eleonora Biondi. *Simulazione LES del flusso nelle cavità nasali*.

[9] A. de Boer, M. S. van der Schoot, and H. Bijl. "Mesh deformation based on radial basis function interpolation". In: *Computers & Structures*. Fourth MIT Conference on Computational Fluid and Solid Mechanics 85.11 (June 2007), pp. 784–795. ISSN: 0045-7949. DOI: 10.1016/j.compstruc.2007.01.013.

[10] Florent Bonnet, Jocelyn Ahmed Mazari, Paola Cinnella, and Patrick Gallinari. "AIRFRANS: high fidelity computational fluid dynamics dataset for approximating reynolds-averaged Navier-Stokes solutions". In: *Proceedings of the 36th International Conference on Neural Information Processing Systems*. NIPS 2022. New Orleans, LA, USA: Curran Associates Inc., 2022. ISBN: 9781713871088.

[11] M. P. Brenner, J. D. Eldredge, and J. B. Freund. "Perspective on machine learning for advancing fluid mechanics". In: *Physical Review Fluids* 4.10 (Oct. 2019). P. 100501. DOI: 10.1103/PhysRevFluids.4.100501.

[12] Steven Brunton, Bernd Noack, and Petros Koumoutsakos. "Machine Learning for Fluid Mechanics". In: *Annual Review of Fluid Mechanics* 52.1 (Jan. 2020). Pp. 477–508. ISSN: 0066-4189, 1545-4479. DOI: 10.1146/annurev-fluid-010719-060214.

[13] M. D. Buhmann. "Radial basis functions". en. In: *Acta Numerica* 9 (Jan. 2000), pp. 1–38. ISSN: 1474-0508, 0962-4929. DOI: 10.1017/S0962492900000015.

[14] Jared L. Callaham, James V. Koch, Bingni W. Brunton, J. Nathan Kutz, and Steven L. Brunton. "Learning dominant physical processes with data-driven balance models". en. In: *Nature Communications* 12.1 (Feb. 2021). P. 1016. ISSN: 2041-1723. DOI: 10.1038/s41467-021-21331-z.

[15] Katia Capellini, Emanuele Vignali, Emiliano Costa, Emanuele Gasparotti, Marco Evangelos Biancolini, Luigi Landini, Vincenzo Positano, and Simona Celi. "Computational Fluid Dynamic Study for aTAA Hemodynamics: An Integrated Image-Based and Radial Basis Functions Mesh Morphing Approach". eng. In: *Journal of Biomechanical Engineering* 140.11 (Nov. 2018), p. 111007. ISSN: 1528-8951. DOI: 10.1115/1.4040940.

[16] Fabien Casenave, Brian Staber, and Xavier Roynard. *MMGP: a Mesh Morphing Gaussian Process-based machine learning method for regression of physical problems under non-parameterized geometrical variability*. Oct. 2023. DOI: 10.48550/arXiv.2305.12871.

[17] Joshua Cates, Shireen Elhabian, and Ross Whitaker. "Chapter 10 - ShapeWorks: Particle-Based Shape Correspondence and Visualization Software". In: *Statistical Shape and Deformation Analysis*. Academic Press, 2017, pp. 257–298. ISBN: 978-0-12-810493-4. DOI: https://doi.org/10.1016/B978-0-12-810493-4.00012-2.

[18] Ubaldo Cella, Daniele Patrizi, Stefano Porziani, Torbjörn Virdung, and Marco Evangelos Biancolini. "Integration within Fluid Dynamic Solvers of an Advanced Geometric Parameterization Based on Mesh Morphing". en. In: *Fluids* 7.9 (Sept. 2022). P. 310. ISSN: 2311-5521. DOI: 10.3390/fluids7090310.

[19] R. Charles, Hao Su, Kaichun Mo, and Leonidas Guibas. "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation". In: July 2017, pp. 77–85. DOI: 10.1109/CVPR.2017.16.

[20] Haecheon Choi, Jungil Lee, and Hyungmin Park. "Aerodynamics of Heavy Vehicles". In: *Annual Review of Fluid Mechanics* 46 (Dec. 2013). DOI: 10.1146/annurev-fluid-011212-140616.

[21] Michele Colombo, Giacomo Boracchi, and Simone Melzi. "Extracting a functional representation from a dictionary for non-rigid shape matching". In: *Computers & Graphics* 113 (June 2023), pp. 43–56. ISSN: 0097-8493. DOI: 10.1016/j.cag.2023.04.010.

[22] Luca Cosmo, Emanuele Rodolà, Jonathan Masci, Andrea Torsello, and Michael M. Bronstein. "Matching Deformable Objects in Clutter". In: *2016 Fourth International Conference on 3D Vision (3DV)*. Oct. 2016, pp. 1–10. DOI: 10.1109/3DV.2016.10.

[23] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. "RoboNet: Large-Scale Multi-Robot Learning". In: *CoRR* abs/1910.11215 (2019). arXiv: 1910.11215.

[24] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. Miami, FL: IEEE, June 2009, pp. 248–255. ISBN: 978-1-4244-3992-8. DOI: 10.1109/CVPR.2009.5206848.

[25] Nicolas Donati, Abhishek Sharma, and Maks Ovsjanikov. "Deep Geometric Functional Maps: Robust Feature Learning for Shape Correspondence". In: June 2020, pp. 8589–8598. DOI: 10.1109/CVPR42600.2020.00862.

[26] Ting Dong, Wei Zhang, and Mingming Dong. "The novel morphing airfoil based on the bistable composite laminated shell". In: *Nonlinear Dynamics* 111 (Aug. 2023), pp. 1–19. DOI: 10.1007/s11071-023-08820-0.

[27] F. Ducros, Nicoud Franck, and Thierry Poinsot. "Wall-Adapting Local Eddy-Viscosity Models for Simulations in Complex Geometries". In: *Numerical Methods for Fluid Dynamics VI* (Jan. 1998).

[28] Karthik Duraisamy, Gianluca Iaccarino, and Heng Xiao. "Turbulence Modeling in the Age of Data". In: *Annual Review of Fluid Mechanics* 51.1 (Jan. 2019). Pp. 357–377. ISSN: 0066-4189, 1545-4479. DOI: 10.1146/annurev-fluid-010518-040547.

[29] Daniel Eastvedt, Greg Naterer, and Xili Duan. "Detection of Faults in Subsea Pipelines by Flow Monitoring with Regression Supervised Machine Learning". In: *Process Safety and Environmental Protection* 161 (Mar. 2022). DOI: 10.1016/j.psep.2022.03.049.

[30] Hamidreza Eivazi, Mojtaba Tahani, Philipp Schlatter, and Ricardo Vinuesa. "Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations". en. In: *Physics of Fluids* 34.7 (July 2022). ISSN: 1070-6631. DOI: 10.1063/5.0095270.

[31] Mohamed Elrefaie, Angela Dai, and Faez Ahmed. "DrivAerNet: A Parametric Car Dataset for Data-Driven Aerodynamic Design and Graph-Based Drag Prediction". en. In: American Society of Mechanical Engineers Digital Collection, Nov. 2024. DOI: 10.1115/DETC2024-143593.

[32] Mohamed Elrefaie, Florin Morar, Angela Dai, and Faez Ahmed. "DrivAerNet++: A Large-Scale Multimodal Car Dataset with Computational Fluid Dynamics Simulations and Deep Learning Benchmarks". In: *Advances in Neural Information Processing Systems*. Vol. 37. Curran Associates, Inc., 2024, pp. 499–536.

[33] N. Benjamin Erichson, Lionel Mathelin, J. Nathan Kutz, and Steven L. Brunton. "Randomized Dynamic Mode Decomposition". In: *SIAM Journal on Applied Dynamical Systems* 18.4 (Jan. 2019). Pp. 1867–1891. DOI: 10.1137/18M1215013.

[34] Michael D. Escobar and Mike West. "Bayesian Density Estimation and Inference Using Mixtures". In: *Journal of the American Statistical Association* 90.430 (1995). Pp. 577–588. DOI: 10.2307/2291069.

[35] C. Farhat, C. Degand, B. Koobus, and M. Lesoinne. "Torsional springs for two-dimensional dynamic unstructured fluid meshes". In: *Computer Methods in Applied Mechanics and Engineering* 163.1 (Sept. 1998), pp. 231–245. ISSN: 0045-7825. DOI: 10.1016/S0045-7825(98)00016-4.

[36] F. Foroozan, V. Guerrero, A. Ianiro, and S. Discetti. "Unsupervised modelling of a transitional boundary layer". en. In: *Journal of Fluid Mechanics* 929 (Dec. 2021), A3. ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2021.829.

[37] Stefania Fresca, Luca Dede', and Andrea Manzoni. "A Comprehensive Deep Learning-Based Approach to Reduced Order Modeling of Nonlinear Time-Dependent Parametrized PDEs". en. In: *Journal of Scientific Computing* 87.2 (Apr. 2021), p. 61. ISSN: 1573-7691. DOI: 10.1007/s10915-021-01462-7.

[38] Kai Fukami, Koji Fukagata, and Kunihiko Taira. "Assessment of supervised machine learning methods for fluid flows". In: *Theoretical and Computational Fluid Dynamics* 34.4 (Aug. 2020). Pp. 497–519. ISSN: 0935-4964, 1432-2250. DOI: 10.1007/s00162-020-00518-y.

[39] Kai Fukami, Romit Maulik, Nesar Ramachandra, Koji Fukagata, and Kunihiko Taira. "Global field reconstruction from sparse sensors with Voronoi tessellation-assisted deep learning". en. In: *Nature Machine Intelligence* 3.11 (Nov. 2021). Pp. 945–951. ISSN: 2522-5839. DOI: 10.1038/s42256-021-00402-2.

[40] Nicholas Geneva and Nicholas Zabaras. "Modeling the dynamics of PDE systems with physics-constrained deep auto-regressive networks". In: *Journal of Computational Physics* 403 (Feb. 2020), p. 109056. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2019.109056.

[41] Zan Gojcic, Caifa Zhou, Jan D. Wegner, and Andreas Wieser. "The Perfect Match: 3D Point Cloud Matching With Smoothed Densities". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2019, pp. 5540–5549. DOI: 10.1109/CVPR.2019.00569.

[42] Carlos González and Joaquín Fernández-León. "A Machine Learning Model to Detect Flow Disturbances during Manufacturing of Composites by Liquid Moulding". In: *Journal of Composites Science* 4 (June 2020), p. 71. DOI: 10.3390/jcs4020071.

[43] Corrado Groth, Emiliano Costa, and Marco Evangelos Biancolini. "RBF-based mesh morphing approach to perform icing simulations in the aviation sector". In: *Aircraft Engineering and Aerospace Technology* 91.4 (Feb. 2019). Pp. 620–633. ISSN: 1748-8842. DOI: 10.1108/AEAT-07-2018-0178.

[44] O. Hassan, E. J. Probert, and K. Morgan. "Unstructured mesh procedures for the simulation of three-dimensional transient compressible inviscid flows with moving boundary components". en. In: *International Journal for Numerical Methods in Fluids* 27.1-4 (1998). Pp. 41–55. ISSN: 1097-0363. DOI: 10.1002/(SICI)1097-0363(199801)27:1/4<41::AID-FLD649>3.0.CO;2-5.

[45] Tobias Heimann and Hans-Peter Meinzer. “Statistical shape models for 3D medical image segmentation: A review”. In: *Medical Image Analysis* 13.4 (Aug. 2009), pp. 543–563. ISSN: 1361-8415. DOI: 10.1016/j.media.2009.05.004.

[46] Qixing Huang, Fan Wang, and Leonidas Guibas. “Functional map networks for analyzing and exploring large shape collections”. In: *ACM Trans. Graph.* 33.4 (July 2014), 36:1–36:11. ISSN: 0730-0301. DOI: 10.1145/2601097.2601111.

[47] J. Hunt, Alan Wray, and Parviz Moin. “Eddies, streams, and convergence zones in turbulent flows”. In: *Studying Turbulence Using Numerical Simulation Databases* -1 (Nov. 1988), pp. 193–208.

[48] Phuoc-Hai Huynh, Van Hoa Nguyen, and Thanh-Nghi Do. “Improvements in the Large p, Small n Classification Issue”. In: *SN Computer Science* 1.4 (June 2020), p. 207. ISSN: 2661-8907. DOI: 10.1007/s42979-020-00210-2.

[49] Hanhui Jin, Rongrui Fan, M.J. Zeng, and K.F. Cen. “Large eddy simulation of inhaled particle deposition within the human upper respiratory tract”. In: *Journal of Aerosol Science* 38 (Mar. 2007), pp. 257–268. DOI: 10.1016/j.jaerosci.2006.09.008.

[50] Michael Jordan and T.M. Mitchell. “Machine Learning: Trends, Perspectives, and Prospects”. In: *Science (New York, N.Y.)* 349 (July 2015), pp. 255–60. DOI: 10.1126/science.aaa8415.

[51] Eurika Kaiser, Bernd R. Noack, Laurent Cordier, Andreas Spohn, Marc Segond, Markus Abel, Guillaume Daviller, Jan Östh, Siniša Krajnović, and Robert K. Niven. “Cluster-based reduced-order modelling of a mixing layer”. en. In: *Journal of Fluid Mechanics* 754 (Sept. 2014), pp. 365–414. ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2014.355.

[52] Kalin Kanov, Randal Burns, Cristian Lalescu, and Gregory Eyink. “ The Johns Hopkins Turbulence Databases: An Open Simulation Laboratory for Turbulence Research ”. In: *Computing in Science & Engineering* 17.05 (Sept. 2015), pp. 10–17. ISSN: 1558-366X. DOI: 10.1109/MCSE.2015.103.

[53] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. “Physics-informed machine learning”. en. In: *Nature Reviews Physics* 3.6 (June 2021). Pp. 422–440. ISSN: 2522-5820. DOI: 10.1038/s42254-021-00314-5.

[54] Gaetan K. W. Kenway, Charles A. Mader, Ping He, and Joaquim R. R. A. Martins. “Effective adjoint approaches for computational fluid dynamics”. In: *Progress in Aerospace Sciences* 110 (2019), p. 100542. ISSN: 0376-0421. DOI: https://doi.org/10.1016/j.paerosci.2019.05.002.

[55] Esther Klann and Ronny Ramlau. "Regularization Properties of Mumford-Shah-Type Functionals with Perimeter and Norm Constraints for Linear Ill-Posed Problems". In: *SIAM Journal on Imaging Sciences [electronic only]* 6 (Feb. 2013). DOI: 10.1137/110858422.

[56] Alex Krizhevsky. "Learning Multiple Layers of Features from Tiny Images". In: *University of Toronto* (May 2012).

[57] Corentin J. Lapeyre, Antony Misdariis, Nicolas Cazard, Denis Veynante, and Thierry Poinsot. "Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates". In: *Combustion and Flame* 203 (May 2019), pp. 255–264. ISSN: 0010-2180. DOI: 10.1016/j.combustflame.2019.02.019.

[58] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (Nov. 1998), pp. 2278–2324. ISSN: 1558-2256. DOI: 10.1109/5.726791.

[59] Yann LeCun, Yere Yere, and Geoffrey Hinton. "Deep Learning". In: *Nature* 521 (May 2015), pp. 436–44. DOI: 10.1038/nature14539.

[60] Myoungkyu Lee, Nicholas Malaya, and Robert D. Moser. "Petascale direct numerical simulation of turbulent channel flow on up to 786K cores". In: *SC '13: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis*. Nov. 2013, pp. 1–11. DOI: 10.1145/2503210.2503298.

[61] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft COCO: Common Objects in Context". en. In: *Computer Vision – ECCV 2014*. Cham: Springer International Publishing, 2014, pp. 740–755. ISBN: 978-3-319-10602-1. DOI: 10.1007/978-3-319-10602-1_48.

[62] J. Ling and J. Templeton. "Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty". In: *Physics of Fluids* 27.8 (Aug. 2015), p. 085103. ISSN: 1070-6631. DOI: 10.1063/1.4927765.

[63] Julia Ling, Andrew Kurzawski, and Jeremy Templeton. "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance". en. In: *Journal of Fluid Mechanics* 807 (Nov. 2016). Pp. 155–166. ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2016.615.

[64] Scott M Lundberg and Su-In Lee. "A Unified Approach to Interpreting Model Predictions". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017.

[65] Mori Mani and Andrew Dorgan. "A Perspective on the State of Aerospace Computational Fluid Dynamics Technology". In: *Annual Review of Fluid Mechanics* 55 (Jan. 2023), pp. 431–457. DOI: 10.1146/annurev-fluid-120720-124800.

[66] Riccardo Margheritti, Andrea Schillaci, Carlotta Pipolo, Maurizio Quadrio, and Giacomo Boracchi. "Leveraging Computational Geometry for Data Augmentation in Medical Flow Fields Classification". In: *Engineering Applications of Neural Networks*. Cham: Springer Nature Switzerland, 2025, pp. 109–122. ISBN: 978-3-031-96199-1. DOI: 10.1007/978-3-031-96199-1_9.

[67] Riccardo Margheritti, Onofrio Semeraro, Maurizio Quadrio, and Giacomo Boracchi. "Feature Extraction from Flow Fields: Physics-Based Clustering and Morphing with Applications". In: *Applied Sciences* 15.23 (2025). ISSN: 2076-3417. DOI: 10.3390/app152312421.

[68] Riccardo Margheritti, Onofrio Semeraro, Maurizio Quadrio, and Giacomo Boracchi. "Physics-Based Region Clustering to Boost Inference on Computational Fluid Dynamics Flow Fields". In: *Machine Learning and Knowledge Discovery in Databases. Applied Data Science Track and Demo Track*. Cham: Springer Nature Switzerland, 2026, pp. 3–20. ISBN: 978-3-032-06129-4. DOI: 10.1007/978-3-032-06129-4_1.

[69] Ryley McConkey, Eugene Yee, and Fue-Sang Lien. "On the Generalizability of Machine-Learning-Assisted Anisotropy Mappings for Predictive Turbulence Modelling". In: *International Journal of Computational Fluid Dynamics* 36.7 (Aug. 2022). Pp. 555–577. ISSN: 1061-8562. DOI: 10.1080/10618562.2022.2113520.

[70] Simone Melzi, Jing Ren, Emanuele Rodolà, Abhishek Sharma, Peter Wonka, and Maks Ovsjanikov. "ZoomOut: spectral upsampling for efficient shape correspondence". In: *ACM Transactions on Graphics* 38.6 (Nov. 2019), 155:1–155:14. ISSN: 0730-0301. DOI: 10.1145/3355089.3356524.

[71] M A Mendez, D Hess, B B Watz, and J-M Buchlin. "Multiscale proper orthogonal decomposition (mPOD) of TR-PIV data—a case study on stationary and transient cylinder wake flows". In: *Measurement Science and Technology* 31.9 (June 2020). P. 094014. DOI: 10.1088/1361-6501/ab82be.

[72] M. A. Mendez, M. Balabane, and J.-M. Buchlin. "Multi-scale proper orthogonal decomposition of complex fluid flows". In: *Journal of Fluid Mechanics* 870 (2019), pp. 988–1036. DOI: 10.1017/jfm.2019.212.

[73] Miguel A Mendez. "Linear and nonlinear dimensionality reduction from fluid mechanics to machine learning". In: *Measurement Science and Technology* 34.4 (Jan. 2023). P. 042001. DOI: 10.1088/1361-6501/acaffe.

[74] Miguel A. Mendez, Andrea Ianiro, Bernd R. Noack, and Steven L. Brunton, eds. *Data-Driven Fluid Mechanics: Combining First Principles and Machine Learning*. Cambridge: Cambridge University Press, 2023. ISBN: 978-1-108-84214-3. DOI: 10.1017/9781108896214.

[75] Arvind T. Mohan, Nicholas Lubbers, Misha Chertkov, and Daniel Livescu. "Embedding hard physical constraints in neural network coarse-graining of three-dimensional turbulence". In: *Physical Review Fluids* 8.1 (Jan. 2023). P. 014604. DOI: 10.1103/PhysRevFluids.8.014604.

[76] Takaaki Murata, Kai Fukami, and Koji Fukagata. "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics". en. In: *Journal of Fluid Mechanics* 882 (Jan. 2020), A13. ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2019.822.

[77] Roger B. Myerson. *Game Theory: Analysis of Conflict*. Harvard University Press, 1991. ISBN: 978-0-674-34116-6. DOI: 10.2307/j.ctvjsf522.

[78] Dorian Nogneng and Maks Ovsjanikov. "Informative Descriptor Preservation via Commutativity for Shape Matching". en. In: *Computer Graphics Forum* 36.2 (2017). Pp. 259–267. ISSN: 1467-8659. DOI: 10.1111/cgf.13124.

[79] Jan Oldenburg, Finja Borowski, Alper Öner, Klaus-Peter Schmitz, and Michael Stiehm. "Geometry aware physics informed neural network surrogate for solving Navier–Stokes equation (GAPINN)". In: *Advanced Modeling and Simulation in Engineering Sciences* 9.1 (June 2022), p. 8. ISSN: 2213-7467. DOI: 10.1186/s40323-022-00221-z.

[80] Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. "Functional maps: a flexible representation of maps between shapes". In: *ACM Transactions on Graphics* 31.4 (2012), 30:1–30:11. ISSN: 0730-0301. DOI: 10.1145/2185520.2185526.

[81] John Papadakis and Eftychia Karasmani. "Gradient of the Cost Function Via the Adjoint Method for Underwater Acoustic Inversion". In: *Journal of Computational Acoustics* (May 2019), p. 21. DOI: 10.1142/S2591728519500105.

[82] Scott Parry. "Free-form deformation of solid geometric models". In: *ACM Siggraph Computer Graphics*. Vol. 20. Aug. 1986, pp. 151–160. ISBN: 0-89791-196-2. DOI: 10.1145/15886.15903.

[83] Stephen B. Pope. *Turbulent Flows*. en. Aug. 2000. DOI: 10.1017/CBO9780511840531.

[84] Maurizio Quadrio, Carlotta Pipolo, Stefano Corti, Francesco Messina, Chiara Pesci, Alberto M. Saibene, Samuele Zampini, and Giovanni Felisati. "Effects of CT resolution and radiodensity threshold on the CFD evaluation of nasal airflow". eng. In: *Medical & Biological Engineering & Computing* 54.2-3 (Mar. 2016), pp. 411–419. ISSN: 1741-0444. DOI: 10.1007/s11517-015-1325-4.

[85] Michele Quattromini, Michele Bucci, Stefania Cherubini, and Onofrio Semeraro. "Active learning of data-assimilation closures using graph neural networks". In: *Theoretical and Computational Fluid Dynamics* 39 (Jan. 2025). DOI: 10.1007/s00162-025-00737-1.

[86] Michele Quattromini, Michele Alessandro Bucci, Stefania Cherubini, and Onofrio Semeraro. "Operator learning of RANS equations: a Graph Neural Network closure model". Mar. 2023. DOI: 10.48550/arXiv.2303.03806.

[87] Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. "Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations". In: *Science* 367.6481 (Feb. 2020). Pp. 1026–1030. DOI: 10.1126/science.aaw4741.

[88] Dakota Ramos, Andrew Glaws, Ryan King, Bumseok Lee, Olga Doronina, James Baeder, Ganesh Vijayakumar, and Zachary Grey. *Airfoil Computational Fluid Dynamics - 2k shapes, 25 AoA's, 3 Re numbers.* Feb. 2023. DOI: 10.25984/2222586.

[89] Dakota Ramos, Andrew Glaws, Ryan King, Bumseok Lee, Olga Doronina, James Baeder, Ganesh Vijayakumar, and Zachary Grey. *Airfoil Computational Fluid Dynamics - 2k shapes, 25 AoA's, 3 Re numbers.* Open Energy Data Initiative (OEDI), National Renewable Energy Laboratory (NREL), https://doi.org/10.25984/2222586. 2023. DOI: 10.25984/2222586.

[90] T. C. S. Rendall and C. B. Allen. "Improved radial basis function fluid–structure coupling via efficient localized implementation". en. In: *International Journal for Numerical Methods in Engineering* 78.10 (2009). Pp. 1188–1208. ISSN: 1097-0207. DOI: 10.1002/nme.2526.

[91] E. Rodolà, L. Cosmo, M. M. Bronstein, A. Torsello, and D. Cremers. "Partial Functional Correspondence". In: *Computer Graphics Forum* 36.1 (2017). Pp. 222–236. ISSN: 1467-8659. DOI: 10.1111/cgf.12797.

[92] Clarence W. Rowley, Igor Mezić, Shervin Bagheri, Philipp Schlatter, and Dan S. Henningson. "Spectral analysis of nonlinear flows". en. In: *Journal of Fluid Mechanics* 641 (Dec. 2009), pp. 115–127. ISSN: 1469-7645, 0022-1120. DOI: 10.1017/S0022112009992059.

[93] Ettore Saetta and Renato Tognaccini. "Identification of flow field regions by Machine Learning". In: *AIAA SCITECH 2022 Forum*. American Institute of Aeronautics and Astronautics. DOI: 10.2514/6.2022-0457.

[94] Iqbal Sarker. "Machine Learning: Algorithms, Real-World Applications and Research Directions". In: *SN Computer Science* 2 (Mar. 2021). DOI: 10.1007/s42979-021-00592-x.

[95] Andrea Schillaci, Kazuto Hasegawa, Carlotta Pipolo, Giacomo Boracchi, and Maurizio Quadrio. "Comparing flow-based and anatomy-based features in the data-driven study of nasal pathologies". In: *Flow* 4 (Apr. 2024). DOI: 10.1017/flo.2024.3.

[96] Andrea Schillaci, Maurizio Quadrio, and Giacomo Boracchi. *A database of CFD-computed flow fields around airfoils for machine-learning applications*. 2021. DOI: 10.5281/zenodo.4638071.

[97] Andrea Schillaci, Maurizio Quadrio, Carlotta Pipolo, Marcello Restelli, and Giacomo Boracchi. "Inferring Functional Properties from Fluid Dynamics Features". In: *2020 25th International Conference on Pattern Recognition (ICPR)*. Jan. 2021, pp. 4091–4098. DOI: 10.1109/ICPR48806.2021.9412157.

[98] François G. Schmitt. "About Boussinesq's turbulent viscosity hypothesis: historical remarks and a direct evaluation of its validity". In: *Comptes Rendus Mécanique* 335.9 (2007), pp. 617–627. ISSN: 1631-0721. DOI: https://doi.org/10.1016/j.crme.2007.08.004.

[99] Connor Shorten and Taghi M. Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *Journal of Big Data* 6.1 (July 2019), p. 60. ISSN: 2196-1115. DOI: 10.1186/s40537-019-0197-0.

[100] P. Spalart and S. Allmaras. "A one-equation turbulence model for aerodynamic flows". In: *30th Aerospace Sciences Meeting and Exhibit*. Aerospace Sciences Meetings. American Institute of Aeronautics and Astronautics, Jan. 1992. DOI: 10.2514/6.1992-439.

[101] K. Stein, T. Tezduyar, and R. Benney. "Mesh moving techniques for fluid-structure interactions with large displacements". In: *Journal of Applied Mechanics, Transactions ASME* 70.1 (2003), pp. 58–63. ISSN: 0021-8936. DOI: 10.1115/1.1530635.

[102] Kunihiko Taira, Steven L. Brunton, Scott T. M. Dawson, Clarence W. Rowley, Tim Colonius, Beverley J. McKeon, Oliver T. Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S. Ukeiley. "Modal Analysis of Fluid Flows: An Overview". In: *AIAA Journal* 55.12 (Dec. 2017). Pp. 4013–4041. ISSN: 0001-1452. DOI: 10.2514/1.J056060.

[103] Kunihiko Taira, Maziar S. Hemati, Steven L. Brunton, Yiyang Sun, Karthik Duraisamy, Shervin Bagheri, Scott T. M. Dawson, and Chi-An Yeh. "Modal Analysis of Fluid Flows: Applications and Outlook". In: *AIAA Journal* 58.3 (Mar. 2020). Pp. 998–1022. ISSN: 0001-1452. DOI: 10.2514/1.J058462.

[104] C.A. Taylor and Carlos Figueroa. "Patient-Specific Modeling of Cardiovascular Mechanics". In: *Annual review of biomedical engineering* 11 (May 2009), pp. 109–34. DOI: 10.1146/annurev.bioeng.10.061807.160521.

[105] Nils Thuerey, Konstantin Weißenow, Lukas Prantl, and Xiangyu Hu. "Deep Learning Methods for Reynolds-Averaged Navier–Stokes Simulations of Airfoil Flows". In: *AIAA Journal* 58.1 (Jan. 2020). Pp. 25–36. ISSN: 0001-1452. DOI: 10.2514/1.J058291.

[106] Jonathan Tran, Chi-An Yeh, and Kunihiko Taira. *Using Optimal Transport Aligned Latent Embeddings for Separated Flow Analysis*. Sept. 2025. DOI: 10.48550/arXiv.2509.07318.

[107] Ricardo Vinuesa and Steven L. Brunton. "Enhancing computational fluid dynamics with machine learning". en. In: *Nature Computational Science* 2.6 (June 2022). Pp. 358–366. ISSN: 2662-8457. DOI: 10.1038/s43588-022-00264-7.

[108] Jonathan Viquerat, Jean Rabault, Alexander Kuhnle, Hassan Ghraieb, Aurélien Larcher, and Elie Hachem. "Direct shape optimization through deep reinforcement learning". In: *Journal of Computational Physics* 428 (Mar. 2021), p. 110080. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2020.110080.

[109] Ganwei Wang and Sibo Cheng. "Can foundation language models predict fluid dynamics?" In: *Engineering Applications of Artificial Intelligence* 158 (Oct. 2025), p. 111427. ISSN: 0952-1976. DOI: 10.1016/j.engappai.2025.111427.

[110] Jack Weatheritt and Richard Sandberg. "A novel evolutionary algorithm applied to algebraic modifications of the RANS stress–strain relationship". In: *Journal of Computational Physics* 325 (Nov. 2016), pp. 22–37. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2016.08.015.

[111] Haizhou Wu, Xuejun Liu, Wei An, and Hongqiang Lyu. "A generative deep learning framework for airfoil flow field prediction with sparse data". en. In: *Chinese Journal of Aeronautics* 35.1 (Jan. 2022), pp. 470–484. ISSN: 1000-9361. DOI: 10.1016/j.cja.2021.02.012.

[112] Pin Wu, Kaikai Pan, Lulu Ji, Siquan Gong, Weibing Feng, Wenyan Yuan, and Christopher Pain. "Navier–stokes Generative Adversarial Network: a physics-informed deep learning model for fluid flow generation". en. In: *Neural Computing and Applications* 34.14 (July 2022), pp. 11539–11552. ISSN: 1433-3058. DOI: 10.1007/s00521-022-07042-6.

[113] Mostafa Zakeri, Mohammad Aziznia, Amirhossein Atef, and Azadeh Jafari. "Hemodynamic predictors of cerebral aneurysm rupture: A machine learning approach". In: *Physics of Fluids* 36 (Sept. 2024). DOI: 10.1063/5.0224289.

[114] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip Torr, and Vladlen Koltun. "Point Transformer". In: Oct. 2021, pp. 16239–16248. DOI: 10.1109/ICCV48922.2021.01595.

[115] Yaomin Zhao, Harshal D. Akolekar, Jack Weatheritt, Vittorio Michelassi, and Richard D. Sandberg. "RANS turbulence model development using CFD-driven machine learning". In: *Journal of Computational Physics* 411 (June 2020), p. 109413. ISSN: 0021-9991. DOI: 10.1016/j.jcp.2020.109413.

This Ph.D. thesis has been typeset by means of the TEX-system facilities. The typesetting engine was pdfLATEX. The document class was `toptesi`, by Claudio Beccari, with option `tipotesi=scudo`. This class is available in every up-to-date and complete TEX-system installation.