Dimensionality Reduction for Validating Engine Flow Simulations

Samuel J. Baker

Balliol College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2024

Acknowledgements

First and foremost, I would like to thank my supervisors Dr. Leo Fang, Prof. Martin Davy, and Prof. Felix Leach. Thank you for guiding and investing in me. It's been an honour to learn from you, and I will forever cherish this experience. I would also like to express my gratitude to Jaguar Land Rover, Siemens Digital Industries Software, and EPSRC for funding this research.

Thanks to the Thermal Propulsion Systems Research Group, and the people that made this house into a home. To my friends Abdullah Bajwa, Kharthik Chakravarthy, Michael Hobley, Sumit Joshi, Laurent Le Page, Maruthi Mallardi, Chris Nicholls, Hannah Rana, Priyav Shah, Li Shen, and Chris Willman. Thank you for your support, good humour, and for showing me what is possible.

To my cohort and great friends Qichi He, Tejo Jehart, Varun Shankar, Sam White, Ruixuan Zhu, and Dan Bundred. Each of you constantly inspires me, and I am so proud of what we have achieved together. Thank you for the good times. To Niko Sekularac, thank you for your advice and friendship from the very beginning. To my essential Oxford support network of Markus Baumgartner, Calvin Chan, Michael Cheng, Ben Isachsen, Josiah Senu, Martin Wafula, Asang Wankhede, Markella Zormpa, the BCBC family, and the Lightweight Blue Boat of 2023. A sincere thanks to you all for making this such a fun and enriching journey. We have filled the unforgiving minute with sixty seconds' worth of distance run.

Klara, you arrived at the turning of the tide and have been a rock ever since. I can't thank you enough for supporting me. Abbott and Papap, you've kept in step with me throughout this whole journey. I am incredibly lucky to have such wonderful mentors and companions as grandparents. I hope I can grow into a fraction of the people that you are one day. Charlie and Libby, here's some bedtime reading for you. Or a doorstop. I hope it goes some way to repaying you for always cheering me on. Mum, this is the hardest one to write. You're the foundation of all of this. You have always picked me up when I've been down, shared with me in the highs, and taught me so much. Thank you from the bottom of my heart.

Dimensionality Reduction for Validating Engine Flow Simulations

Samuel J. Baker

Abstract

As the form of propulsion technology that is most prevalent among road and marine transport, the internal combustion engine (ICE) has a significant role to play in supporting global human development and decarbonising the transport sector. Simulations of the turbulent airflows through ICEs using computational fluid dynamics (CFD) can accelerate the rate at which new concept designs and configurations can be tested and iterated. Indeed, the turbulent flows through an ICE are essential for promoting optimal fuel-air mixing and efficient combustion. However, practical CFD simulations commonly used in industry are not fully predictive, so CFD results often need to be validated against experimental data to ensure sufficient accuracy. Such validation processes are not straightforwards, as CFD and experimental data are characterised by fundamental differences such as the types of uncertainty and error that occur, the amount of data that can be gathered, and the extent of the physical domain that can be easily accessed.

This thesis aims to develop robust CFD validation processes by leveraging the respective advantages of experimental and simulated data with the use of state-of-the-art numerical methods. Recent studies indicate that dimensionality reduction techniques, which are widely used for compressing data, have significant potential for analysing fluid mechanics data, partly due to the connections that can be drawn between the dominant features of a dataset and the coherent structures in a turbulent flow. The first results section of this thesis uses two linear dimensionality reduction methods in order to create validation targets from data. Validation targets are typically created by simply averaging the data, but if vector data (such as turbulent flow velocities) are being considered in the average, opposing vectors can cancel out leading the resultant magnitudes to become diminished. This effect is exacerbated in variational flows such as engines experiencing cycle-to-cycle variations (CCVs).

A novel solution to this problem is proposed using a dimensionality reduction method known as sparsity-promoting dynamic mode decomposition (SPDMD). This technique is shown to significantly outperform previously-proposed methods of validation target creation, due to the ability to collapse an ensemble of vector fields into a single validation target while retaining the original vector magnitudes. This finding is used to

develop CFD validation pipelines for both Reynolds-averaged Navier-Stokes (RANS) and large-eddy simulation (LES) data. As a result, the similarity between simulated and experimental datasets can be more accurately quantified, improving diagnostic capabilities.

A range of linear and non-linear dimensionality reduction methods are then leveraged in order to investigate the possibility of accurately reconstructing data from vector fields characterised by large gaps. Although CFD data can be obtained at any location within the domain in question, the acquisition of experimental image data is often limited by physical factors such as restricted lines of sight and occlusions due to geometrical constraints of the experimental rig. Accurately predicting the turbulent flow behaviour inside these gaps would therefore facilitate more complete comparisons of CFD and experimental data, in addition to providing additional insights into the flow. To date, there has been a lack of discussion on the best way to evaluate the performance of an inpainting method for turbulent flows; discrepancies in how the problems are established obfuscate the degree to which the results from a single study are relevant to more general usage. The computer vision research community has been adept in establishing open-source, reproducible benchmarks that can assess model performance across a range of relevant tasks, which has facilitated the development of improved methods. However, the introduction of ML into engine flow field research is more recent, and there is currently a noticeable absence of benchmarks available for standard tasks such as inpainting in turbulent flows.

Therefore, the final section of work presented in thesis establishes such a benchmark for turbulent engine flows, known as EngineBench. The data used are the well-known, open-source TCC-III data, and the inpainting task is to predict the flow inside large, block gaps at the edges of the field of view. This challenges the ML models and reflects a range of realistic scenarios whereby optical access is more difficult to achieve next to a wall than in the centre of the domain in question. A full codebase and set of tutorials have been published along with the data for ease of use on behalf of the wider community, with all relevant links centralised on the landing page `https://eng.ox.ac.uk/tpsrg/research/enginebench/`. Version control is managed via the github repository also linked on this webpage. The results of the initial benchmarking showed that UNet-based neural networks were able to reconstruct the flow inside the edge gaps to a reasonable degree of accuracy, significantly outperforming both the gappy proper orthogonal decomposition (GPOD) and the context encoder generative adversarial network (CE-GAN) across all considered metrics for two different gap sizes. The reasons for the success are linked to the fact that the UNet models were able to utilise contextual information in the centres of the images more effectively, likely due to the skip connections present in these architectures. These results show that accurate flow reconstruction in highly challenging scenarios is possible, demonstrating potential to greatly enhance turbulent flow diagnostics and CFD validation processes.

Contents

Abbreviations

ADMM . Alternating direction method of multipliers

aTDCf .. After the firing TDC

BCE .... Binary cross entropy

BEV .... Battery electric vehicle

bTDCf .. Before the firing TDC

CAD .... Crank angle degrees

CCV .... Cycle to cycle variation

CC ..... Convergence-checking gaps for GPOD convergence

CDA .... Cylinder deactivation

CFD .... Computational fluid dynamics

CFL ..... Courant–Friedrichs–Lewy

CI ..... Compression ignition

CNN .... Convolutional neural network

CTP .... Cross-tumble plane

DI ..... Direct injection

DMD ... Dynamic mode decomposition

DNS .... Direct numerical simulation

DOC .... Diesel oxidation catalyst

DPF .... Diesel particulate filter

EM ..... Ensemble mean

GAN .... Generative adversarial network

GB ..... Gigabytes

GCI ..... Gasoline compression ignition

GDI ..... Gasoline direct injection

GPF .... Gasoline particulate filter

GPOD .. Gappy proper orthogonal decomposition

HEV .... Hybrid electric vehicle

ICE ..... Internal combustion engine

KL ..... Kullback–Leibler divergence

LES ..... Large eddy simulation

LNT .... Lean NO_X trap

LSP ..... Lower swirl plane

MB ..... Megabytes

MF ..... Median filter

MI ..... Magnitude index

ML ..... Machine learning

MSE .... Mean square error

NN ..... Neural network

PIV ..... Particle image velocimetry

PI ..... Positive ignition

PN ..... Particulate number

POD .... Proper orthogonal decomposition

RANS .. Reynolds-averaged Navier-Stokes

Re ..... Reynolds number

RI ..... Relevance index

rpm ..... Revolutions per minute

RS ..... Reynolds stress

SACI .... Spark-assisted compression ignition

SCR .... Selective catalyst reduction

SI ..... Spark ignition

SPDMD Sparsity-promoting DMD

SPOD ... Spectral proper orthogonal decomposition

SVD .... Singular value decomposition

TDC .... Top dead-centre

TJI ..... Turbulent jet ignition

TP ..... Tumble plane

TWC ... Three-way catalyst

UHC .... Unburned hydrocarbons

VVT .... Variable valve timing

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Decarbonising the transport sector

The issue of climate change is one of urgency, with countries around the world declaring climate emergencies and committing to net zero carbon strategies. Road transport accounts for 12% of global greenhouse gas emissions, with the transport sector as a whole contributing to 16% [1]. The scale of this problem is large, with well over 1 billion cars on the road, the majority of which rely on the internal combustion engine (ICE) to provide power today [2]. Low-cost transport is one of the leading enablers of human development [3]; indeed, the poverty rate for US households without cars has risen since 1960, with the wealth of these households declining both in absolute terms and relative to households with cars [4]. The ICE has given more people access to more opportunities, but burning hydrocarbon fuels in ICE vehicles results in emissions of carbon monoxide (CO), unburned hydrocarbons (UHC), soot, and nitrogen oxides (NO_X), in addition to carbon dioxide (CO_2). The challenge is therefore to develop affordable powertrain technology that can facilitate human development while minimising impacts on the environment and human health.

Alternative powertrain options include fully electric (tending to mean battery electric vehicles or BEVs), and electrified vehicles (which include the different types of hybrids). Mild hybrid electric vehicles (MHEVs) recover energy through braking for use in acceleration assistance, full hybrids (FHEVs) include the ability to recharge the battery with the ICE, and plug-in hybrids (PHEVs) facilitate the use of larger batteries by providing external charge capabilities [5]. Increases in battery size and usage tend to mean less use of the ICE, and therefore fewer tailpipe emissions. In addition, as renewable energy becomes more widespread, more electric vehicles will be able to use this electricity to recharge more sustainably.

However, larger batteries also come at a cost. Lithium mining is a water- and energy-intensive process today, involving the drilling and pumping of brine from salt flats commonly found in dry regions of Bolivia, Argentina and Chile [6]. In addition, production of lithium-ion battery cathodes requires the refining of cobalt, nickel, and manganese in continuously heated kilns at temperatures of over 1000 °C [7]. A large proportion of the CO_2 emissions from a BEV comes from battery production, and in locations where significant amounts of coal are used for electricity generation, the use of a standard mid-sized BEV can produce 50% more CO_2 emissions than a comparable ICE vehicle on a life-cycle basis [8]. Several studies have shown that hybrid electric vehicles and a mixed fleet approach is in fact the fastest way to save the most amount of carbon in a robust and sustainable fashion, without solely relying on any one technology, and making use of existing ICE infrastructure [9–11].

1.2 ICE fundamentals

The ICE has seen an 86% increase in average fuel economy since 1975, resulting in real-world CO_2 emissions per mile being reduced by half [12]. In addition, there

has been a 1000-fold decrease in UHC, soot, NO_X and CO emissions in the past 40 years [13]. This is encouraging, but there remains a significant scope for innovation, which will be discussed following a brief discussion of relevant engine fundamentals in this section.

1.2.1 Operation and thermodynamics

ICEs can be divided into two categories; positive ignition (PI) and compression ignition (CI) engines. PI engines use an external energy source to create a localised area of high temperature that triggers the onset of combustion, with sparks, lasers, and turbulent jets as examples [14]. Spark ignition (SI) is by far the most common form of PI due to the minimal cost and complexity of spark plugs [15]. In general, SI engines ignite gasoline fuel with a spark plug, while CI engines initiate self-ignition of diesel fuel by raising the temperature and pressure in the cylinder. It is useful to begin with the framework of these two categories while discussing engine fundamentals; however, amalgamations exist that aim to combine the benefits of traditional gasoline and diesel engines, which will be discussed in due course.

The operation of SI and CI engines involves the sequential compression and expansion of the working fluid along with heat addition and heat rejection events. As a result, SI and CI engines can be modelled against ideal thermodynamic cycles (known as the Otto and Diesel cycles for SI and CI engine cycles respectively). Analysis of these idealised cycles provides a theoretical upper limit for the efficiency of these engine operations, which is useful in understanding how to maximise the efficiency and minimise the environmental impact of modern engines. Thermodynamic cycles are typically visualised as diagrams of evolving pressure against volume, as shown in Figure 1.1. In the Otto cycle (**1.1a**), process $1 \rightarrow 2$ represents isentropic compression, $2 \rightarrow 3$ is heat addition at constant volume, $3 \rightarrow 4$ is isentropic expansion, and $4 \rightarrow 1$

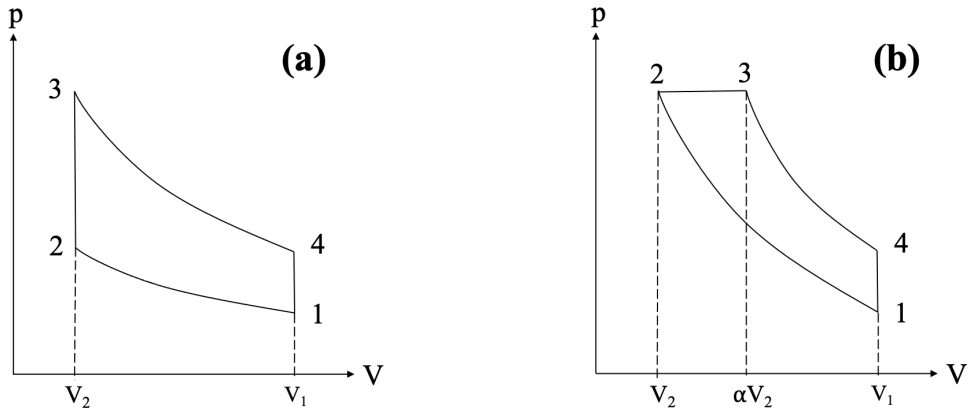

Figure 1.1: Pressure (p) - volume (V) diagrams illustrating **(a)** the ideal Otto cycle and **(b)** the ideal diesel cycle. V_1 and V_2 represent the combustion chamber volume at bottom dead-centre (BDC) and top dead-centre (TDC) respectively.

is heat rejection at constant volume. The Diesel cycle (**1.1b**) is similar but with process $2 \rightarrow 3$ representing heat addition at constant pressure. The diagrams show the compression ratio (r_c) defined as V_1/V_2, and the cut-off ratio α, defined as V_3/V_2. Equations 1.1 and 1.2 show that ideal cycle efficiency can be improved by increasing the r_c and the ratio of specific heat capacities (γ).

$$\eta_{Otto} = 1 - \frac{1}{{r_c}^{\gamma-1}} \tag{1.1}$$

$$\eta_{Diesel} = 1 - \frac{1}{{r_c}^{\gamma-1}} \left[\frac{\alpha^{\gamma} - 1}{\gamma(\alpha - 1)} \right] \tag{1.2}$$

As α is always greater than one, the term in the square brackets of Equation 1.2 is also greater than one. This means that for the same compression ratio, the ideal Diesel cycle efficiency will be less than that of the Otto cycle. However, CI engines are usually more efficient than SI engines for three reasons. Firstly, CI engines can operate at greater compression ratios; SI engines are more limited in order to prevent uncontrolled auto-ignition of the fuel ahead of the spark, a phenomenon known as

engine knock. Secondly, CI engines control engine load by varying the amount of fuel injected into the cylinder, while SI engines have been constrained to running at near-stoichiometric air-fuel ratios for the past few decades [15]. This is due to the success of the three-way catalyst (TWC) in controlling NO_X and CO emissions to within legal limits. The TWC requires near-stoichiometric operation for peak conversion efficiency, so SI engines rely on throttling for load control, which increases pumping losses and reduces efficiency compared to CI engines. Thirdly, the ability to control load with fuel injection means that CI engines operate in globally lean conditions for a wide range of engine speeds and loads, which raises the ideal cycle efficiency by increasing the value of γ in the expansion stroke. Lean fuel-air mixtures are defined as having fuel-air equivalence ratios (ϕ) of less than one, meaning that the ratio of fuel to air in the mixture is less than stoichiometric. The practical lean limit in an SI engine is approximately $\phi = 0.7$, below which the reactivity of the mixture becomes so low that partial burns and misfires become prevalent [16]. On the other hand, modern diesel engines can run reliably at leaner fuel-air ratios of $\phi = 0.6$ [15].

After-treatment system requirements for handling emissions are also different for typical SI and CI engines. Whereas near-stoichiometric SI engines can use the TWC to capture and convert CO and NO_X, in CI engines the excess oxygen due to lean operation works against the reduction of NO_X, causing the TWC to become less effective. Instead, CI engines require more complicated and expensive emissions after-treatment systems including diesel oxidation catalysts (DOCs) for CO and UHC, and lean NO_X traps (LNT) and selective catalyst reduction (SCR) systems to store and reduce NO_X.

Another key difference in the configuration of CI and PI engines is whether or not the fuel is directly injected into the combustion chamber, or premixed with the air and

introduced into the chamber via the intake ports. Direct injection is significanltly more common in CI engines, as the timing of the injection can be used to control the combustion timing, which is particularly important in the absence of a spark. Direct injection engines tend to emit more particulates than port injection engines in both PI and CI variants. In the former case, the air and fuel do not have as much time to mix which creates locally fuel-rich pockets that may not oxidise completely [17]. In particular, ultra-fine particles (below 100 nm in size) are potentially more hazardous as they can penetrate further into the lungs and are more likely to enter into the bloodstream [15]. These particles contribute more significantly to the total number of particulates than the mass, leading to the introduction of particulate number (PN) regulations. Gasoline direct injection (GDI) and diesel direct injection engines therefore require gasoline and diesel particulate filters (GPFs) and (DPFs) respectively for effective particulate control [5]. The complex after-treatment devices required by direct injection systems increases the cost of CI engines, which is compounded by their greater size and weight, needed to cope with the higher peak pressures from greater compression ratios. Smaller and cheaper SI gasoline engines have therefore been the ICE of choice for most light-duty vehicles in recent years.

1.2.2 In-cylinder flows

A major point of interest for modern engine development is the fluid flow through and within the engine cylinder, which is responsible for creating and transporting the air-fuel mixture (also known as the charge) that is subsequently burned. In an engine, air (or pre-mixed air and fuel) enters the combustion chamber via the intake valves. This intake flow is often characterised in two ways. Firstly, the flow is turbulent, and therefore exhibits chaotic and unsteady behaviour. A more in-depth discussion on the nature of turbulence is provided later in §2.2. Secondly, there may be a larger scale coherent or organised motion that is influenced by the geometry of the inlet

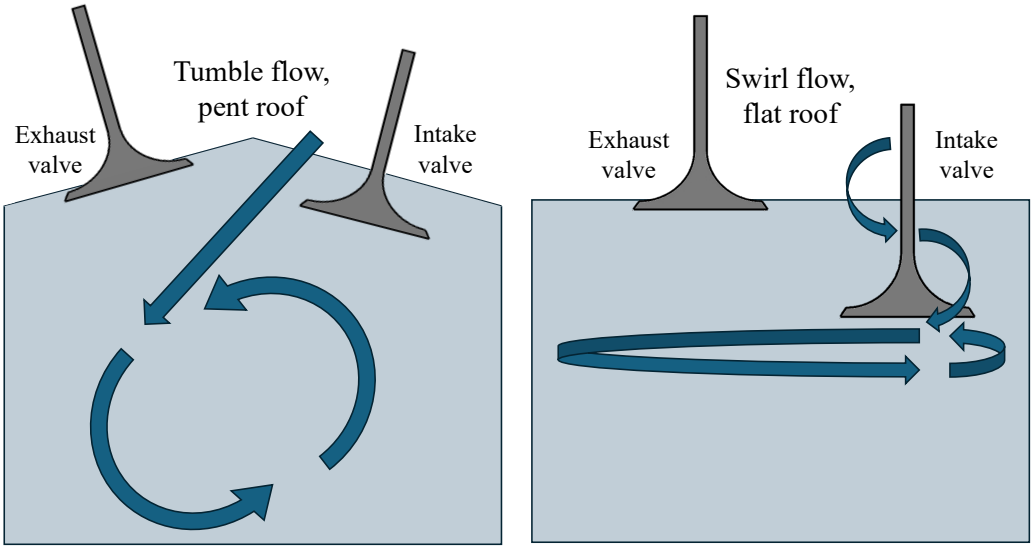

Figure 1.2: Side-view of a tumble flow motion in a pent-roof engine (left), and a swirl motion in a flat roof engine (right).

ports and valves, as well as their opening and closing timings [18, 19]. Large scale motions that have rotational axes about the cylinder axis and orthogonal to it are known as swirl and tumble flows respectivley. These types of flow are illustrated in Figure 1.2.

Controlling the level of turbulence in the engine cylinder, particularly at the point of ignition, is extremely important for facilitating clean and efficient combustion. For example, turbulence can increase the flame's surface area by wrinkling and stretching it, which increases the rate of fuel consumption per unit volume [20]. Turbulence also increases transport of heat and chemical species (i.e. the fuel), and both of theses effects lead to faster burn rates. The faster burn improves the thermodynamic efficiency, as more of the energy release can happen closer to TDC [15]. Furthermore, turbulence can enhance the air-fuel mixing process, which can reduce the emission of harmful unburned and partially burned fuel pockets. Conversely, too much turbulence can over-stretch the flame front, leading to excessive heat losses, combustion instabilities, and ultimately flame quenching. A more detailed discussion of combus-

tion physics is beyond the scope of this thesis, so the interested reader is referred to standard textbooks on the subject [20, 21].

The level of turbulence varies throughout the engine cycle, and can grow or decay as the geometry changes due to the moving piston and valves. During the intake stroke, the turbulence intensity is initially high due to the fast-moving intake jet, but it then declines after intake valve closing as energy is gradually lost due to viscous effects [19]. Therefore, one of the main reasons for introducing organised motion such as tumble and swirl is to persist some of the momentum of the intake jet through until the point of ignition, by virtue of the larger spatial scales being less dissipative in nature [19]. Then, close to TDC, the organised motion can break down into turbulence as it struggles to maintain its form in the reduced space, creating conditions that are conducive to efficient combustion. Swirl flows can also aid in fuel stratification, which is a technique that can be used to improve fuel economy as discussed in the next section.

Tumble and swirl flows are often induced via directed engine geometries. For example, engines with pent-roof designs are very effective at generating tumble flows, as the angled intake ports direct the intake charge down and against the far wall of the engine cylinder, leaving space for the fluid to be pushed around to the back and form a spinning loop [19]. Pent-roof designs are especially common in SI engines, as the extra space also allows for the placement of a central spark plug, minimising the distance that the resulting flame has to travel [15]. However, a significant drawback is that this reduces the geometric compression ratio, and therefore thermodynamic efficiency. Therefore, for CI engines, where the combustion is controlled by the injection and mixing processes instead of a spark, a flat roof is more typical. Rather, swirl flows are more common in CI engines, where air-fuel mixing is a crucial process to control

[15]. Other methods of inducing swirl include variable valve lift (VVL) and variable valve timing (VVT) strategies that keep one valve closed during the intake stroke, using helical intake ports, or adding shrouds (i.e. shaped ridges) around the rim of the intake valves. This thesis handles data from both tumble-dominated pent-roof engines (the Oxford and Darmstadt optical engines, presented in §2.5.1 and 2.5.2) and the flat-roofed TCC−III (§2.5.3.)

1.3 Developing engine technology

Due to the near-stoichiometric constraint for SI engine operation over the past few decades, key efficiency improvements have revolved around decreasing pumping losses due to throttling at part loads. One approach is to use turbocharging to increase the specific power of an engine (also known as boosting) and therefore downsize it. This involves compressing the intake charge, thereby increasing the mass flow rate, engine torque, and subsequently engine power. Intercooling can be used to further increase the density of the compressed gas for higher mass flow rates, and the lower temperatures also reduce NO_X emissions and knocking propensity [15]. Downsized engines reduce vehicle weight, relative friction and don't require as much time and energy to heat up during a cold start [22]. In addition, in the case of SI engines, pumping losses can be reduced by controlling the power output by changing the additional pressure from the turbocharger, lessening the need for throttling [5]. Limitations to boosting and downsizing include an increased knock propensity due to higher pressures and temperatures, maintaining load requirements at low engine speeds with restricted boost flow rates, transient performance due to turbolag, and the re-optimisation requirements for the engine geometries and layouts for extremely small engines [23].

Gasoline direct injection (GDI) is an example of convergent SI/CI technology that has become prevalent in recent years. Evaporative cooling of the fuel lowers in-cylinder temperatures, leading to reduced knocking propensity which facilitates the use of a higher r_c [5, 17]. Although GDI engines can increase efficiency, diesel-like after-treatment systems in the form of GPFs are also required in order to remove particulate emissions due to fuel-wall impingement and incomplete mixing, which increases the cost of the vehicle and can penalise fuel economy [17].

Miller and Atkinson cycles have also been used in SI engines, as alternatives to the typical Otto cycle-like operation. Idealised representations of Miller and Atkinson cycles are plotted alongside the Otto cycle on the pressure-volume diagram shown in Figure 1.3. The idea is to increase the expansion ratio ($r_e = V_{4M}/V_2$ for the Miller

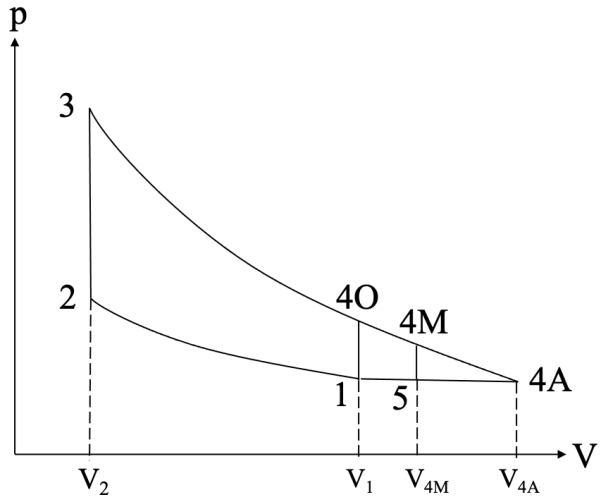

Figure 1.3: Pressure (p) - volume (V) diagram comparing the Otto cycle to the Miller and Atkinson cycles. The Otto cycle follows 1→2→3→4O→1, the Miller cycle 1→2→3→4M→5→1, and the Atkinson cycle 1→2→3→4A→1.

cycle and $r_e = V_{4A}/V_2$ for Atkinson) relative to the compression ratio ($r_c = V_1/V_2$) in order to extract more useful work out of the engine, as the r_c is usually knock-limited in SI engines [15]. The Atkinson cycle has the highest thermal efficiency with the maximum r_e, but the low pressure (shown by point 4A in Figure 1.3) results in larger

exhaust pumping losses, and the longer cylinder body increases engine weight and frictional losses [24]. Real over-expansion cycle engines typically use Miller cycles instead, by altering the timing of the intake and exhaust valves for the constant pressure compression (process 5→1). The timing of the intake valve closure can be made earlier (EIVC) or later (LIVC) than usual. EIVC is slightly more efficient than LIVC, as LIVC does extra compression work due to backflow in the intake manifold [25]. However, this compression work also helps to retain higher in-cylinder temperatures and pressures. LIVC therefore does not suffer from as large power losses, and it is commonly preferred in modern vehicles [24, 26]. This has numerous benefits, namely power control without throttling, lower temperatures resulting in less NO_X formation, and enhanced knock resistance [5, 24]. The main drawbacks are the requirement for expensive variable valve timing (VVT) technology, and limited power outputs due to the restricted intake in the cylinder [24, 27]. Miller cycle engines commonly use large turbochargers to alleviate the power density problem, which adds to the cost and complexity of the vehicle [28].

Gasoline compression ignition (GCI) may be able to provide diesel-like efficiency without suffering from the same emissions penalties. The higher volatility and longer ignition delay of gasoline facilitates better mixing of the fuel and air to reduce particulate emissions [29]. However, some control over the start of ignition is lost in premixed CI engines, so spark-assisted compression ignition (SACI) has been developed to remedy this. The 2019 Mazda Skyactiv X is the first example of a SACI engine in a production vehicle, initiating combustion with a spark but making use of high pressures and temperatures to trigger subsequent autoignition through a lean fuel-air mixture [30]. A 10% fuel economy improvement over a comparable Atkinson cycle engine was reached, which increased to 30% when coupled with a mild hybrid system [10]. However, large pressure rise rates mean that the engine has to resort to

conventional stoichiometric SI operation at high loads, so further work will look to implementing lean combustion across the entire engine map.

Fuel stratification via multiple direct injections can also be used to stabilise flame initiation in ultra-lean engines, supplementing a weak premixed background charge with a small pilot injection just before the spark. Sandia National Laboratory used this method to control end-gas auto-ignition for rapid combustion through an ultra-lean mixture [31]. Stable combustion at global $\phi = 0.45$ was achieved, but an excessively advanced spark timing was required which generated significant NO_X emissions and heat losses. Potential solutions could be increasing the energy density of the mixture with boosting or investigating a range of different injection strategies.

Turbulent jet ignition (TJI) makes use of a spark plug located inside a small pre-chamber, which is connected to the main chamber by an array of small orifices. When the spark ignites the mixture in the pre-chamber, the increase in pressure forces the flames through the orifices into the main chamber, forming several turbulent jets that ignite the rest of the air and fuel in the main chamber [32]. This simultaneously increases turbulence and creates several ignition spots in the main chamber, facilitating the use of ultra-lean combustion. Passive configurations rely on gas exchange with the main chamber to supply the pre-chamber with fuel, whereas active systems utilise an additional fuel injector inside the pre-chamber. Active systems extend the lean limit further than passive ones by creating stronger flames in the pre-chamber, but add to the complexity of the system. Combustion stability at low loads can be a problem as the lower pressures inhibit gas exchange with the pre-chamber, and a lack of oxygen in the pre-chamber results in more frequent misfires [10, 33]. However, this issue could potentially be made less significant by implementing TJI into a hybrid powertrain, as the engine can be set to run at its most efficient operating point with

the battery meeting the more extreme high and low load demands, as demonstrated by Brannys *et al.* [34]. Other challenges are the hardware modifications required by TJI, and the NO_X emissions that cannot be handled with a TWC.

Cylinder deactivation (CDA) involves motoring individual engine cylinders in periods of low torque demand. This can be achieved with open or closed intake valves; closing the valves requires a more complex control system, but it is a more efficient strategy due to the reduced pumping losses [35]. By deactivating some cylinders, in order to meet the torque demand of the engine, each active cylinder needs to run at a higher specific power. This reduces the need for throttling and improves the conversion efficiency of the TWC by limiting the amount of oxygen in the exhaust [35, 36]. However, additional components in the valve train increase the number of contact points, which increases friction, reduces the rigidity of the valve train and makes the system more susceptible to resonant vibrations [37].

Table 1.1: Summary of potential CO_2 reductions associated with different upcoming engine technologies. Adapted from [10].

Engine Technology	**CO_2 reduction**
Baseline: GDI, turbo, stoich.	0
Atkinson cycle (+ VVT)	3-5%
Dynamic CDA + mild hybrid or Miller	10-15%
Stratified lean-burn GDI	10-20%
Spark assisted GCI	10%
Turbulent jet ignition	15-20%

The potential CO_2 savings associated with the engine technologies discussed here are reported in Table 1.1, with figures taken from Joshi [10]. There is therefore substantial scope for continuing the trend of reduced CO_2 emissions from ICEs in the short to medium term, with the added possibility of adopting carbon-neutral fuels for further reductions in the longer term.

1.4 Simulating engines

Whichever technological pathways are ultimately chosen, methods that can accelerate the development and adoption of such technologies are of significant value. Computer simulations are a particularly relevant example in modern technology development, as they can be a much cheaper and faster way of testing multiple iterations of a design without having to manufacture a prototype in each case [38, 39]. In addition, for ICEs, simulations can provide insight into the physics of in-cylinder processes which can be difficult to observe and measure directly [15]. This information can be used to optimise performance, diagnose problems, and provide explanations for observed experimental results. In particular, computational fluid dynamics (CFD) is a form of computer simulation that can analyse turbulent flows by solving a system of governing equations based on physical conservation laws. For a three-dimensional flow these governing equations comprise of five differential equations, representing mass continuity, conservation of momentum in three directions, and conservation of energy (see Chapter 2).

It is possible to compute the solutions to each of the governing equations at every point in the domain down to the smallest scales in an approach known as direct numerical simulation (DNS). However, DNS is limited to relatively simple flows for the foreseeable future despite progress in modern supercomputer performance, as the runtime for DNS increases rapidly with the Reynolds number [40, 41]. For a more detailed explanation and derivation of the relationship between computational cost and Reynolds number, see Section 2.2.1. As a result of these resource constraints, DNS studies of full-scale ICEs are currently extremely rare in the literature and limited to simplified geometries and operating conditions such as the ETH Zurich valve/piston assembly, which consists of a flat-top cylinder head with a fixed, axis-centered valve and a moveable piston [42, 43].

Instead, alternative CFD frameworks that model the effects of turbulence are widely used to study engine phenomena, such as Reynolds-averaged Navier-Stokes (RANS) and large-eddy simulations (LES). These approaches will be further discussed in §2.2. Although less computationally intensive, turbulence models are accompanied by a number of simplifications which reduce the accuracy of the CFD simulation (see §2.2). Consequently, in order to increase the reliability of such simulations, experimental data are often needed to validate the results from a RANS or LES simulation [44, 45]. Such validation processes are deemed to be essential in standard CFD textbooks [46, 47], with Richard Hamming's famous words "The purpose of computing is insight, not numbers ... it is not a step to be taken in isolation from reality" still ringing true today; the core value of numerical results is in their interpretation by the user, and how the results interact with reality [48]. However, characteristic differences between data from experiments and simulations mean that the validation process is not always straightforward. This concept is discussed along with applications to engine in-cylinder flows in the next section.

1.5 Validation processes

Validation is defined by the American Institute of Aeronautics and Astronautics (AIAA) as "the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model" [49]. Of particular note is the idea that whether or not a model has been validated depends on the intended use of the model, which is subjective. In addition, it is understood that validation applies to individual outcomes of a model, not the model in its entirety; the latter case refers to the process of model verification, which is out of scope for this thesis (see [50, 51] for more details). Validation processes therefore need to evolve in order to reflect changes in how the models are used.

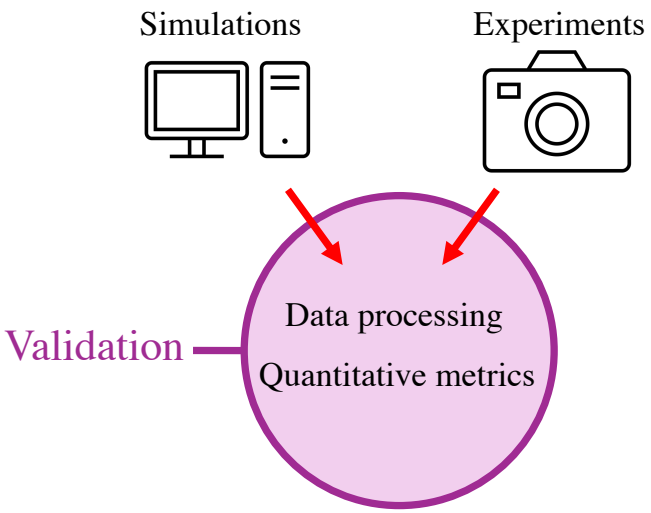

Figure 1.4: Schematic of a validation process involving the comparison of data from simulations and experiments. Data processing methods and quantitative metrics are the two key components. Data processing methods are used to reconcile key differences between the two data sources, identify important features of interest, and form validation targets. Quantitative metrics are used provide an objective analysis on the accuracy and suitability of the simulation with respect to the experimental data.

As an example, for a parametric investigation of different piston shapes on global quantities such as resultant engine-out emissions, it may not be necessary to validate the simulated in-cylinder flow fields, and a comparison of pressure traces may be sufficient [52, 53]. However, if a deeper understanding of the local phenomena that contribute to differences in the outcome is desired, a more detailed validation process of the flow fields may be required in order to extend the inferences that can be made from the simulation results [54].

A validation process between a model and a source of truth often consists of two critical components: some data processing, to transform the raw data into a useful format, and a metric, to objectively measure the closeness of the match [49]. This relationship is emphasised in Figure 1.4; the literature in this section is reviewed through this lens, and this thesis makes contributions to both components.

Within engine research, it was initially common practice to quantitatively validate global parameters such as pressure and engine-out emissions, while local quantities were often limited to qualitative comparisons [55, 56]. Subsequent works developed more rigorous validation methodologies by quantifying differences between large scale flow features such as vortex centres [57], velocities along lines of interest in the cylinder [58], and the shape and size of the intake air jet [59, 60]. Further progress in CFD validation was made following the introduction of a quantitative metric: the relevance index (RI) by Liu and Haworth [61], which faciliated objective comparisons of overall vector alignment between measured and simulated velocity fields. Similarly, the magnitude index (MI) was introduced into engine research by Hu *et al.* [62] to quantify the similarity of vector magnitudes. More details on these vector-based indices are provided in Section 4.2.1.

The vector-based metrics enabled more objective comparisons to be made between local flow features from simulations and experiments, and were used extensively in the development of the proper orthogonal decomposition (POD) technique for engine flow field analyses [63, 64]. POD is a linear dimensionality reduction technique, used for data processing in this case, that is discussed in detail in § 3.3. These early works with POD demonstrate how a new metric can accelerate the development of a data processing method; for example, the RI was used to correlate high energy-containing POD modes with coherent structures in the flow, and determine thresholding criteria for the number of POD modes to retain in turbulent flow reconstructions [65]. Such studies exemplify the potential that more objective validation processes can have for understanding new techniques, facilitating their widespread usage with increased confidence, and providing additional insight into turbulent flow phenomena. Current engine CFD validation processes now typically follow a workflow of increasing complexity, beginning with an evaluation of global quantities, followed by more detailed

considerations of smaller-scale physical processes with the use of vector-based metrics [44, 45, 66].

Data processing methods have had an explosion in capability in recent years, largely driven by data-driven approaches. Fuelled by drastic advancements in computer hardware and increased access to large datasets [67], data-driven discovery has been referred to as a fourth paradigm of scientific discovery, after empirical experimentation, analytical derivation, and computational prediction [68, 69]. Within this framework, the various classes of machine learning (ML) algorithms are of particular note, which are flexible methods that can learn patterns from data without the need for explicit information about the problem at hand [70]. Data-driven discovery is a very flexible paradigm, capable of handling very large datasets with many competing internally-correlated variables, and uncovering hidden and complex patterns and relationships in data. They are especially useful in situations for which developing a governing equation is untenable; for example, the validation of a CFD model is inherently specific to individual outcomes of that model, and its intended usage by the practitioner. The derivation of universal governing equations is clealy unsuitable for such a task.

Recent review papers cover the transformative potential of ML for experimental fluid mechanics [71] and CFD [72] alike. Such data-driven tools have been shown to be similiarly capable in the field of engine research [73, 74]. Indeed, Zhao and Hung [75] include ML as a fundamental pillar of modern engine flow research, alongside CFD and laser diagnostics methods. However, it should also be noted that the advantages of data-driven methods can come at the expense of interpretability; in addition, they are typically not universally generalisable in the same way that a physical equation would be [76, 77], and like other forms of numerical modelling, extensive care should be taken in defining the scope of a data-driven model as well as how the model should

be used outside of that scope.

Strong use cases for data-driven methods within engine research have been identified as flow field data processing, prediction of dynamics, and control [73, 74]. This thesis is concerned solely with data-driven methods for flow field data processing, as this is the most relevant application for engine CFD validation. Furthermore, the unique advantages and disadvantages of data-driven methods mean that scientific investigations can often benefit from integrating them with other paradigms of scientific discovery. The engine CFD validation process is an excellent example of this, where data processing techniques sit at the intersection of simulations and experiments, enabling objective comparisons to be made between the two, as illustrated in Figure 1.4. There is therefore significant opportunity to take advantage of the recent developments in data-driven methods in order to improve data comparison within engine CFD validation processes.

1.5.1 Current challenges for validation processes

Two key challenges arising from current engine CFD validation processes have been identified, corresponding to two key steps in the validation process, which are listed below.

- The creation of validation targets from data.
- The enhancement of validation data.

Each of these topics are described in this section, beginning with the creation of validation targets from data.

Creation of validation targets

As will be discussed in §3.2, the ability to collapse a large dataset onto a subset that can serve as a more manageable validation target is useful for facilitating interpretable and objective validation processes. In the case of experimental particle image velocimetry (PIV) or large eddy simulation (LES) data, this is often achieved with the ensemble mean (EM). Naturally, the EM calculation for vector fields can lead to the cancellation of opposite vectors, which may diminish the overall vector magnitudes in the EM field, as illustrated in Figure 1.5. This vector-cancelling feature of the EM is prevalent in the study of the highly turbulent airflow through internal combustion engines (ICEs). For example, when taking velocity measurements at the same point across consecutive engine cycles (rotations of the crank shaft), the airflow might point in different directions due to reasons such as the stochastic effects of turbulence and physical oscillations of the engine assembly [78]. This makes the dataset susceptible to the diminished magnitudes effect when attempting to take an average. Therefore, care is needed when attempting to use the EM to interpret the velocity magnitude information in a dataset, particularly when there are high levels of variation in the flow.

Accurate validation of the simulated airflow through ICEs is of particular interest as the intake airflow is a major source of kinetic energy in the engine cylinder [79, 80], significantly influencing subsequent cycle-to-cycle variations (CCVs) [81, 82], combustion quality [83, 84], wall heat transfer [85], and ultimately engine efficiency [15]. In particular, Van Dam and Rutland [86] showed that variability in engine flows is primarily due to variable velocity magnitudes rather than variable velocity directions. It is therefore of utmost importance to explore methodologies that can produce simple representations of vector fields with magnitudes that accurately reflect the original data.

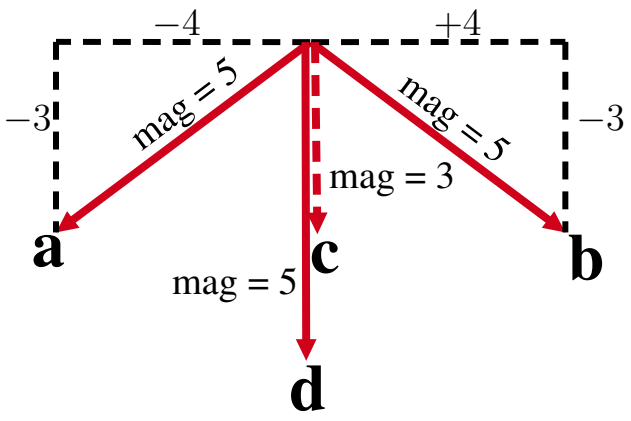

Figure 1.5: Diagram showing two vectors **a** and **b** with equal magnitudes, pointing in different directions. An arithmetic average of **a** and **b** gives the vector **c**, which has a diminished magnitude due to the cancellation of horizontal velocity components. The vector **d** represents a potentially more desirable form of the average, which can be achieved with dimensionality reduction methods. In the diagram, 'mag' refers to the vector magnitude.

Prior studies have shown the deficiencies of simple averaging processes when attempting to create reliable representations of highly variational engine flows [54, 64, 87]. This is particularly true for cases where strong CCVs cause each of the individual snapshots to appear differently from the overall ensemble average. A clarification is warranted here regarding the definition of CCVs, and what causes the changes in flow features inside the cylinder between subsequent cycles. Lumley [19] notes that 'the flow is deterministic, and the flow at TDC is entirely determined from (and very sensitive to) the flow at the inlet valve closing. Since this is entirely determined by the flow in the inlet manifold during filling, it must be this flow that is responsible for the cycle-to-cycle variability', and, later 'relatively small differences at inlet valve closing attributable to the manifold turbulence are enough to cause considerable variability at TDC, due to the extreme sensitivity of the turbulence to initial conditions'. Finally, he writes that 'since the cycle-to-cycle variation, and the small scale turbulence in the cylinder appear to be physically the same, differing only in scale, it probably makes sense to lump them together, because the cycle-to-cycle variations are also responsible for transport in the cylinder'.

The turbulent fluctuations and larger-scale CCVs can be 'lumped together' and isolated by subtracting the mean from any individual measurement. This approach has been widely used in the literature [64, 65, 88], and it is also implemented in this thesis, as the present work is more concerned with the effect that variations can have on the average rather than defining the sources of the variations themselves. However, it is worth noting that although the flow is indeed deterministic, Lumley's conclusion that the CCVs and turbulence appear to be physically the same seems to neglect the impact that physical vibrations of the apparatus can have on the level of flow field variability. For example, the General Motors − University of Michigan group found a correlation between the lateral position of the intake valve and the resulting flow patterns inside an optical engine experiencing valve ringing (lateral oscillations), which caused variable mass flux through the intake valve [89]. This of course influences the intitial and boundary conditions of the flow inside the cylinder, is not determined by prior turbulence in the intake manifold, and likely exhibits different physcial characteristics to true turbulence (i.e. a lack of stochasticity). In any case, the complex relationship between turbulence and CCVs is still very much an open and active research question, and currently evades a precise definition [90–92].

The existence of CCVs in flow fields has motivated the use of modal decomposition methods, which are a branch of dimensionality reduction techniques [93]. Modal decomposition methods decompose complex datasets into a set of modes, which are linear combinations of the original data that can isolate the effects of specific patterns in the data. More detail on these techniques is provided in §3.3. The proper orthogonal decomposition (POD) is the most widespread modal decomposition method used in engine research [88, 94].

POD is a technique that can decompose a high-dimensional dataset into a hierarchy

of low-dimensional structures known as modes, and was first introduced to the field of fluid dynamics by Lumley in 1967 [95]. The low-rank feature extraction capabilities of the POD technique have been demonstrated widely in engine contexts; for a flow where the ensemble mean was a poor representation of the individual cycles, Chen et al. [64] showed that POD was able to separate and quantify the cyclic variability of the average flow structure and the turbulence energies. Liu et al. [96] investigated LES and PIV flow data from a single-cylinder, two-valve piston engine using various implementations of POD, including both phase-dependent and phase-invariant versions. The authors found that the reconstruction of complex in-cylinder flow fields was possible with just the first few POD modes. POD has also been used to gain further insights into CCV and the energy cascade in ICEs via the triple [65] and quadruple [97] decomposition methods. Recently, Wu et al. [98] used triple POD to investigate the interaction of dominant, coherent, and incoherent flow structures in a combustion chamber under motored conditions and were able to correlate the significant CCVs with the coherent flow structures identified by triple POD.

However, in its standard form, POD modes are only spatially coherent, so each spatial mode may consist of structures that are characterised by a mixture of different temporal dynamics [93]. A more appropriate tool for analysing spatial and temporal dynamics together is the dynamic mode decomposition (DMD), developed by Schmid [99], and described in detail in §3.3.5. DMD produces spatial modes that each oscillate at a single frequency with an associated growth or decay rate, thus capturing the changes in spatial modes over time. DMD has been broadly applied to the study of flows [100–102], but also in areas as diverse as epidemiology, medical imaging, and robotics [103].

Studies of engine flows using DMD are relatively scarce in the literature, possibly due

to the difficulty in relating spatial structures to physical time in cyclostationary processes. For example, in order to fully interpret the results from a phase-dependent* application of DMD, it is likely that one needs to assume that there is a link between flow fields at the same phase across consecutive cycles. However, a number of complex processes occur in the time it takes for an engine to return to the same phase, such as expansion, compression, and a replacement of the working fluid via the exhaust and fresh intake [15]. The degree to which the flow at a certain phase in one cycle is correlated with the flow at the same phase in the next cycle is therefore unclear. By assuming that the engine operates as a continuous process, Qin *et al.* [97] implemented phase-dependent DMD to investigate an engine flow, but did not explore the spatial structures of the DMD modes or relate them to physical features of the flow. An alternative phase-dependent implementation was proposed by Liu *et al.* [104], who treated each snapshot as an independent measurement. Therefore, they conducted a random permutation and averaging process to uncover the statistically global DMD features from cycle to cycle. Applications of DMD to engine flows are perhaps more naturally suited to phase-invariant approaches, where the physical links between consecutive snapshots are more apparent. Liu *et al.* [105] used phase-invariant DMD to investigate the dynamic properties of CCV structures. However, in order to bring physical meaning to the DMD modes, they required a combinatorial search to match the DMD modes to phase-invariant POD modes.

For the purpose of producing a validation target from a vector field dataset in a similar manner to how the vector $\mathbf{d}$ can represent the vectors $\mathbf{a}$ and $\mathbf{b}$ in Figure 1.5, Shen *et al.* [54] compared phase-dependent POD to ensemble averaging and speed-based averaging. They concluded that among these options, POD-reconstructed flow fields provided the most suitable representations of the flow in the considered scenario.

*The terms phase-dependent and phase-invariant are discussed in §3.2.

Fang *et al.* [106] highlighted the subjectivity in determining the threshold number for the number of POD modes retained in the reconstructions, and proposed the use of kernel principal component analysis (KPCA) as a more objective solution. However, neither of these solutions proved capable of reliably collapsing a dataset onto a single validation target, which has remained an open question, and is addressed in Chapters 4 and 5 with the use of a DMD variant known as the sparsity-promoting dynamic mode decomposition (SPDMD).

Enhancement of validation data

In situations where PIV images contain no data from any engine cycles at a particular location, neither the EM nor any of the modal decomposition techniques can be calculated completely. Such situations can arise when conducting PIV experiments due to challenges such as shadowing (occlusions due to walls or other components), laser alignment issues, irregular seeding density of the tracer particles, background reflections and light scatter, and strong out-of-plane motion for 2D measurements [107, 108]. Attempting to rectify gappy PIV data through experimental reruns may be costly or in some cases impossible, and as design work becomes increasingly digitalised, experimental data will need to be compared to or assimilated with typically clean simulation data for validation purposes [47, 109] or in the construction of digital twins [38, 110, 111].

The development of numerical methods that can fill gaps in spatio-temporal turbulent flow data, a task also known as inpainting in the image processing community [112, 113], has a history spanning several decades. Of particular note is the family of methods stemming from what came to be known as the gappy proper orthogonal decomposition (GPOD), introduced by Everson and Sirovich in 1995 [114]. These methods employ the POD to identify dominant flow structures in a dataset, which

are used to inform the velocity predictions inside the gaps. The predictions are updated iteratively as the number of POD modes (principal components) considered for the reconstruction is incremented. More details on the method can be found in § 3.3.3.

GPOD became an industry-norm in the field of turbulent flow diagnostics, and received several updates and improvements over the past few years [115–118]. One reason for GPOD's popularity in the fluid mechanics community is due to its ability to out-perform state-of-the-art interpolation schemes [117]. Another reason is due to the focus that POD methods place on dominant low-rank features, which may be analogous to the concept of coherent structures in turbulent flows [93, 119]. Therefore, results from GPOD retain a degree of physical explainability. In addition, as fluid flow data are often negatively affected by noise, outliers, and potentially less relevant small-scale turbulent structures [65, 120], high levels of reconstruction accuracy can be achieved by mainly focusing on these low-rank structures [107, 117].

However, it should be noted that the singular value decomposition (SVD), integral to POD-based techniques, utilises linear combinations of the original data in order to find new bases with which to represent the data. Therefore, techniques that rely on the SVD such as POD and DMD are unable to model more complex non-linear interactions between the variables [121]. As turbulence is inherently non-linear, the use of non-linear models has the potential to further enhance flow data post-processing capabilities and take on more challenging problems within CFD validation pipelines. An autoencoder (AE) neural network is a type of dimensionality reduction method that may be considered to be a non-linear generalisation of the SVD, due to the use of non-linear activation functions [121]. Autoencoder neural networks have been widely used in turbulent flow applications, partly because the dimensionality reduction through

the latent space maintains the focus on dominant flow structures [122, 123]. Convolutional neural networks (CNNs) are also widely used due to their ability to utilise the local spatial relationships inherent in turbulent flow data on grids [124–126]. In particular, the UNet architecture [127] has demonstrated success in a variety of tasks including flow field prediction and super resolution [75, 128–131].

Regarding alternative architectures, generative adversarial networks (GANs) have been used successfully in preserving multi-scale statistics of turbulent flows for super-resolution [132, 133] and inpainting [134]. Physics-informed neural networks (PINNs) [135] also show promise for creating more generalizable ML models, especially for laminar and fully 3D flows [136–138]. However, the suitability of PINNs for 2D PIV data is currently unclear, as the only data available are two velocity components along a 2D slice of a 3D system, which stretches the validity of the conservation equations. In addition, the spatially correlated noise introduced by the cross-correlation algorithm in the PIV process has been shown to significantly degrade the results from a PINN [139].

In the literature on inpainting for turbulent flows, there is a noticeable lack of discussion on how artificial gaps should be created for training and testing ML models. Common methods of adding gaps to clean data include random noise [140, 141], clustered dropouts [116, 117], and block gaps [75, 118, 142]. Rectifying random noise and clustered dropouts is often an easier task for ML models due to the large amount of spatially local information that remains available [143]. Conversely, due to the larger gap sizes, block gaps can be more challenging to handle. However, studies to date have only considered blocks of standard shapes and fixed orientations, which can be unrealistic and of limited use in practical scenarios where complex geometries can obscure the field of view in any number of ways.

There are large number of inpainting models that are available to choose from, and it is not straightforwards to determine which scenarios will cause one model to perform better than another one and why. In addition, there is a lack of clarity in how different gap-handling approaches affect the results of an inpainting model, and these points motivate the need for an objective benchmark to be established. Benchmarks typically consist of an open-source dataset, as well as a well-defined task that can be used to objectively compare model performances, and they can be essential for developing numerical methods. For example, the ImageNet Large Scale Visual Recognition Challenge benchmark is often credited with catalysing the deep learning explosion, having facilitated the development of the famous AlexNet model [144, 145].

Within the realm of turbulent flow research, several large flow physics datasets exist, including the Johns Hopkins Turbulence Database [146], BLASTNet database [147], and the turbulence data from McConkey *et al.* [148]. However, each of these datasets represent idealised flows over small domain sizes, which do not reflect the complex geometries and operating environments associated with physical machinery. Other databases address more practical geometries and domain sizes, such as the AirfRANS dataset for airfoil shape optimisation [149], and the Cambridge-Sandia burner for a variety of swirling stratified flows [150]. Regarding engine-specific flows, PIV datasets have been published by the Engine Combustion Network (ECN) [151] and the General Motors University of Michigan Automotive Cooperative Research Laboratory [152]. These datasets need to be paired with a benchmark task in order to objectively evaluate inpainting models, and this is the focus of the work presented in Chapter 6.

1.6 Thesis aim and structure

The aim of this thesis is to develop advanced numerical methods that can improve validation processes for engine CFD models against PIV data. In particular, the following research questions are addressed.

1. **How can an objective validation target be created from an ensemble of PIV data?**

   Specifically, previous methods of validation target creation have not been fully representative of high-variation PIV datasets due to the diminished magnitudes problem. As outlined in the literature review, there is potential for advanced numerical methods to overcome this issue. This is investigated in Chapter 4 of this thesis, and publication 1 (as listed in the next section).

2. **How can an ensemble of CFD data be objectively compared to an ensemble of PIV data?**

   Here, the methodologies developed in question 1 are extended towards the validation of an ensemble of LES flow fields against a PIV dataset. This work is presented in Chapter 5 of this thesis, and publication 2.

3. **To what extent can PIV data be enhanced by using numerical methods to repair large gaps inside the images?**

   As an essential first step towards answering this question, a benchmark is established that can objectively evaluate the performance of various inpainting methods. The methodologies developed in questions 1 and 2 are used to assess the practical utility and relevance of the predictions made by the inpainting models investigated. This work constitutes Chapter 6 of this thesis, and publication 3.

The remainder of the thesis is therefore structured as follows. Chapter 2 introduces the different datasets used in the analysis and discusses their key characteristics. Chapter 3 presents the numerical methods used to create the results and findings in this thesis, and discusses their usage with respect to the aims of this thesis. The diminished magnitudes problem for constructing validation targets is introduced in Chapter 4, along with the proposed SPDMD methodology for comparing a Reynolds-average Navier-Stokes dataset to a particle image velocimetry (PIV) ensemble. The methodology is extended for the validation of large eddy simulations (LES) in Chapter 5. The flow reconstruction benchmark is presented in Chapter 6, along with the analysis of different model performances. Finally, the overall conclusions and recommendations are provided in Chapter 7.

1.7 List of publications

Chapters are 4−6 comprised of research that has been previously published. A list of publications completed during the DPhil is provided here.

1. **S. Baker**, X. Fang, L. Shen, C. Willman, J. Fernandes, F. Leach, M. Davy, "Dynamic Mode Decomposition for the Comparison of Engine In-Cylinder Flow Fields from Particle Image Velocimetry (PIV) and Reynolds-Averaged Navier–Stokes (RANS) Simulations", Flow, Turbulence and Combustion 1-26, 2023, doi.org/10.1007/s10494-023-00424-3.

2. **S. J. Baker**, X. H. Fang, A. Barbato, S. Breda, M. Magnani, S. Fontanesi, F. C. P. Leach, and M. H. Davy, "Extracting dominant structures from a cyclic engine flow with dimensionality reduction", Physics of Fluids 36 (2), 2024, doi.org/10.1063/5.0189368.

3. **S. J. Baker**, M. A. Hobley, I. Scherl, X. H. Fang, F. C. P. Leach, and M. H. Davy, "Machine learning benchmark for flow reconstruction in the TCC−III optical engine", International Journal of Engine Research 14680874251330354, 2025, doi.org/10.1177/146808742513303.

4. N. Sekularac, X. Fang, V. Shankar, **S. Baker**, F. Leach and M. Davy, Development of a Laminar Burning Velocity Empirical Correlation for Combustion of iso-octane/ethanol Blends in Air, Fuel 307, 2022, doi.org/10.1016/j.fuel.2021.121880.

Chapter 2

Data sources

2.1 Overview

This chapter describes the methods used to obtain the results presented in this thesis. It begins with a description of the CFD and experimental methods used for obtaining the quantitative velocity data describing turbulent flows. The latter section then introduces the specific datasets used in this thesis, comprising of two CFD and three PIV datasets.

2.2 Turbulence modelling

2.2.1 Introduction

Flows in ICEs are highly turbulent, which is defined as when the Reynolds number (the ratio of inertial forces to viscous forces, Re) exceeds a critical value of approximately 4000, causing the fluid motion to become unstable [40]. Rotational structures

with a wide range of time and length scales known as turbulent eddies are formed and the flow is subject to random fluctuations. The turbulent eddies increase local gradients of fluid properties due to convective transport, which increases rates of diffusion and enables rapid combustion. Many combustion applications, including the ICE, therefore rely on turbulent flows to achieve high volumetric power outputs [20]. As an engine-relevant example, an air flow at room temperature with density 1.2 kg/m^3 and dynamic viscosity 1.8 e$-$5 Pa.s flowing through an intake valve with diameter 0.03 m at 30 m/s gives Re $\approx$ 60000 (see Equation 2.1).

Of particular importance to turbulence modelling is the concept of the energy cascade, proposed by Richardson in 1922 [153] and developed by Kolmogorov in the early 1940s [154]. In this concept, kinetic energy is introduced into the turbulence at the largest scales, through production mechanisms such as the piston and intake valve dynamics in an ICE, and transferred onto progressively smaller scales until eventual dissipation at the smallest scales. This is occurs because the eddy structures at larger length scales are unstable and prone to breaking up. This is illustrated in the Reynolds number equation, where larger length scales correlate to a larger ratio of inertial to viscous forces and therefore greater instabilities:

$$\mathrm{Re} = \frac{\rho u \ell}{\mu} \tag{2.1}$$

where ρ is the density, u is the velocity, ℓ is the length scale, and μ is the dynamic viscosity. This energy transfer process continues with reducing ℓ until the Re is sufficiently small for the kinematic viscosity to become effective in dissipating the energy; these smallest scales are known as Kolmogorov scales. Crucially, this sequential chain of events implies that the rate of dissipation, ϵ, may be determined from transfer of energy from the largest eddies. This process is illustrated in Figure 2.1, and poetically

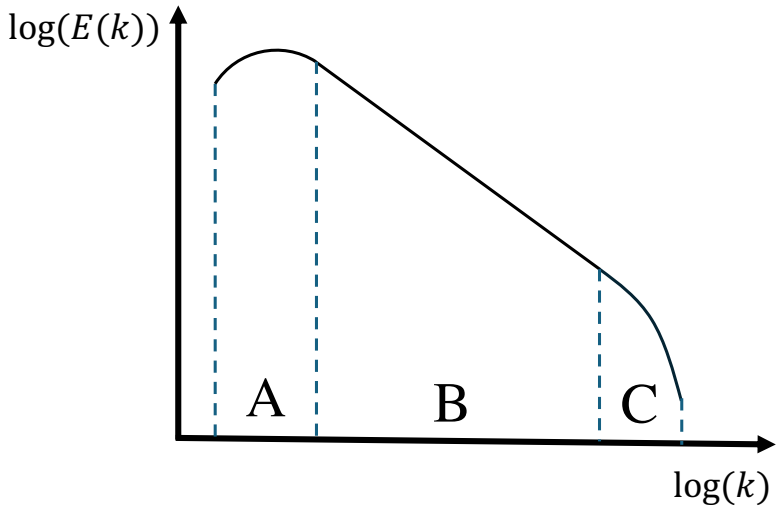

Figure 2.1: A diagram of the energy cascade, showing the kinetic energy E as a function of the wavenumber k. Three distinct regions are labelled, where A is the integral range where energy is produced, B is the inertial subrange where energy is transferred at an approximately constant rate, and C is the disspation range.

put by Richardson [153]:

> Big whorls have little whorls,
>
> Which feed on their velocity;
>
> And little whorls have lesser whorls,
>
> And so on to viscosity.

As mentioned in §1.4, models for turbulence are required because directly calculating all relevant quantities over large space and time-scales, in an approach known as direct numerical simulation (DNS), quickly becomes intractable. Following Pope [40], the smallest turbulence scales (Kolmogorov scales) are given by $\eta \sim \left(\frac{\nu^3}{\epsilon}\right)^{1/4}$ where ν is the kinematic viscosity (units m^2/s) and ϵ is the rate of turbulent energy dissipation (units of kinetic energy per unit mass per second, or m^2/s^3). Due to the energy cascade, ϵ can be related to the integral length scale L and the root mean square velocity u'

by $\epsilon \sim \frac{u'^3}{L}$. Substituting for the Reynolds number $Re = \frac{u'L}{\nu}$ gives $\eta \sim \left(\frac{L^4}{Re^3}\right)^{1/4} = LRe^{-3/4}$. The number of grid points needed to cover all spatial scales is given by the ratio of the largest and smallest volumes, so that $N \sim \left(\frac{L}{\eta}\right)^3 = Re^{9/4}$. The total computational time t_{total} is proportionate to the number of grid points multiplied by the number of timesteps, $N \times T$. The minimum timestep Δt is constrained by the Courant–Friedrichs–Lewy (CFL) condition $\Delta t \sim \frac{\eta}{u'}$ * . Therefore, $\Delta t \sim \frac{L}{u'}Re^{-3/4}$. Finally, as the number of timesteps is inversely related to the timestep size, we have $T \sim \frac{1}{\Delta t} \sim Re^{3/4}$, and the overall runtime scales with $t_{total} \sim Re^{9/4} \times Re^{3/4} = Re^3$. This rapid scaling of computational time with Reynolds number limits the use of DNS to simpler flow set-ups for the foreseeable future, requiring turbulence effects to be modelled instead of calculated explicitly [40, 41].

2.2.2 RANS and LES overview

There are two main classes of numerical methods for modelling the effects of turbulence: Reynolds-averaged Navier-Stokes (RANS), and large eddy simulations (LES). RANS approaches calculate the effects of turbulence on the mean flow using Reynolds- or Favre-averaging, which are procedures that will be discussed in §2.2.3. The focus on average quantities means that the stochastic nature of turbulence cannot be captured explicitly, and turbulence effects must be modelled. However, RANS is used extensively in industry as it is the least computationally expensive method while still being capable of producing accurate results for global quantities [20, 156].

On the other hand, LES is an approach that can provide more detailed results than RANS at additional computational expense. The LES method uses spatial-filtering

*The CFL condition promotes numerical stability in the solution of partial differential equations by ensuring that relevant information should not travel across more than one grid cell per timestep, given by $\frac{u_{max}\Delta t}{\Delta x} \leqq C$, where C is a constant typically set to 1 or below [155]. For the Kolmogorov scales, $\Delta x \sim \eta$, giving $\Delta t \sim \frac{\eta}{u'}$.

to calculate large scales of turbulence explicitly, but still uses sub-grid scale (SGS) models at small turbulence scales. The theory behind this filtering of scales has its roots in the Kolmogorov hypotheses [154], which assume that small-scale eddies are isotropic and therefore have invariant statistics under translations and rotations of the coordinate system. As a result, the average of the fluctuating properties comes to zero, making the small-scale eddies easier to model. Large eddies are still calculated explicitly as they directly interact with the mean flow and are therefore more problem-dependent and more challenging to handle with a single turbulence model [46].

2.2.3 RANS equations

In RANS, the random nature of turbulent flows is modelled using the Reynolds decomposition. A generic flow variable (ϕ) dependant on time t is defined in terms of a steady mean value ($\overline{\phi}$) and a fluctuating component (ϕ') which has an average value of zero, for $\phi' = \phi - \overline{\phi}$. In this section, the overbar $\overline{\ }$ will be used to denote a time average, and the prime symbol $'$ for the fluctuating mean-subtracted component. The mean is 'time-averaged', and defined in Equation 2.2:

$$\overline{\phi} = \frac{1}{\Delta t} \int_0^{\Delta t} \phi(t) dt \tag{2.2}$$

This technique is widely used in simulations of non-reacting flows where the density is assumed constant. However, using this approximation for the simulation of engine flows leads to difficulties. This is because piston compression and heat release due to combustion cause the density to fluctuate, resulting in many unknown fluctuating terms that would require additional modelling. For example, the instantaneous mass

continuity equation (units $\mathrm{kg.m^{-3}s^{-1}}$) in index form ($i = 1, 2, 3$) is given as [156]:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_i)}{\partial x_i} = 0 \tag{2.3}$$

for density ρ, velocity u_i, and spatial coordinate x_i. Substituting the Reynolds decomposition into the continuity equation gives:

$$\frac{\partial (\overline{\rho} + \rho')}{\partial t} + \frac{\partial [(\overline{\rho} + \rho')(\overline{u_i} + u_i')]}{\partial x_i} = 0 \tag{2.4}$$

and multiplying out the brackets in the second term produces the $\overline{\rho u_i}$, $\rho' \overline{u_i}$, $\overline{\rho} u_i'$, $\rho' u_i'$ quantities. Taking a time average of Equation 2.4 brings $\overline{\rho'}$ to zero by definition as well as the two middle quantities from the second term due to: $\overline{\rho' \overline{u_i}} = \overline{\rho'} \times \overline{u_i} = 0 \times \overline{u_i} = 0$, leaving the Reynolds-averaged continuity equation:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial (\overline{\rho u_i})}{\partial x_i} + \frac{\partial (\overline{\rho' u_i'})}{\partial x_i} = 0 \tag{2.5}$$

$\overline{\rho' u_i'}$ is an additional unknown term that represents the correlation between density and velocity fluctuations. Favre-averaging is an alternative technique that removes the need to model fluctuating terms like $\overline{\rho' u_i'}$ explicitly by taking a density-weighted approach, defined as:

$$\widetilde{\phi} = \frac{\overline{\rho \phi}}{\overline{\rho}}; \qquad \phi'' = \phi - \frac{\overline{\rho' \phi'}}{\overline{\rho}} \tag{2.6}$$

where the effects of density fluctuations are now included inside the ϕ'' term, so $\rho u_i = \rho(\widetilde{u} + u_i'')$. Expanding the brackets and taking a time average gives $\overline{\rho u_i} = \overline{\rho \widetilde{u_i}} + \overline{\rho u_i''}$. The last term in this equation can be shown to equal zero: $\overline{\rho u_i''} = \overline{\rho (u_i - \widetilde{u_i})} = \overline{\rho u_i} - \overline{\rho \widetilde{u_i}} = \overline{\rho u_i} - \overline{\rho \frac{\overline{\rho u_i}}{\overline{\rho}}} = \overline{\rho u_i} - \overline{\rho u_i} = 0$, so that $\overline{\rho u_i} = \overline{\rho} \widetilde{u_i}$. The time averaged continuity equation can therefore be written in terms of the Favre-averaged velocity.

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial (\overline{\rho u_i})}{\partial x_i} = \frac{\partial \overline{\rho}}{\partial t} + \frac{\partial (\overline{\rho} \widetilde{u_i})}{\partial x_i} = 0 \tag{2.7}$$

Similarly, the first two terms in the instantaneous momentum equation (units $\mathrm{kg.m^{-2}s^{-2}}$) contain multiplications of ρ and u whose fluctuations do not equal zero when averaged:

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \tag{2.8}$$

where p is the pressure and τ_{ij} is the viscous stress tensor. The ρu_i term in the momentum equation can be handled in the same way as in the continuity equation ($\overline{\rho u_i} = \overline{\rho}\widetilde{u_i}$). Reynolds averaging the convective term in the momentum equation would give: $\overline{\rho u_i u_j} = \overline{(\overline{\rho} + \rho')(\overline{u_i} + u_i')(\overline{u_j} + u_j')} = \overline{\rho}\overline{u_i}\overline{u_j} + \overline{\rho}\overline{u_i' u_j'} + \overline{u_i}\overline{\rho' u_j'} + \overline{u_j}\overline{\rho' u_i'} + \overline{\rho' u_i' u_j'}$. The Favre averaging technique simplifies this to: $\overline{\rho u_i u_j} = \overline{\rho(\widetilde{u_i} + u_i'')(\widetilde{u_j} + u_j'')} = \overline{\rho}\widetilde{u_i}\widetilde{u_j} + \overline{\rho u_i'' u_j''}$. The Favre-averaged momentum equation is therefore given in Equation 2.9.

$$\frac{\partial \overline{\rho}\widetilde{u_i}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u_i}\widetilde{u_j}}{\partial x_j} = -\frac{\partial \overline{\rho u_i'' u_j''}}{\partial x_j} - \frac{\partial \overline{p}}{\partial x_i} + \frac{\partial \overline{\tau_{ij}}}{\partial x_j} \tag{2.9}$$

The components of the $\overline{\rho u_i'' u_j''}$ term ($\mathrm{kg.m^{-1}s^{-2}}$ or Pa) are known as the Reynolds stresses (denoted as RS from here onwards), and they represent momentum exchanges due to convective transport by the turbulent eddies. It is the aim of RANS turbulence models to close RS.

2.2.4 RANS turbulence models

RANS turbulence models can be broadly classified into zero-, one- or two-equation models, and Reynolds stress equation models (RSM) [20]. Two-equation models are the most used in industrial RANS simulations as a compromise between computational cost and accuracy [157, 158]. On the one hand, zero- and one-equation models typically have reduced accuracy as they rely on a simple algebraic relation between the eddy velocity scale and the mean flow which neglects additional contributions to the turbulence due to transport such as convection and diffusion, while RSM approaches on the other introduce new balance equations for each of the Reynolds stresses, greatly

increasing computational cost [156]. The re-normalisation group (RNG) $k-\epsilon$ is a two-equation model that is widely used for flows with bulk compression such as ICE flows, as the fixed model constants in the standard turbulent kinetic energy transport equation have been shown to be incapable of capturing the disparate processes in the compression and expansion strokes [159–161]. The mathematics of the RNG approach are quite involved, so the interested reader is referred to Yakhot and Orszang [162] for more details.

The closure of RS with the RNG $k-\epsilon$ model begins with the Boussinesq hypothesis that the mechanism of energy transfer between the scales is analogous to the kinetic theory of gases, whereby molecular motion draws energy from the mean flow via molecular viscosity, generally in the direction of the velocity gradient [163]. RS is therefore modelled as being proportional to the mean rates of deformation, with the constant of proportionality as the turbulent viscosity μ_t (kg.m^{-1}s^{-1}), which models the additional transfer of momentum due to the turbulent eddies:

$$\mathrm{RS} = \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij} \tag{2.10}$$

where k is the turbulent kinetic energy (units m^{2}s^{-2}) and δ_{ij} is the Kronecker delta [†]. Dimensional analysis yields:

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \tag{2.11}$$

where C_μ is a dimensionless model constant and ϵ (m^{2}s^{-3}) is the dissipation rate of k. k and ϵ each have their own transport equations (units kg.m^{-1}s^{-3} and kg.m^{-1}s^{-4} respectively), consisting of terms representing transport via convection and diffusion,

[†]The Kronecker delta is defined as $\delta_{ij} = \begin{cases} 1, & \text{if} \quad i = j \\ 0, & \text{if} \quad i \neq j \end{cases}$.

and rates of production and destruction:

$$\frac{\partial(\overline{\rho}k)}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_i}k)}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] + P_k - \overline{\rho}\epsilon \tag{2.12}$$

$$\frac{\partial(\overline{\rho}\epsilon)}{\partial t} + \frac{\partial(\overline{\rho}\widetilde{u_i}\epsilon)}{\partial x_i} = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_i}\right] + C_{\epsilon 1}\frac{\epsilon}{k}P_k - C_{\epsilon 2}\overline{\rho}\frac{\epsilon^2}{k} - \mathcal{R} \tag{2.13}$$

where $C_{\epsilon 1}$ and $C_{\epsilon 2}$ are dimensionless model constants and the source term P_k is: $P_k = -\overline{\rho u_i'' u_j''}\frac{\partial \widetilde{u_i}}{\partial x_j}$, representing turbulent stress induced by a velocity gradient. The dimensionless Prandtl numbers σ_k and σ_ϵ represent the ratio of momentum diffusivity to thermal diffusivity and are also assumed to be constant. The traditional $k-\epsilon$ model was extended by Yakhot and Orszag using renormalisation group (RNG) methods to form the RNG $k-\epsilon$ model [162]. In the RNG $k-\epsilon$ model an additional correction term $\mathcal{R}$ is included in the ϵ transport equation which changes dynamically with the mean strain rate. The increased sensitivity to strain has improved the RNG $k-\epsilon$ model's performance in modelling separated flows, which are commonly found in engines due to flow motion around the intake valves.

2.2.5 LES equations

Rather than time-averaging, LES approaches make use of spatial filtering to separate the scales for modelling and explicit calculation. The following decomposition is used: $\phi' = \phi - \overline{\phi}$, where in this section, the overbar $\overline{\ }$ represents spatial filtering rather than time averaging, and ϕ' is the sub-grid component. The filtered component is obtained by applying a spatial filiter G to the instantaneous flow variable ϕ:

$$\bar{\phi}(x,t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(x_i, \xi_i, \Delta)\phi(\xi_i, t)\, d\xi_1\, d\xi_2\, d\xi_3 \tag{2.14}$$

where G depends on the nearby grid points ξ in each dimension as defined by the cutoff width Δ. G is commonly defined as a uniform box filter:

$$G(x_i, x_i', \Delta) = \begin{cases} \frac{1}{\Delta^3} & \text{if } |x_i - \xi_i| \leq \frac{\Delta}{2} \text{ for } i = 1, 2, 3 \\ 0 & \text{if } |x_i - \xi_i| > \frac{\Delta}{2} \text{ for } i = 1, 2, 3. \end{cases} \tag{2.15}$$

In practice, the value of Δ is commonly taken to be on the order of the computational grid size, so that $\Delta = \sqrt[3]{\Delta x \Delta y \Delta z}$ for width Δx, length Δy, and height Δz. Applying the filtering operation to the instantaneous continuity equation (Equation 2.3) gives:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \overline{(\rho u_i)}}{\partial x_i} = 0 \tag{2.16}$$

assuming the use of a uniform filter such that the filtering operation commutes with the differential for $\frac{\overline{\partial \rho}}{\partial t} = \frac{\partial \bar{\rho}}{\partial t}$ and $\frac{\overline{\partial (\rho u_i)}}{\partial x_i} = \frac{\partial \overline{(\rho u_i)}}{\partial x_i}$. Favre-filtering is applied in the same way as Favre-averaging:

$$\widetilde{\phi} = \frac{\overline{\rho \phi}}{\bar{\rho}} \tag{2.17}$$

where in this case $\widetilde{\phi}$ is a Favre-filtered quantity and overbars represent spatially filtered terms. The filtered continuity equation can therefore be written as:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \overline{\rho} \widetilde{u}_i}{\partial x_i} = 0. \tag{2.18}$$

Similarly, the filtering the instantaneous momentum equation (Equation 2.8) gives:

$$\frac{\partial \overline{(\rho u_i)}}{\partial t} + \frac{\partial \overline{(\rho u_i u_j)}}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial \overline{\tau_{ij}}}{\partial x_j}. \tag{2.19}$$

Using Favre-weighted quantities, we have $\overline{\rho u_i} = \overline{\rho}\widetilde{u}_i$ and:

$$\begin{aligned}\overline{\rho u_i u_j} &= \overline{\rho}\widetilde{u_i u_j} \\ &= \overline{\rho}\widetilde{u}_i\widetilde{u}_j - \overline{\rho}\widetilde{u}_i\widetilde{u}_j + \overline{\rho}\widetilde{u_i u_j} \\ &= \overline{\rho}\widetilde{u}_i\widetilde{u}_j + \overline{\rho}\left(\widetilde{u_i u_j} - \widetilde{u}_i\widetilde{u}_j\right).\end{aligned}$$

The unclosed residual is denoted as the sub-grid stress tensor:

$$\widetilde{\tau}^r_{ij} = \bar{\rho}\left(\widetilde{u_i u_j} - \widetilde{u}_i\widetilde{u}_j\right). \tag{2.20}$$

Finally, the viscous stress tensor can be written in Favre-filtered form as $\overline{\tau_{ij}} = \overline{\tau_{ij}} + \widetilde{\tau}_{ij} - \widetilde{\tau}_{ij}$. This leaves the Favre-filtered momentum equation:

$$\frac{\partial\bar{\rho}\tilde{u}_i}{\partial t} + \frac{\partial\bar{\rho}\tilde{u}_i\tilde{u}_j}{\partial x_j} = -\frac{\partial\bar{p}}{\partial x_i} - \frac{\partial\bar{\rho}\widetilde{\tau}^r_{ij}}{\partial x_j} + \frac{\partial\tilde{\tau}_{ij}}{\partial x_j} + \frac{\partial}{\partial x_j}\left(\bar{\tau}_{ij} - \tilde{\tau}_{ij}\right) \tag{2.21}$$

where $\frac{\partial}{\partial x_j}\left(\bar{\tau}_{ij} - \tilde{\tau}_{ij}\right)$ represents a sub-filter viscous contribution, which is often considered to be negligible [163].

2.2.6 LES turbulence models

Perhaps the simplest LES tubulence model for closing the sub-grid stress tensor τ^r_{ij} is the Smagorinsky model, which begins with a Boussinesq-style hypothesis [163, 164]:

$$\widetilde{\tau}^r_{ij} = -2\mu_{\text{SGS}}\widetilde{S}_{ij} + \frac{1}{3}\widetilde{\tau}_{ii}\delta_{ij} \tag{2.22}$$

where the strain rate tensor is:

$$\widetilde{S}_{ij} = \frac{1}{2}\left(\frac{\partial\widetilde{u}_i}{\partial x_j} + \frac{\partial\widetilde{u}_j}{\partial x_i}\right) \tag{2.23}$$

and the sub-grid viscosity is deconstructed using dimensional analysis via characteristic length and velocity scales l_0 and U_0:

$$\mu_{\text{SGS}} \propto \rho l_0 U_0. \tag{2.24}$$

As the small eddies are assumed to be isotropic, they can be characterised by their size, so l_0 is modelled as the average sub-grid eddy size, calculated as the grid size multiplied by a constant:

$$l_0 = C_s \Delta \tag{2.25}$$

and the characteristic velocity of the eddies is defined by the average velocity gradients:

$$U_0 = l_0 \sqrt{2 \widetilde{S}_{ij} \widetilde{S}_{ij}} \tag{2.26}$$

so that finally:

$$\mu_{\text{SGS}} = \rho \left(C_s \Delta\right)^2 \sqrt{2 \widetilde{S}_{ij} \widetilde{S}_{ij}} \tag{2.27}$$

However, it is challenging to accurately represent a range of turbulent fields in rotating or sheared flows, near solid walls, or in transitional regimes with a single constant C_s [165]. Therefore, Germano *et al.* [166] proposed the Dynamic Smagorinsky model, which modifies C_s according to predictions of the sub-grid turbulent stresses on a local basis. To this end, two filters are defined, the regular grid filter $\widetilde{\ }$ as well as a coarser test filter $\widehat{\ }$. Applying the test filter to Equations 2.20 and 2.21 give:

$$\widehat{\widetilde{\tau}}^r_{ij} = \bar{\bar{\rho}} \left(\widehat{\widetilde{u_i} \widetilde{u_j}} - \widehat{\widetilde{u_i}} \widehat{\widetilde{u_j}} \right) \tag{2.28}$$

$$\widehat{\tau}^r_{ij} = \bar{\bar{\rho}} \left(\widehat{\widetilde{u_i} \widetilde{u_j}} - \widehat{\widetilde{u}_i} \widehat{\widetilde{u}_j} \right). \tag{2.29}$$

A difference between these two equations gives the effects of the stresses between the grid and test filters:

$$L_{ij} = \widehat{\tau}^r_{ij} - \widehat{\widetilde{\tau}}^r_{ij} = \widehat{\widetilde{u}_i \widetilde{u}_j} - \widehat{\widetilde{u}}_i \widehat{\widetilde{u}}_j. \tag{2.30}$$

Germano [166] and Lilly [167] showed that L_{ij} can be written into an error function that represents the expected scale-dependent error, which can then be minimised in order to update the Smagorinsky constant:

$$C_s^2 = \frac{\langle L_{ij} M_{ij} \rangle}{\langle M_{ij} M_{ij} \rangle} \tag{2.31}$$

where $\langle \quad \rangle$ indicates an average and:

$$M_{ij} = -2\widehat{\Delta}^2 |\widehat{\widetilde{S}}| \widehat{\widetilde{S}}_{ij} + 2\widetilde{\Delta}^2 \widehat{|\widetilde{S}| \widetilde{S}}_{ij}. \tag{2.32}$$

This version of the Dynamic Smagorinsky model is widely used, and was implemented by collaborators from the University of Modena and Reggio Emilia in order to gather the LES data that were analysed in this thesis [44].

2.3 Particle image velocimetry

Optical access in ICEs has played a crucial role in visualising and understanding in-cylinder processes since at least the 1930s with the use of qualitative Schlieren imaging [168]. To provide optical access inside an engine cylinder, metal sections can be replaced with components made of transparent materials, such as a silica cylinder liner and piston window. Particularly since the development of digital image processing techniques in the 1980s, and high-power lasers and high-speed cameras in the 2000s, particle image velocimetry (PIV) has emerged as an effective optical technique for taking flow velocity measurements over large areas [169]. PIV offers a significant advantage over previously-established techniques that are primarily designed for point

measurements such as hot-wire anemometry and laser-Doppler velocimetry [170]. In PIV, velocity vectors are obtained across a flow field (typically a two-dimensional plane) by tracking the displacement of seeder particles between pairs of images taken over short time intervals. The result is a set of images that contain velocity vectors at discrete points in the measurement plane, representing the motion of the neighbouring fluid [169]. The seeder particles need to be small enough to follow the turbulent flow motion without influencing the flow or experiencing slippage, but large enough to scatter the light and create a clear signal. Adrian and Westerweel [169] suggest that $\approx$1 μm diameter oil droplets (olive oil is commonly used in practical atomisers and aerosols for PIV systems) are sufficient for scattering light with negligible slippage losses.

A conventional planar PIV setup in an engine involves passing a laser beam through a set of optics to produce a 2D light sheet, which is then directed into the cylinder to illuminate the measurement plane. An example PIV schematic is provided in §2.5.1. Light scattered by the seeder particles in the flow is captured by the high-speed camera. The short timescales involved (the piston in an engine running at 1000 rpm moves at $\approx$167 μs per crank angle) necessitate short exposure times for the camera, so it is important to use high-power lasers (typically 5-200 mJ [170]) in order to produce sufficient illumination and clear final images. Dual-cavity lasers that can fire two independent laser beams are typically used as more cost-effective solutions than ultra-high repitition rate single-cavity lasers for producing pulses that are rapid enough to take pairs of PIV images separated by tens of microseconds [169].

To generate velocity vectors from the raw image pairs, the images are split into several interrogation windows; the windows from each image in the pair are then cross-correlated such that the correlation peak defines the most likely displacement

of the particles. As the time interval between the snapshots is known, a velocity vector can therefore be calculated and assigned to the interrogation region. The interrogation windows are often chosen to overlap, commonly by 50%, in order to reduce random errors [170]. Multi-pass approaches are sometimes implemented to increase the range of velocities that can be recorded by the system (known as the dynamic range); a larger interrogation window size is used at first to capture the large-scale motion, followed by smaller windows that correct the velocity calculations at the smaller scales. Final window sizes are commonly chosen to encompass around ten seeder particles in order to provide sufficiently reliable displacement estimates without excessively compromising the spatial resolution [169, 171]. An illustration of the PIV vector generation process is provided in Figure 2.2.

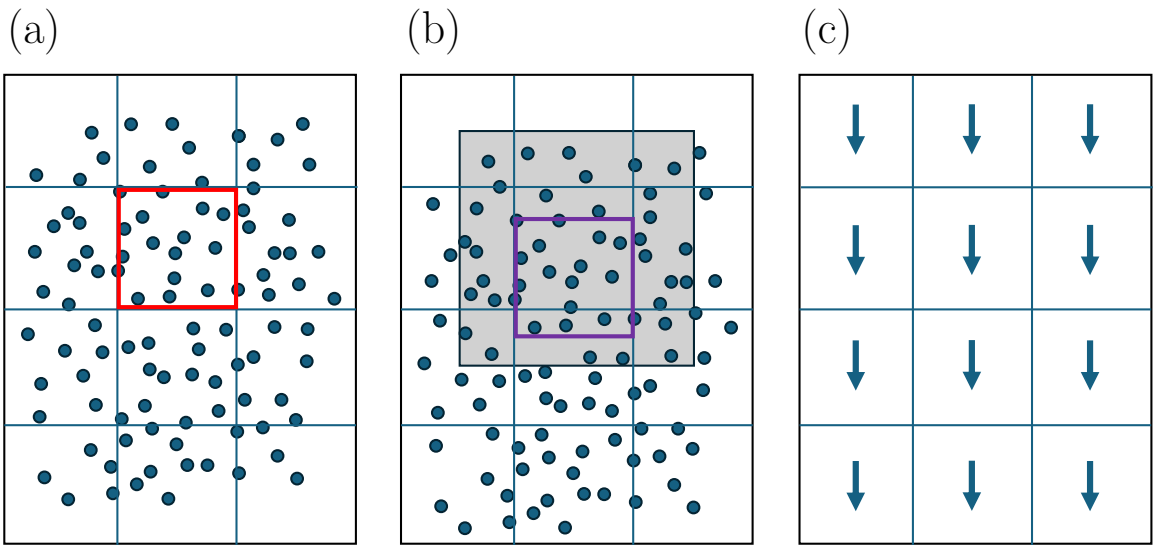

Figure 2.2: A diagram of the PIV vector generation process. Part (a) illustrates an example raw PIV image showing the seeder particles, divided into a set of interrogation windows. Part (b) shows the image from the next timestep, where the particles have been shifted downwards by the turbulent flow. The particles in the red interrogation window from the previous image have been identified by the cross-correlation algorithm and are located inside the purple square, where the grey region is the user-defined particle search zone. Finally, in part (c) the average displacements between the particles in the interrogation windows have been calculated in order to produce a set of velocity vectors on the PIV grid.

2.4 CFD datasets

Two CFD datasets are explored in this thesis for compairson to PIV data. Firstly, a RANS dataset is considered due to the widespread usage of the RANS technique in the automotive engineering industry. In this configuration, a single RANS snapshot per crank angle needs to be validated against a larger set of PIV images (typically $100-1000$ per crank angle). Therefore, a many-to-one operation is desired that can collapse the PIV data onto a single image per crank angle that can serve as a validation target for the RANS data. Secondly, an LES dataset is investigated, as the LES technique is capable of capturing more complex turbulent behaviours while still remaining feasible for dedicated research applications. In this configuration, the validation process requires modification as two ensembles of data are involved, with both the LES and PIV datasets supplying multiple realisations of data per crank angle.

The RANS simulations presented in this thesis were designed and run by the author in order to complement the PIV dataset from the Oxford optical engine. This engine is of particular industrial relevance, being based on the Jaguar Land Rover (JLR) AJ200 gasoline direct injection engine. On the other hand, the LES data were gathered by the Gruppo Motori at the University of Modena and Reggio Emilia. They ran simulations of the Darmstadt optical engine, which is a well-characterised and widely-studied apparatus among international research institutions [44, 171, 172]. The respective computational setups are described below.

2.4.1 Oxford RANS

Engine simulations were run in Simcenter STAR-CCM+ In-cylinder solution v2021.2 in a RANS framework. The turbulence model was the RNG $k-\epsilon$ two-layer model

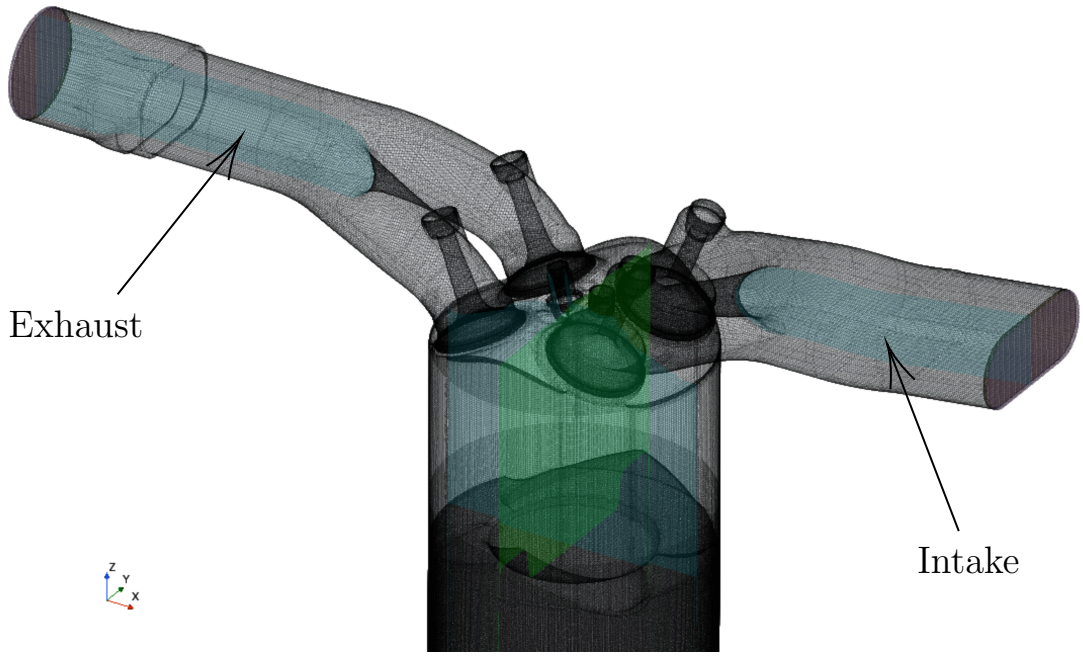

Figure 2.3: Sample RANS CFD mesh for the Oxford TPSRG optical engine, with a base size of 0.7 mm and volumetric refinements down to 0.35 mm around the valves. The tumble ($x - z$) plane is shown in blue, and the cross-tumble ($y - z$) plane is shown in green. The CFD inlet and outlet boundaries are also shown at the ends of the intake and exhaust plena, which coincide with the locations of the pressure transducers in the physical engine.

with default constants [173], and the heat transfer model was the Grumo-UniMORE model [85]. A trimmed mesh was used with hexahedral cells throughout most of the domain and prismatic cells next to wall surfaces. The base mesh size was 0.7 mm with volumetric refinements around the valves set to a minimum 0.35 mm, for approximately 3.4 million cells at bottom dead-centre (BDC). The mesh is shown along with the tumble and cross-tumble planes in Figure 2.3. The available geometry for the CFD included a spark plug and an injector that aimed to mirror the experimental setup, in which a metal blank was used in place of a real spark plug to reduce the background scatter for the PIV measurements.

The crevice length (defined as the distance between the top piston ring and the piston surface) was increased to match the ensemble-average experimental pressure trace,

which accounts for the unknown amount of blow-by losses in the physical engine. This approach is commonly taken when simulating non-reacting optical engines [44, 174]. More detailed models of crevice flows and blow-by losses require knowledge of parameters surrounding thermal expansion of the piston rings and ageing rates, which are challenging to obtain accurately [175]. To test the impact of modifying the crevice region on the in-cylinder flow fields, Stocchi *et al.* [176] modelled a firing single-cylinder SI engine in a RANS framework with and without a crevice region in the piston geometry. They found that the complete removal of the crevice volume from the CFD geometry had a minimal impact on the predicted turbulence kinetic energy and dissipation rate for all crank angles prior to spark timing.

Ensemble-averaged measured intake and exhaust valve lift profiles were used as inputs to the simulation. Boundary and initial conditions were as follows.

- Experimentally-measured ensemble-averaged crank angle-resolved pressure at the inlet and outlet.

- Temperatures at the inlet (experimentally controlled to be constant) and outlet (constant due to steady state) of 318 K and 354 K, respectively.

- Constant turbulent intensity of 0.1 and turbulent length scale of 1 mm at both the inlet and the outlet, as suggested in the STAR CCM+ documentation.

The simulations were initialised during the exhaust stroke of the previous cycle (at −600 CAD after firing top dead-centre (aTDCf) in order to take measurements from −330 CAD aTDCf onwards) in order to improve the convergence of the simulation. Multi-cycle analyses were also conducted to study the effect of the initial conditions;

the differences in peak pressure were all within 0.2%, so results from the first cycles were retained for this work. Note that the results presented in this thesis arise from two separate CFD simulations that were run in order to reflect the separate tumble and cross-tumble experiments. This is because the experimentally-measured inlet and outlet pressures differed between the two experiments, despite efforts to control the conditions at the same test point. This resulted in two different sets of pressure boundary conditions for the CFD, which manifested as two slightly different peak pressures. This difference is quantified as part of the results discussed in §4.4.1.

2.4.2 Modena LES

LES conducted by the Modena group of the engine cold-flow were run using Simcenter STAR-CCM+ In-cylinder solution v2020.2. Turbulence was modelled with the Dynamic Smagorinsky Subgrid Scale Model [166]. To improve computational efficiency, the overall domain was restricted to the space between the intake and exhaust pressure probes, which were treated as inlet and outlet boundaries. Time-dependent pressure and temperature boundary conditions were sourced from a 1D GT-Power model of the engine, which was previously validated in Refs.[174, 177].

The computational grid, shown in Figure 2.5, was primarily hexahedral with a core grid size of 0.75 mm in the cylinder. Fixed and moving control volumes were added near the valves to refine the grid size to 0.375 mm, while the mesh in the ports was coarsened to 1.5 mm. Eight prismatic cell layers were applied to all walls, where the first layer measured 10 μm and the total thickness was set to 0.6 mm. Overall, the maximum number of cells was approximately 11 million (of which 5.1 million cells were located inside the cylinder) at bottom dead-centre (BDC), while the cells at top dead-centre (TDC) numbered around 5.5 million (with 2.7 million inside the cylinder). The computational domain included a slight extension of the crevice volume which

reduced the compression ratio by just under 5%. This choice was made following extensive investigations reported in [66, 177] to account for the unknown levels of blow-by losses in the physical engine due to the crevice volume.

The simulations were initialised with a two-step approach. Firstly, four consecutive RANS cycles were performed to remove the effect of the initial conditions and to reach cyclic convergence. Secondly, a single LES cycle was run to obtain a consistent initial field for the subsequent simulations. Once initialised, 50 consecutive LES cycles were run, providing the LES dataset that was analysed in this thesis. The LES quality criterion as suggested by Pope [40] was used to verify the LES simulation:

$$Q = \frac{k_{\text{res}}}{k_{\text{res}} + k_{\text{sgs}}} \tag{2.33}$$

where k_{res} is the resolved kinetic energy and k_{sgs} is the sub-grid kinetic energy, which is modelled following Yoshizawa [178, 179] where $\langle \quad \rangle$ indicates an ensemble average:

$$k_{\text{sgs}} = 2C_i\Delta^2 \left|\langle S_{ij} \rangle\right|^2 \tag{2.34}$$

$$C_i = 0.202. \tag{2.35}$$

The LES results exhibited a quality index of more than 0.8 for every crank angle and engine cycle, indicating that at least 80% of the turbulence energy is resolved on the grid, classifying the simulation as sufficiently well-resolved [40]. A cycle convergence test was run using the relevance and magnitude indices (see §4), with minimal differences reported in the ensemble averages between the first 25 and 50 LES cycles according to these metrics, indicating satisfactory convergence for use in this analysis.

2.5 PIV datasets

The PIV datasets used for comparison with the CFD data were gathered by the Oxford Thermal Propulsion Systems Research Group and the Institute of Reactive Flows and Diagnostics at the Technical University of Darmstadt, as listed in Table 2.4. Furthermore, in order to establish the benchmark for filling gaps in PIV images, data from the transparent combustion chamber (TCC-III) gathered by the General Motors University of Michigan Automotive Cooperative Research Laboratory were used. This is because the TCC-III data were released open-source to the public, which is essential for allowing the benchmark results to be improved upon by the wider ICE and fluid mechanics research community. In addition, the TCC-III engine was specifically designed to promote significant cycle-to-cycle variations (CCV) due to the two-valve, pancake chamber with a large piston-diameter to valve-diameter ratio, as opposed to an engine with the more naturally directed flow of a 4-valve pent roof [152]. The use of a challenging dataset can push predictive methods to become more robust and generalisable while uncovering important failure modes, and this challenge-seeking philsopophy was successful when using TCC-III data to develop robust numerical methods for CFD and POD in the past [55, 63, 64, 180, 181].

2.5.1 Oxford

Experiments were run by motoring (i.e. running using a dynamometer without fuelling) an optically-accessible single-cylinder SI engine. Table 2.1 contains engine specifications along with details of the operating conditions for the test point studied. The cylinder liner consists of a fused silica upper section to provide optical access, and has a variable height of between 25 and 39 mm. The lower part of the liner is metal with internal water cooling required by the Torlon piston rings. The optical engine experiences piston ring ageing which affects the quality of the seal with the

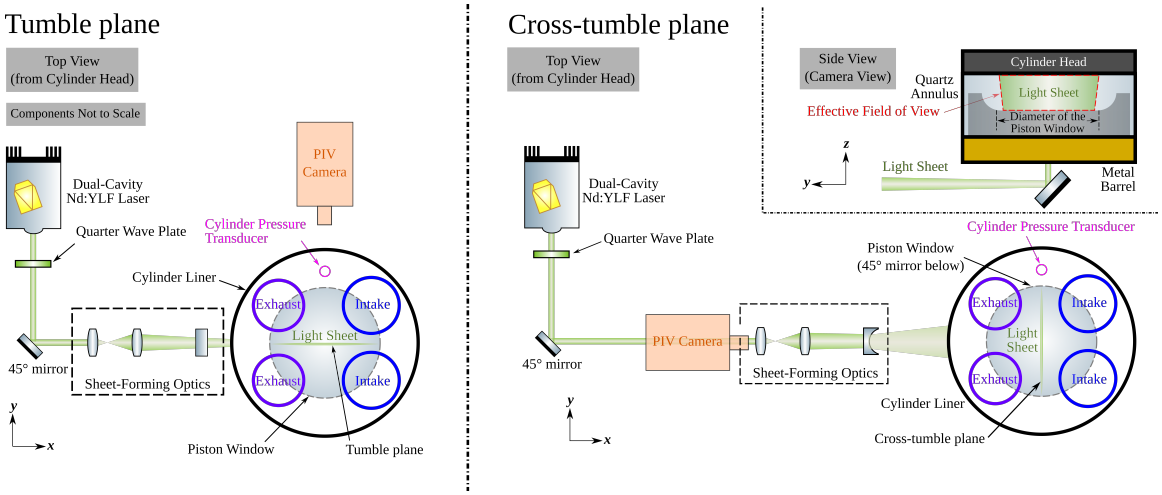

Figure 2.4: PIV set-up for tumble (left) and cross-tumble (right) measurements. In both cases, the light sheet is reflected up into the engine cylinder through the piston, and the PIV camera is positioned above the laser beam in line with the optical window (quartz annulus) in the cylinder. Note the different sets of sheet-forming optics and the resultant light sheet orientation for each plane; for example, on the cross-tumble plane where the combination of spherical and cylindrical lenses form a horizontal light sheet that is then reflected up through the bottom of the piston. Adapted from [54, 106].

wall, so there are unknown levels of blow-by losses.

Table 2.1: Oxford engine specifications and operating conditions.

Parameter	Description
Valves per cylinder [-]	2 intake, 2 exhaust
Bore × stroke [mm]	85.0 × 90.3
Compression ratio [-]	12.5
Engine speed [rpm]	1500
Intake manifold absolute pressure [kPa]	80
Intake air volume flow rate [L/s]	1.57
Intake air temperature [°C]	45

Crank angle-resolved pressure data were recorded with a Kistler 6043A60 pressure transducer. The PIV method utilised a Photonics Industries DM20-527-DH laser with a set of optics to create a light sheet approximately 1 mm thick in the measurement region. A 45° mirror was used to reflect the light beam upwards through a fused silica window insert in the piston and illuminate seeded olive oil droplets 0.2−0.9 μm in

diameter on the desired planes in the cylinder. Images were then taken through the fused silica liner with a Vision Research Phantom VEO 710L camera. The schematic for the set-up is shown in Figure 2.4. The piston window had a diameter of 46 mm, and the width × height dimensions of the tumble and cross-tumble plane fields of view after processing the PIV data were 48×38 mm and 46×17 mm, respectively.

Two separate PIV experiments were run at the same test point, one for each of the planes in the cylinder. The two planes of interest are illustrated together in Figure 2.3 with the tumble ($x-z$ with 1 mm offset away from the flywheel) plane shown in purple, and the cross-tumble ($y-z$, central) plane shown in cyan. Each PIV experiment consisted of a total of 300 cycles of data, with each set of 300 comprising three runs of 100 consecutive cycles. Within each engine cycle, PIV data were gathered every 5 crank angle degrees (CAD) in the range −330 to −30 CAD after the firing top dead-centre (aTDCf). At an engine speed of 1500 rpm, this corresponds to a data acquisition frequency of 1800 Hz. Vector fields were generated from pairs of images that were processed using the DaVis software (LaVision, V. 8.4.0). A multi-pass algorithm was used to reduce the interrogation window size from 128×128-pixels to 32×32-pixels with 50% overlap, resulting in a vector spacing of 0.84 mm. More details on the experimental set-up and the PIV data post-processing methods can be found in previous publications [54, 106].

2.5.2 Darmstadt

The Darmstadt optical engine is a single-cylinder DISI engine with optical access provided via a transparent 55 mm tall quartz-glass cylinder liner and a flat quartz-glass 75 mm diameter piston window. The PIV data were gathered at the 'OP. A' engine operating point, which is characterised by an engine speed of 800 rpm and an intake pressure of 0.95 bar. The seeder particles for the PIV measurements

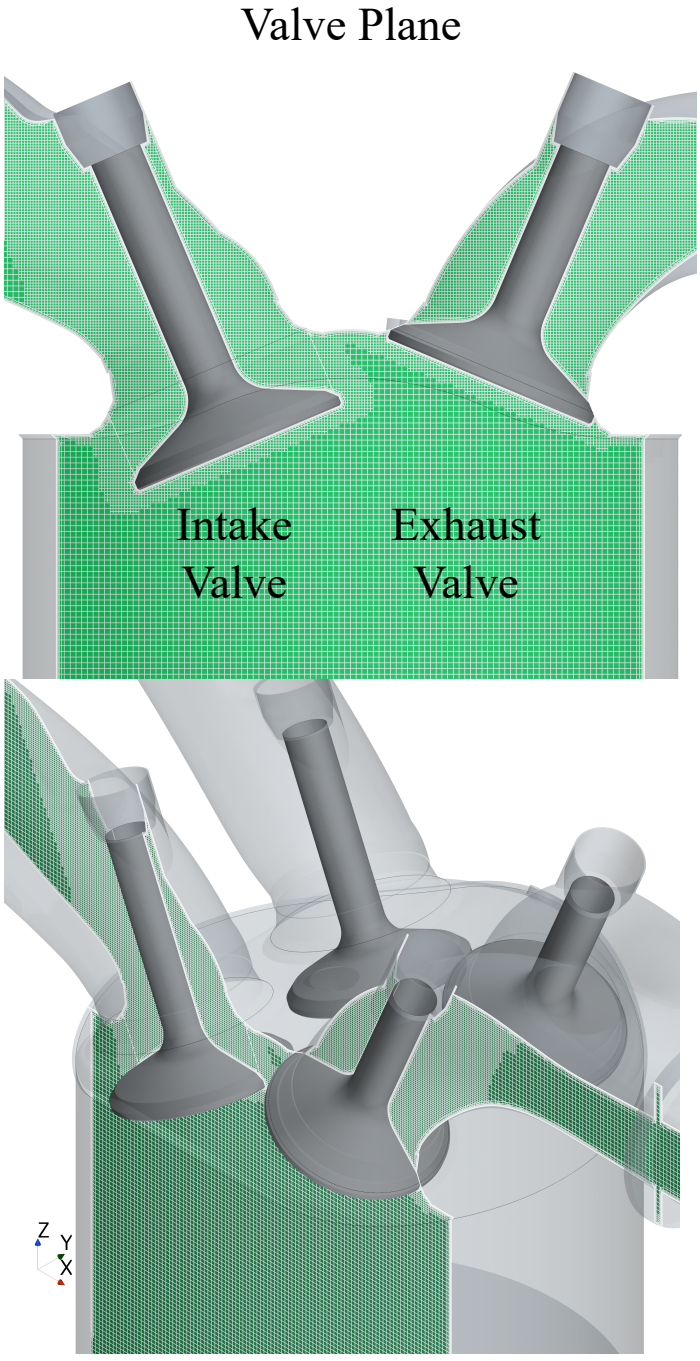

Figure 2.5: Diagram showing the valve plane in green for both an in-plane view (top) and an isometric view (bottom). The plane is offset from the cylinder centre, cutting through an intake valve and an exhaust valve on one side of the engine cylinder.

were silicone oil droplets measuring ≈ 1 μm in diameter. The seeding density was optimised to produce between 8 and 10 particle pairs in the final interrogation window size. The raw images were processed with the DaVis, LaVision software using a multi-pass algorithm with a decreasing window size from 96×96 to 32×32 pixels with a 75% overlap. The Edgewave INNOSLAB IS4 II-DE laser was used pulsed at 0.75 mJ with a higher rate of 4.8 kHz to provide data at every crank angle, and provided a light sheet 0.8 mm thick. A multi-pass iteration scheme was used for the image processing with a final interrogation window of 32 x 32 pixel and 75% overlap, resulting in a spatial resolution of 0.60 mm.

The PIV dataset consists of 250 snapshots of consecutive cycles measured on the so-called valve-plane. This plane is offset from the cylinder centre by 19 mm in order to align with the intake valves, as shown in Figure 2.5. PIV measurements were taken from 360 to 720 CAD after firing the top dead-centre (aTDCf). This study focuses on the results at two fixed phases, namely 470 and 700 CAD aTDCf. At 470 CAD, the intake valves are at their maximum lift, and the intake jet can be observed, which is an important physical phenomenon in engine research. 700 CAD is close to a typical spark timing and therefore represents one of the final states of fuel-air mixing before combustion, where typically a higher CCV is present in both the experiments and large eddy simulations. More details can be found in [171, 182].

Table 2.2: Darmstadt engine specifications and operating conditions.

Parameter	Description
Valves per cylinder [-]	2 intake, 2 exhaust
Bore × stroke [mm]	86.0 × 86.0
Compression ratio [-]	8.7
Engine speed [rpm]	800
Intake manifold absolute pressure [kPa]	95
Intake air temperature [°C]	23.9

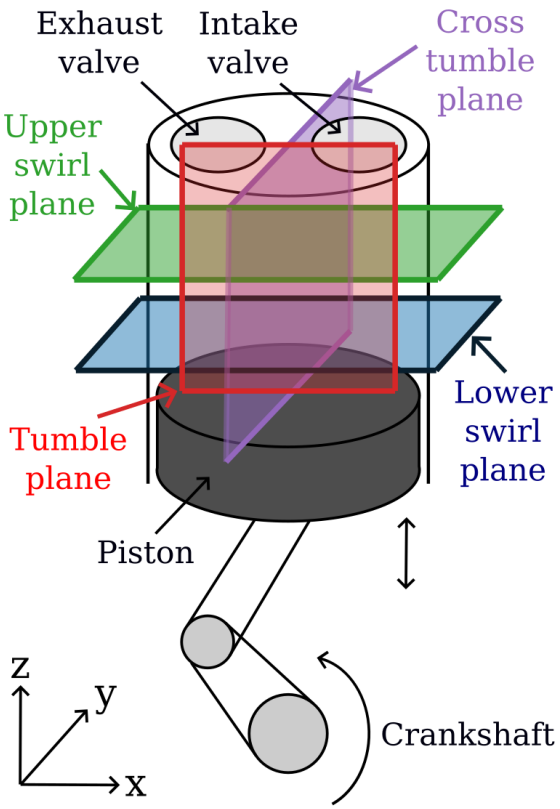

Figure 2.6: Schematic showing the TCC-III and associated PIV measurement planes.

2.5.3 Michigan-General Motors

The TCC-III is a port-fuelled spark-ignition single-cylinder optical research engine with a single intake valve, an exhaust valve, and a pancake-shaped combustion chamber. Optical access is provided via a full quartz cylinder and a 70 mm diameter flat quartz piston window. A Darwin Duo, Quantronix laser was used to illuminate silicone-oil droplets 1 mm in diameter, and images were taken with a Vision Research, Phantom v1610 camera. A multi-pass algorithm was used to process the vectors, with a decreasing interrogation window size from 18×128 to 32×3 pixels with 50% overlap. The final window size produced vectors with a spatial resolution of 1.25 mm.

Numerous expeirments were run across a six month experimental campaign to assess the test-to-test repeatability. Overall, Schiffmann *et al.* [152] achieved a test-to-test

repeatability that were within the experimental uncertainty of the measurements, with an average velocity error of 1.5−8% between the mid-intake and mid-compression strokes (90−270 CAD) depending on the PIV measurement plane. Data are available for four PIV planes, namely the tumble, cross-tumble, upper-swirl, and lower-swirl planes. These planes are shown in the schematic in Figure 2.6. However, the investigation in this thesis was restricted to data on the lower-swirl plane in order to simplify the analysis, as the field of view remains constant with the varying piston position. 1041 cycles of data were used across 90−70 CAD for the construction of the machine learning benchmark presented in Chapter 6.

Table 2.3: TCC-III engine specifications and operating conditions.

Parameter	Description
Valves per cylinder [-]	1 intake, 1 exhaust
Bore × stroke [mm]	92.0 × 86.0
Compression ratio [-]	10
Engine speed [rpm]	1300
Intake manifold absolute pressure [kPa]	40
Intake air temperature [°C]	45

2.6 Summary

An overall summary of the data used in this thesis is provided in Table 2.4.

Table 2.4: Dataset summary regarding their usage in this thesis.

Dataset	**Engine**	**Snapshots per CAD**	**Specific advantages**	**Usage**
Oxford RANS	Oxford	1	Industrially relevant	RANS validation
Oxford PIV	Oxford	300	Industrially relevant	RANS validation
Modena LES	Darmstadt	50	Widely studied	LES validation
Darmstadt PIV	Darmstadt	250	Widely studied	LES validation
TCC-III PIV	TCC-III	1157	Open-source Challenging CCVs	Flow reconstruction ML benchmark

Chapter 3

Dimensionality reduction methods

3.1 Overview

This chapter introduces the dimensionality reduction methods which form the basis for most of the data analysis conducted in this thesis. To introduce the topic, this chapter begins with a discussion of the simple ensemble mean and how it interfaces with cyclic vector field data from engines. More advanced dimensionality reduction techniques are then presented, followed by a discussion of their relative advantages with respect to their potential for improving engine CFD validation processes.

3.2 Engine PIV data and the ensemble mean

As ICEs are cyclic devices, PIV data from ICE flows are often gathered as a set of snapshots taken at certain crank angles (phases) across a number of different engine

cycles [183]. This presents an opportunity to consider two different time vectors, either by taking measurements at consecutive phases within the same cycle, or at a fixed phase from cycle to cycle. Separate analysis approaches have been developed for each framework; namely, phase-invariant methods for considering consecutive phases, and phase-dependent methods for consecutive cycles [184]. Phase-invariant methods can be used to track the progression of flow structures through consecutive time within an engine cycle [61, 78]. However, this technique relies on potentially expensive high-speed equipment, and physically interpreting phase-invariant flow statistics can be challenging. This is because the piston moves and engine flows exhibit transient behaviour within each cycle, causing statistical descriptors such as the mean and variance to lose physical significance [59, 96].

On the other hand, phase-dependent frameworks are used to investigate flow structures that appear at the same point from cycle to cycle. The work in this thesis focusses on phase-dependent approaches due to the broader availability of datasets of this kind as a result of the reduced demand on high-speed experimental apparatus. Crucially for measurements of turbulence, the chaotic nature of turbulent flows means that they are extremely sensitive to perturbations in the initial conditions, for example vibrations of the apparatus, small inhomogeneities in temperature or the presence of impurities [40]. Consequently, measurements of turbulent flows will differ between multiple runs of the same experiment; or, in the case of an engine, from cycle to cycle. For the CFD validation process, this means that achieving a realisation-by-realisation match between simulated velocity fields and experiments is highly unlikely; therefore, in order to reliably capture the behaviour of a flow feature that appears at the same phase in every cycle, statistical descriptors of the flow are constructed using numerous realisations of the flow at that point [40].

These statistical descriptors can be conveniently calculated by presenting the PIV data in the form of an $N \times M$ data matrix $\mathbf{D}$, with N variables and M sets of measurements (snapshots). Each column of $\mathbf{D}$ contains a separate set of measurements arranged as a column vector, denoted as $\mathbf{d}_{:,m}$ for the mth snapshot where $m = 1, 2, \ldots, M$. Each $\mathbf{d}_{:,m}$ contains a set of two-dimensional velocity vectors for one snapshot, which are recorded at distinct locations on the plane illuminated by the laser sheet. These distinct measurement locations make up the PIV grid, which arises from the interrogation window algorithm discussed earlier (see §2.3). The velocity measurement at each grid point is then treated as a separate variable in subsequent analyses. Furthermore, to preserve the directional information, the velocity measurements can be stored as two separate horizontal ($\mathbf{d^{(1)}}$) and vertical ($\mathbf{d^{(2)}}$) components. Each snapshot therefore consists of $N/2$ measurements of $\mathbf{d^{(1)}}$ and $N/2$ measurements of $\mathbf{d^{(2)}}$, for $N/2$ distinct locations in the PIV grid and N total variables. Therefore, each snapshot $\mathbf{d}_{:,m}$ can be written as a column vector containing individual measurements d:

$$\mathbf{d}_{:,m} = \left[d^{(1)}_{1,m} \ , \ \ldots \ , \ d^{(1)}_{N/2,m} \ , \quad d^{(2)}_{1,m} \ , \ \ldots \ , \ d^{(2)}_{N/2,m} \right]^{\mathrm{T}} \tag{3.1}$$

where $^{\mathrm{T}}$ denotes the matrix transpose. The full data matrix $\mathbf{D}$ is then:

$$\mathbf{D} = \begin{bmatrix} | & & | \\ \mathbf{d}_{:,1} & \cdots & \mathbf{d}_{:,M} \\ | & & | \end{bmatrix} . \tag{3.2}$$

The simplest statistical descriptor is perhaps the time average, also known as the ensemble mean (EM) for discrete phase-dependent datasets. The EM (denoted using the overbar $\overline{\ }$) is calculated by taking a column-wise average for each location in the

PIV grid:

$$\overline{\mathbf{d}} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{d}_{:,m}. \tag{3.3}$$

However, in situations with strong CCVs, the average flow is not a prominent feature in the dataset; this leads to the EM becoming unrepresentative of each of the individual snapshots, as demonstrated in Chapter 4. This observation motivates the investigation of more powerful methods for representing the most important features of a turbulent flow, such as dimensionality reduction.

3.3 Dimensionality reduction

Turbulent flow field data are often considered to be high-dimensional, meaning that there are a large number of variables such as measurement locations and numbers of snapshots [121]. Dimensionality reduction is a category of unsupervised machine learning (ML), and defined as the transformation of high-dimensional data into a meaningful representation of reduced dimensionality, where this reduced dimension is ideally the minimum number of variables needed to account for the observed properties of the data [185]. By placing emphasis on dominant features of the data, dimensionality reduction offers benefits such as data compression, feature extraction, and increased robustness to noise [107, 120, 186].

In fluid mechanics, dimensionality reduction approaches make use of the fact that dominant patterns exist within complex flows. For example, coherent structures exist in many turbulent flows, which are observed as large-scale bulk motions that persist over time [187]. The connection between statistical insights from dimensionality reduction and physical features of the flow has helped such techniques to become prominent in many fluid mechanics applications including engine research [73, 188],

and the advantages of dimensionality reduction over the EM in particular have been recently reported [54, 106].

3.3.1 Singular value decomposition

The roots of dimensionality reduction can be traced back to the early development of key matrix decompositions such as the singular value decomposition (SVD) in the late 1800s [189], with the SVD forming the foundation of many dimensionality reduction techniques used today. The objective of the SVD is to find a minimal number of basis functions that preserve as much of the variance in the original dataset as possible. By capturing more variance, the basis functions can describe more complex behaviours present in the data. The SVD method is outlined below, following Taira *et al.* [190].

Consider a square matrix $\mathbf{P}$ with N rows and columns, eigenvectors $\mathbf{q}$, and eigenvalues λ that satisfy the following equation:

$$\mathbf{P}\mathbf{q} = \lambda\mathbf{q}. \tag{3.4}$$

This equation states that pre-multiplying a vector by the matrix $\mathbf{P}$ has the same effect as multiplying that vector by a scalar λ, indicating a purely stretching process. If $\mathbf{P}$ has N linearly independent eigenvectors $\mathbf{q}_i$ with eigenvalues λ_i, the matrix equation is:

$$\mathbf{P}\,\mathbf{Q} = \mathbf{Q}\Lambda \tag{3.5}$$

or:

$$\mathbf{P} = \mathbf{Q}\Lambda\mathbf{Q}^{-1} \tag{3.6}$$

where $\mathbf{Q} = [\mathbf{q}_1\ \mathbf{q}_2\ \ldots\ \mathbf{q}_N]$ and $\Lambda = diag\,(\lambda_1, \lambda_2, \ldots, \lambda_N)$. Equation 3.6 is called the

eigenvalue decomposition. As the matrix $\mathbf{P}$ is square, a single set of basis vectors (the eigenvectors, or eigenbasis) can reproduce the entirety of both the column and the row spaces*. However, in the case of rectangular matrices, the number of rows and columns will be different, so it is no longer possible for both the row space and column space to be represented by a single set of basis vectors.

For a rectangular $N \times M$ data matrix $\mathbf{D}$, the singular value decomposition (SVD) is used to find an orthonormal (orthogonal and normalised) basis in the row space of $\mathbf{D}$ that maps onto an orthonormal basis in the column space, in order to diagonalise the original data matrix $\mathbf{D}$. Diagonalisation is a useful process that will later be shown to facilitate the separation of time- and space-dependent components of a dataset.

The objective is therefore to find a set of vectors such that a linear transformation $\mathbf{D}$ maps a unit vector in the row space $\mathbf{r}_1$ into some multiple σ_1 of a unit vector in the column space $\mathbf{l}_1$, ie. $\mathbf{D}\,\mathbf{r}_1 = \sigma_1\,\mathbf{l}_1$. In matrix form, we have $\mathbf{D}\,\mathbf{R} = \mathbf{L}\,\Sigma$. As $\mathbf{R}$ and $\mathbf{L}$ are orthogonal vectors, their inverses are equal to their transposes, so that:

$$\mathbf{D} = \mathbf{L}\,\Sigma\,\mathbf{R}^{\mathrm{T}}. \tag{3.7}$$

Equation 3.7 is the SVD, and it has a number of useful interpretations. Σ is a diagonal matrix with entries that are called the singular values. The singular values σ_i can be shown to be the square roots of the eigenvalues λ_i of both the $N \times N$ row-wise correlation matrix $\mathbf{D}\mathbf{D}^{\mathrm{T}}$ and the $M \times M$ column-wise correlation matrix $\mathbf{D}^{\mathrm{T}}\mathbf{D}$:

$$\mathbf{D}\mathbf{D}^{\mathrm{T}} = \left(\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}}\right)\left(\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}}\right)^{\mathrm{T}} = \mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}}\mathbf{R}\Sigma\mathbf{L}^{\mathrm{T}} = \mathbf{L}\Sigma^2\mathbf{L}^{\mathrm{T}} \tag{3.8}$$

*Column and row spaces are formed by all possible linear combinations of the columns and rows of a matrix respectively, and a basis is the smallest set of independent vectors that can form linear combinations that reproduce the entire vector space.

$$\mathbf{D}\mathbf{D}^{\mathrm{T}}\,\mathbf{L} = \mathbf{L}\,\Sigma^2 = \mathbf{L}\,\Lambda \tag{3.9}$$

$$\mathbf{D}^{\mathrm{T}}\mathbf{D} = \left(\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}}\right)^{\mathrm{T}}\left(\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}}\right) = \mathbf{R}\Sigma\mathbf{L}^{\mathrm{T}}\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}} = \mathbf{R}\Sigma^2\mathbf{R}^{\mathrm{T}} \tag{3.10}$$

$$\mathbf{D}^{\mathrm{T}}\mathbf{D}\,\mathbf{R} = \mathbf{R}\,\Sigma^2 = \mathbf{R}\,\Lambda. \tag{3.11}$$

In addition, the columns of $\mathbf{L}$ are called the left-singular vectors, and are the eigenvectors of the $N \times N$ row-wise correlation matrix $\mathbf{D}\mathbf{D}^{\mathrm{T}}$, while the columns of $\mathbf{R}$ are called the right-singular vectors, and are the eigenvectors of the $M \times M$ column-wise correlation matrix $\mathbf{D}^{\mathrm{T}}\mathbf{D}$. If N represents the number of variables and M represents the number of observations in the data matrix $\mathbf{D}$, then $\mathbf{D}\mathbf{D}^{\mathrm{T}}$ represents the correlation between the variables, and $\mathbf{D}^{\mathrm{T}}\mathbf{D}$ represents the correlation between the observations. Referring back to Equation 3.7, each column of $\mathbf{L}$ can then be interpreted as a set of variables constructed as an 'eigenfield', scaled by the entries of Σ, with the columns of $\mathbf{R}^{\mathrm{T}}$ giving the correct mixtures of $\mathbf{L}$ and Σ to reconstruct the original data matrix $\mathbf{D}$. This relation to the covariance matrices is how the SVD quantifies the amount of variance that is preserved in the original dataset, with the interpretation that structures that contribute to the most amount of variance are the most important ones. If the dataset comprises of velocity vectors, then the variance matrices have units $\mathrm{m}^2/\mathrm{s}^2$ and are therefore connected to the specific kinetic energy of the flow, such that the structures maximising the variance also contain the most amount of energy [69].

3.3.2 Proper orthogonal decomposition

The proper orthogonal decomposition (POD) utilises the SVD in order to reconstruct low-dimensional representations of a set of flow fields. Following the example of Brunton and Kutz [121], we return to the rectangular $N \times M$ data matrix $\mathbf{D}$ and calculate its SVD, given by Equation 3.7. After computing the SVD, individual columns of the singular vector matrices can be multiplied together to form 'POD

components' $\mathbf{l}_{:,m} \times \sigma_{m,m} \times (\mathbf{r}_{:,m}{}^{\mathrm{T}})$. Combinations of POD components are then used to create the POD-reconstructed flow fields, which can be directly compared to the original PIV snapshots. The columns of the left singular matrix $\mathbf{L}$ are ordered by the amount of variance they capture, and are known as the POD modes. They therefore represent an optimal hierarchy, and the matrix $\mathbf{D}$ can be approximated by retaining the first few dominant columns of $\mathbf{L}$ and $\mathbf{R}$ along with the corresponding singular values. For a reduced rank p such that $p < min(N, M)$, we define the truncated SVD:

$$\mathbf{D} \approx \tilde{\mathbf{L}}\tilde{\Sigma}\tilde{\mathbf{R}}^{\mathrm{T}} \tag{3.12}$$

where $\tilde{\mathbf{L}}$ now has dimensions $N \times p$, $\tilde{\Sigma}$ is now $p \times p$, and $\tilde{\mathbf{R}}^{\mathrm{T}}$ is $p \times M$. The truncated SVD is therefore a subset of the full decomposition, consisting of the p most energetic modes. An interpretation of the truncated SVD is that it represents the dominant, coherent structures in a fluid flow characterised by the most amount of energy [64]. The discarded higher-order POD modes can be associated with measurement noise [120, 191], smaller-scale turbulence or random Gaussian fluctuations [65]. It should be noted that the choice of p can be subjective, and remains a point of discussion in the literature [188]. The truncated SVD can be used to reconstruct flow fields consisting of the first few POD components, representing the most dominant flow structures. For example, the combination of the first five dominant POD components is dubbed as the fifth-order reconstruction in this thesis.

3.3.3 Gappy proper orthogonal decomposition

The POD can be leveraged for the reconstruction of data inside gaps via the gappy proper orthogonal decomposition (GPOD), first proposed by Everson and Sirovich [114]. The general principle is to initialise the gaps in the data with an initial guess, typically the ensemble mean. These guesses are then iteratively updated by computing

POD-based reconstructions of the flow with an incrementing number of modes until a convergence criterion is reached. The algorithm is iterative in nature because the optimal number of POD modes to be included in the reconstructions is not known in advance, and the inclusion of too few or too many POD modes is akin to under- or over-fitting the data [140]. Several improvements have been made to the original GPOD algorithm, mostly regarding the criterion for deciding whether to replace the an old guess with a new one [115, 116]. The most recent and best-performing method is the median filter GPOD (GPOD-MF) introduced by Saini *et al.* [117], and this is the technique used in this thesis.

In GPOD-MF the following algorithm is applied, consisting of a nested loop of iterations until convergence, with sub iterations at a fixed number of POD modes indexed by j and main iterations where the number of POD modes is incremented given by n up to a maximum total of N. The measurement field is denoted as $\phi(\mathbf{x}, t)$, defined by spatial measurement vectors $\mathbf{x}$ taken at discrete points in time t_k at the kth timestep. The number of POD modes starts at two and then increments one at a time until main loop convergence [117].

1. To initialise the algorithm, all gaps in the input field are replaced by the ensemble mean at the corresponding spatial location, to give the filled field $\tilde{\phi}_{0,0}$:

$$\tilde{\phi}_{0,0}\left(\mathbf{x}, t_k\right) = \begin{cases} \phi\left(\mathbf{x}, t_k\right) & \mathbf{x} \in \mathbf{x}_{\mathrm{d}}|_{t_k} \\ \bar{\phi}(\mathbf{x}) & \mathbf{x} \in \mathbf{x}_{\mathrm{g}}|_{t_k} \end{cases}$$

   where $\mathbf{x}_{\mathrm{d}}$ and $\mathbf{x}_{\mathrm{g}}$ represent the locations of data points and gaps respectively, and $\bar{\phi}(\mathbf{x})$ is the ensemble mean. Note that the algorithm cannot initialise if a gap exists in all snapshots.

2. At a given main iteration, POD is performed on $\tilde{\phi}_{n,j}$ using n modes, resulting in a POD-reconstructed approximation $\breve{\breve{\phi}}$:

$$\breve{\breve{\phi}}_{n,j} = \sum_{i=1}^{n} l_i \sigma_i r_i$$

where l_i, σ_i and r_i indicate the ith left-singluar vector, singular value, and right-singular vector as before.

3. The gap locations in the filled field are then updated with values from the POD approximation.

$$\tilde{\phi}_{n,j+1} = \begin{cases} \phi & \mathbf{x} \in \mathbf{x}_{\mathrm{d}}|_{t_k} \\ \breve{\breve{\phi}}_{n,j} & \mathbf{x} \in \mathbf{x}_{\mathrm{g}}|_{t_k} \end{cases}$$

4. Steps (2−3) are repeated until the POD eigenvalues computed in step 2 converge to within a user-defined tolerance. The final approximation at main iteration n is retained as $\hat{\phi}_n$.

5. A median filter (MF) is then applied as an outlier detection technique in order to adaptively retain promising guesses and revert poor guesses to their previous values. The MF selection is implemented by calculating the residual R of a centre pixel $\mathbf{x}_{\mathrm{cp}}$ in a neighborhood of adjacent pixels $\mathbf{x}_{ad}$.

$$R(\phi)|_{\mathbf{x}_{\mathrm{cp}},t_k} = \mathrm{abs}\left(\phi\left(\mathbf{x}_{\mathrm{cp}}, t_k\right) - \mathrm{median}\left(\phi\left(\mathbf{x}_{ad}, t_k\right)\right)\right)$$

The updated guesses at each pixel are retained if the residual at that pixel is

reduced relative to the previous main iteration, and reverted otherwise:

$$\hat{\phi}_n\left(\mathbf{x}, t_k\right) = \begin{cases} \hat{\phi}_{n-1}\left(\mathbf{x}_{\mathrm{cp}}, t_k\right), & \text{if } R_{n-1}|_{\mathbf{x}_{\mathrm{cp}},t_k} \leq R_n|_{\mathbf{x}_{\mathrm{cp}},t_k} \\ \hat{\phi}_n\left(\mathbf{x}_{\mathrm{cp}}, t_k\right), & \text{if } R_{n-1}|_{\mathbf{x}_{\mathrm{cp}},t_k} > R_n|_{\mathbf{x}_{\mathrm{cp}},t_k} \end{cases}$$

and the next main iteration begins with $\tilde{\phi}_{n+1,0} = \hat{\phi}_n$.

6. The algorithm increments the main iterations n until a main convergence criterion is satisfied, as discussed below.

In practical scenarios, the true values of the data in the edge gaps would not be known, so the accuracy of the GPOD reconstructions cannot be assessed directly. A convergence criterion is therefore needed in order to prevent over-fitting and terminate the algorithm at a number of modes that is close to the true optimal value. A commonly-used criterion was introduced by Gunes *et al.* [115] known as convergence checking (CC) gaps. With this method, additional gaps are added to the input data in locations where the true values are known. The reconstruction error of the GPOD reconstructions inside these CC gaps can then be calculated and tracked as a proxy for the true reconstruction errors inside the real gaps. The GPOD algorithm is terminated when the CC reconstruction errors are calculated to have reached a minimum.

3.3.4 Spectral proper orthogonal decomposition

As outlined by Taira et al. [190], when taking POD in its standard implementation (taken to be the space-only phase-dependent form), each POD mode is only spatially coherent. This means that each mode can contain a mixture of different frequencies, clouding their physical interpretation in some circumstances. Towne et al. [192] de-

tailed that in standard POD, time is taken as a stochastic parameter, where instances in time represent snapshots of measurements in an ensemble. Therefore, all sense of ordering between snapshots is lost, and POD modes cannot capture any temporal correlations in the data.

Spectral POD (SPOD) is an alternative implementation of POD that dates back to Lumley's original work [95], not to be confused with the technique introduced by Sieber *et al.* under the same name [193]. Unlike standard POD, SPOD is able to capture flow structures that evolve coherently in both time and space [190]. This is done by conducting a Fourier transform on the data and then performing the singular value decomposition (SVD) on the cross-spectral density matrix to yield SPOD modes that are associated with specific frequencies. If $\widehat{\mathbf{D}}_\omega$ is the Fourier transform of the data matrix $\mathbf{D}$ at a specific frequency ω, then the cross-spectral density matrix is:

$$\widehat{\mathbf{D}}_\omega \widehat{\mathbf{D}}_\omega^{\mathrm{T}} \psi_\omega = \lambda_\omega \psi_\omega \tag{3.13}$$

with spectral modes ψ_ω and eigenvalues λ_ω.

Like the POD, SPOD modes are orthogonal, which is useful in hierarchically ordering the modes. Orthogonality ensures that a maximal amount of variance is captured with a minimal number of modes, as each new mode describes a unique variation in the data. However, orthogonality can be a limitation when trying to capture phenomena where the principal directions do not correlate with the most dynamically important directions [194–196]. In the results chapters of this thesis, the turbulent flow motion during the intake stroke is shown to fluctuate in a fashion reminiscent of a bi-modal distribution. In this case, the intake jet has a tendency to point to the left or the right rather than straight downwards; both left and right directions are important in

describing the dynamics, so it is less justified to enforce orthogonality of the POD modes to a single principal direction.

3.3.5 Dynamic mode decomposition

Dynamic mode decomposition (DMD) is a dimensionality reduction technique that produces spatio-temporally coherent modes without enforcing orthogonality, developed by Schmid [99]. DMD characterises the time dynamics by finding the best-fit linear operator that maps one snapshot of data onto the next, producing modes of structures that oscillate together and grow or decay in time [103]. Following Lumley [95] and Futrzynski [197], a fluid quantity $\phi(x,t)$ that is dependent on both space (x) and time (t) may be decomposed into a set of new basis functions via a separation of variables:

$$\phi(x,t) = \sum_j \psi_j(x)\alpha_j(t) \tag{3.14}$$

where $\psi_j(x)$ represent the spatial modes, and $\alpha_j(t)$ are the expansion coefficients in time. To find structures that oscillate at set frequencies, $\alpha_j(t)$ is expressed as a Fourier series:

$$\alpha_j(t) = a_j e^{i(\omega_j t + \theta_j)} \tag{3.15}$$

where a_j is the magnitude of $\alpha_j(t)$, ω_j is the frequency, and θ_j is the phase. For the analysis of experimental measurements, this expansion is more useful in discrete-time format. For a constant time-step (Δt) between measurements, we therefore have:

$$\begin{aligned}
\alpha_j(t_k) &= a_j e^{i\theta_j} e^{i\omega_j t_k} && \left[\gamma_j := a_j e^{i\theta_j}\right] \\
&= \gamma_j e^{i\omega_j k\Delta t} \\
&= \gamma_j \left(e^{i\omega_j \Delta t}\right)^k && [\psi_j := \omega_j \Delta t] \\
&= \gamma_j \left(e^{i\psi_j}\right)^k && \left[\lambda_j := e^{i\psi_j}\right] \\
&= \gamma_j \lambda_j^k.
\end{aligned} \tag{3.16}$$

Returning to Equation 3.14:

$$\phi(x, t_k) = \sum_j \psi_j(\gamma_j \lambda_j^k). \tag{3.17}$$

Marching this equation forward in time gives:

$$\phi(x, t_{k+1}) = \sum_j \psi_j(\gamma_j \lambda_j^{k+1}). \tag{3.18}$$

Establishing the relationship between Equations 3.17 and 3.18 in matrix form:

$$\begin{aligned}
\phi_{k+1} &= \underbrace{\begin{bmatrix} | & | & \\ \psi_1 & \psi_2 & \dots \\ | & | & \end{bmatrix}}_{\Psi} \underbrace{\begin{bmatrix} \gamma_1 \lambda_1^{k+1} \\ \gamma_2 \lambda_2^{k+1} \\ | \end{bmatrix}}_{\boldsymbol{C}_{k+1}} \\
&= \Psi \underbrace{\begin{bmatrix} \ddots & & 0 \\ & \lambda_j & \\ 0 & & \ddots \end{bmatrix}}_{\Lambda} \boldsymbol{C}_k \\
&= \Psi \Lambda \boldsymbol{C}_k \\
&= \Psi \Lambda \Psi^{-1} \Psi \boldsymbol{C}_k \\
&= \Psi \Lambda \Psi^{-1} \phi_k.
\end{aligned} \tag{3.19}$$

Hence there exists a matrix $\mathbf{A}$, for which $\Psi \Lambda \Psi^{-1}$ is its eigendecomposition, that advances a set of snapshots ϕ_k forward in time to ϕ_{k+1}, for:

$$\phi_{k+1} = \mathbf{A} \phi_k. \tag{3.20}$$

The matrix $\mathbf{A}$ is therefore a linear operator that controls the temporal evolution of the data; its eigenvectors (ψ_j) represent coherent structures that evolve in time

according to a frequency (ω_j) and a growth/decay rate given by its eigenvalues (λ_j). The objective of DMD is to model a dynamical system by optimally calculating the eigendecomposition of $\mathbf{A}$. To this end, the problem is constructed as follows. The ensemble of ϕ_{k+1} and ϕ_k vectors can be represented as two $N \times (M-1)$ data matrices Φ' and Φ respectively, where Φ' and Φ are simply composed of the last and the first $(M-1)$ columns of the full data matrix $\mathbf{D}$:

$$\Phi' = \mathbf{D}_{[:,2:M]} = \begin{bmatrix} | & | & & | \\ \phi_2 & \phi_3 & \cdots & \phi_M \\ | & | & & | \end{bmatrix} \tag{3.21a}$$

$$\Phi = \mathbf{D}_{[:,1:(M-1)]} = \begin{bmatrix} | & | & & | \\ \phi_1 & \phi_2 & \cdots & \phi_{M-1} \\ | & | & & | \end{bmatrix}. \tag{3.21b}$$

With the data matrices established, the linear operator can be approximated as:

$$\mathbf{A} \approx \Phi' \Phi^{\dagger} \tag{3.22}$$

where † denotes the Moore-Penrose pseudo-inverse for the non-square matrix Φ. For an over-determined system with more rows than columns in the Φ matrix, the Moore-Penrose pseudo-inverse can be interpreted as a least squares regression algorithm that minimises $\|\Phi' - \mathbf{A}\Phi\|_F$, where $\mathbf{A}$ gives the best-fit slope mapping the data from Φ to Φ' in the linear system. The subscript $_F$ indicates the Frobenius norm, given by:

$$\|\Phi\|_F = \sqrt{\sum_{j=1}^{n} \sum_{k=1}^{m} \phi_{jk}^2}. \tag{3.23}$$

To aid computational efficiency and also reduce the sensitivity to noise, the calculation

of Equation 3.22 begins by taking the rank-reduced SVDs of Φ and Φ':

$$\Phi \approx \tilde{\mathbf{L}}\tilde{\Sigma}\tilde{\mathbf{R}}^{\mathrm{T}} \tag{3.24a}$$

$$\Phi' \approx \mathbf{A}\tilde{\mathbf{L}}\tilde{\Sigma}\tilde{\mathbf{R}}^{\mathrm{T}} \tag{3.24b}$$

truncated at a rank p, where $\tilde{\mathbf{L}}$ is the $N \times p$ matrix containing the leading left-singular vectors of Φ, $\tilde{\Sigma}$ is the diagonal $p \times p$ matrix containing the largest p singular values of Φ, and $\tilde{\mathbf{R}}$ is the $(M-1) \times p$ matrix containing the leading right-singular vectors of Φ.

The next stage of the DMD algorithm makes use of Equations 3.22 and 3.24a to give the linear operator in terms of Φ':

$$\mathbf{A} \approx \Phi'\tilde{\mathbf{R}}\tilde{\Sigma}^{-1}\tilde{\mathbf{L}}^{\mathrm{T}} \tag{3.25}$$

noting that both $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{R}}$ are unitary matrices, meaning that $\tilde{\mathbf{L}}^{\mathrm{T}}\tilde{\mathbf{L}} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^{\mathrm{T}} = \tilde{\mathbf{L}}\tilde{\mathbf{L}}^{-1} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Rather than calculating the full $N \times N$ matrix $\mathbf{A}$, a reduced $p \times p$ similar matrix [†] $\tilde{\mathbf{A}}$ can be found by projecting $\mathbf{A}$ onto the left-singular vectors:

$$\tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{\mathrm{T}}\mathbf{A}\tilde{\mathbf{L}} = \tilde{\mathbf{L}}^{\mathrm{T}}\Phi'\tilde{\mathbf{R}}\tilde{\Sigma}^{-1}. \tag{3.26}$$

The eigendecomposition of $\tilde{\mathbf{A}}$ is defined as:

$$\tilde{\mathbf{A}}\psi = \psi\lambda. \tag{3.27}$$

As the matrices $\mathbf{A}$ and $\tilde{\mathbf{A}}$ are similar, they have the same eigenvalues λ. They do

[†]Similar matrices preserve the same vector addition and scalar multiplication operations under a change of basis. In general, two matrices $\mathbf{A}$ and $\mathbf{B}$ are defined as similar if the matrix $\mathbf{C}$ exists, for $\mathbf{A} = \mathbf{C}^{-1}\mathbf{B}\mathbf{C}$.

not, however, have the same eigenvectors, so the DMD modes are defined as ξ:

$$\xi \coloneqq \frac{1}{\lambda}\mathbf{B}\psi \tag{3.28}$$

for

$$\mathbf{B} \coloneqq \Phi'\mathbf{R}\Sigma^{-1}. \tag{3.29}$$

This definition ensures that the exact eigenvectors of the matrix $\mathbf{A}$ are recovered, with the proof given in Tu et al. [198]:

$$\begin{aligned}
\mathbf{A} &= \mathbf{B}\tilde{\mathbf{L}}^{\mathrm{T}} && \text{from} \quad \Phi' = \mathbf{A}\mathbf{L}\Sigma\mathbf{R}^{\mathrm{T}} \\
\tilde{\mathbf{A}} &= \tilde{\mathbf{L}}^{\mathrm{T}}\mathbf{B} && \text{from} \quad \tilde{\mathbf{A}} = \tilde{\mathbf{L}}^{\mathrm{T}}\mathbf{A}\tilde{\mathbf{L}} \\
\mathbf{A}\psi &= \tfrac{1}{\lambda}\mathbf{B}\tilde{\mathbf{L}}^{\mathrm{T}}\mathbf{B}\psi \\
&= \tfrac{1}{\lambda}\mathbf{B}\tilde{\mathbf{A}}\psi \\
&= \mathbf{B}\psi && \text{from} \quad \text{Equation 3.27} \\
&= \xi\lambda.
\end{aligned} \tag{3.30}$$

Finally, the solution to the system 3.20 may be expressed in the eigenvector basis:

$$\phi_{k+1} \approx \sum_{m=1}^{p} \xi_m \lambda_m^k b_m = \mathbf{\Xi}\Lambda^k\mathbf{b} \tag{3.31}$$

where the matrix $\mathbf{\Xi}$ has columns containing ξ_m the eigenvectors of $\mathbf{A}$, Λ is a diagonal matrix containing λ_m the eigenvalues of $\mathbf{A}$, and $\mathbf{b}$ is a vector consisting of the mode amplitudes b_m.

By considering the initial conditions with $k = 0$, $\mathbf{b}$ can be calculated from $\mathbf{b} = \mathbf{\Xi}^{\dagger}\phi_1$, where ϕ_1 is the first snapshot of data. This definition of mode amplitudes relies

heavily on the assumption that snapshots evolve linearly in time from the initial condition, which may be only approximately valid for experimental measurements of non-linear flows where there can be significant CCVs and measurement noise [103, 194]. Outlier flow structures and anomalous measurements may suddenly appear in one snapshot but be absent from the other measurements, causing very high decay rates and large amplitudes [194]. High-amplitude modes defined in this way therefore do not necessarily contribute to the full time series of measurements or reflect the most important flow dynamics.

3.3.6 Sparsity-promoting dynamic mode decomposition

Sparsity-promoting dynamic mode decomposition (SPDMD) is a DMD variant that provides a robust method of defining the mode amplitudes, introduced by Jovanovic *et al.* [196]. Here, an optimal set of amplitudes is calculated by finding the DMD modes that have the highest contribution to the dynamics across the full dataset. This is cast as an optimisation problem, where the fewest number of modes that produce the smallest reconstruction error is sought. Mathematically, this is achieved by minimising the sum of the reconstruction error and the sparsity of the solution:

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \;\; J(\boldsymbol{\alpha}) + \gamma \sum_{i=1}^{m} |\boldsymbol{\alpha}_i| \,, \tag{3.32}$$

where $\boldsymbol{\alpha}$ is the unknown vector of amplitudes, $J(\boldsymbol{\alpha})$ is an objective function representing the reconstruction error, for which the derivation can be found in Ref. [196], γ is a user-defined regularisation parameter that controls the trade-off between minimising reconstruction error and promoting sparsity, and m is the number of modes. The L_1 norm denoted by $|\cdot|$ is the sum of the absolute values of the coefficients, which promotes sparsity in the solution when minimised [196]. Therefore, Equation 3.32

finds the fewest number of modes that retain the most amount of information in the dataset.

Equation 3.32 is the SPDMD problem definition, which can be solved using standard optimisation routines, such as the method of Lagrange multipliers [199]. As a consequence of the Lagrange multiplier theorem [200], for a stationary point of an objective function subject to equality constraints, the gradient of the function can be expressed as a linear combination of the gradients of the constraints at that point, and the coefficients are known as the Lagrange multipliers [200]. Using this relationship, the constraints can be incorporated into the objective function, formulating the Lagrangian function, which can be optimised directly. For example, consider the optimisation problem:

$$\min f(x), \qquad \text{subject to } h(x) = 0. \tag{3.33}$$

The Lagrangian is given as:

$$\mathcal{L}(x, \lambda_L) = f(x) + \lambda_L^T h(x) \tag{3.34}$$

where λ_L is the vector of Lagrange multipliers. An additional penalty term is often introduced to ensure the enforcement of the constraints $h(x)$, aiding in the stability of the optimisation and speed of the convergence, leading to the Augmented Lagrangian [201]:

$$\mathcal{L}_{\text{aug}}(x, \lambda_L) = f(x) + \lambda_L^T h(x) + \frac{C_{pen}}{2} \|h(x)\|_2^2 \tag{3.35}$$

where C_{pen} is a parameter that controls the size of the penalty to be applied. For multi-objective problems, instead of attempting to solve the full problem simultaneously, a more efficient technique can be employed that considers each part of the

optimisation problem separately, known as the alternating direction method of multipliers (ADMM) [202]. In order to facilitate use of the ADMM for the SPDMD problem statement in Equation 3.32, a separation of variables is introduced, which is achieved with the dummy variable $\boldsymbol{\beta}$:

$$\underset{\boldsymbol{\alpha}}{\text{minimize}} \;\; J(\boldsymbol{\alpha}) + \gamma \sum_{i=1}^{m} |\boldsymbol{\beta}_i| \tag{3.36}$$

subject to the constraint $\boldsymbol{\alpha} - \boldsymbol{\beta} = 0$. Then, the augmented Lagrangian can be written as:

$$\begin{aligned} &\mathcal{L}_{\text{aug}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}, \lambda_L\right) := \\ &\qquad J(\boldsymbol{\alpha}) + \gamma \sum_{i=1}^{m} |\boldsymbol{\beta}_i| + \frac{1}{2}\left(\lambda_L^*(\boldsymbol{\alpha} - \boldsymbol{\beta}) + (\boldsymbol{\alpha} - \boldsymbol{\beta})^* \lambda_L + C_{pen} \|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2{}^2\right) \end{aligned} \tag{3.37}$$

where * denotes the complex-conjugate-transpose and $\|\cdot\|_2$ is the L_2-norm. The second Lagrange term $(\boldsymbol{\alpha} - \boldsymbol{\beta})^* \lambda_L$ is used to cancel out the imaginary parts arising from the first Lagrange term, ensuring that the Lagrangian remains a real-valued objective function; complex numbers are challenging to optimise as they cannot be ordered straightforwardly [203]. With the Lagrangian defined, the ADMM can be applied with three steps; minimise $\boldsymbol{\alpha}$ with a fixed $\boldsymbol{\beta}$; update $\boldsymbol{\alpha}$ and then minimise $\boldsymbol{\beta}$; update λ_L:

$$\begin{aligned} \boldsymbol{\alpha}^{k+1} &:= \arg\min_{\boldsymbol{\alpha}} \; \mathcal{L}_{\text{aug}}\left(\boldsymbol{\alpha}, \boldsymbol{\beta}^k, \lambda_L^k\right), \\ \boldsymbol{\beta}^{k+1} &:= \arg\min_{\boldsymbol{\beta}} \; \mathcal{L}_{\text{aug}}\left(\boldsymbol{\alpha}^{k+1}, \boldsymbol{\beta}, \lambda_L^k\right), \\ \lambda_L^{k+1} &:= \lambda_L^k + C_{pen}\left(\boldsymbol{\alpha}^{k+1} - \boldsymbol{\beta}^{k+1}\right). \end{aligned} \tag{3.38}$$

In this way, ADMM provides an efficient solution to Equation 3.36 by solving each optimisation problem separately, switching between minimising the reconstruction error (min $\boldsymbol{\alpha}$) and maximising the sparsity (min $\boldsymbol{\beta}$). The algorithm begins with an

initial point $(\boldsymbol{\beta}^0, \lambda_L^0)$ and iterates until two user-defined tolerances ϵ_1 and ϵ_2 are reached:

$$\left\|\boldsymbol{\alpha}^{k+1} - \boldsymbol{\beta}^{k+1}\right\|_2 \leq \epsilon_1 \text{ and } \left\|\boldsymbol{\beta}^{k+1} - \boldsymbol{\beta}^{k}\right\|_2 \leq \epsilon_2 \,. \tag{3.39}$$

Finally, the value of the γ parameter in Equation 3.36 must be determined by the user, with the recommended method being to compare the reconstruction errors over a computational sweep of different γ values [194, 196]. The reconstruction error is quantified as a performance loss:

$$\%\,\Pi_{loss} = 100 \times \frac{\|\Phi - \tilde{\mathbf{L}}\tilde{\Sigma}\tilde{\mathbf{R}}^{\mathrm{T}}\|_F}{\|\Phi\|_F}. \tag{3.40}$$

With a value of γ chosen, the vector of amplitudes given by SPDMD is fixed, and the flow fields can be re-constructed using combinations of different modes.

3.3.7 Neural networks

Neural networks are a class of universal function approximator that have been gaining significant traction in the recent fluid mechanics literature [204, 205]. In a breakthrough for dimensionality reduction, Baldi and Hornik [206] showed that a specific type of neural network known as an autoencoder (discussed in §3.3.9) can be viewed as a non-linear generalisation of the principal component analysis (known as the POD in fluid mechanics applications [121]). This is a powerful finding that has since allowed some of the capabilities of linear SVD-based dimensionality reduction approaches to be extended and applied to a wider range of engineering challenges [71].

A neural network consists of a stack of connected layers, with each layer containing a series of basic computation units known as neurons. Each neuron receives input data, performs a linear operation on the data using a set of weights and biases (that are

learned over time), and passes the result through an activation function to produce an output. The activation functions are the mechanisms for introducing non-linearity into the model, allowing the network to capture complex patterns and relationships in the data. Some of the most common activation functions are given by [121]:

$$\begin{aligned} f(x) &= x && - \text{ linear} \\ f(x) &= \frac{1}{1+\exp(-x)} && - \text{ logistic (sigmoid)} \\ f(x) &= \tanh(x) && - \text{ TanH} \\ f(x) &= \begin{cases} 0 & x \leq 0 \\ x & x > 0 \end{cases} && - \text{ rectified linear unit (ReLU).} \end{aligned} \tag{3.41}$$

Although occasionally useful for interpretability, linear activation functions limit the types of relationships that can be learned by the model [207]. Sigmoid and tanh loss functions are similar S-curve functions but with different ranges, with sigmoid varying between 0 and 1 while tanh varies between -1 and 1. These functions are useful for classification problems where binary decision boundaries are desired [208]. The steeper gradients of the tanh function means that it is sometimes able to achieve faster learning than the sigmoid [207]. ReLU is suitable for regression tasks, and the zeroing of negative values can be advantageous for computational efficiency, as well as promoting sparsity which helps to mitigate overfitting [207, 209].

A generic architecture for a neural network is shown in Figure 3.1. Mathematically, for a given input $\mathbf{x}$, the output $\mathbf{y}$ is obtained by successfully passing the data through

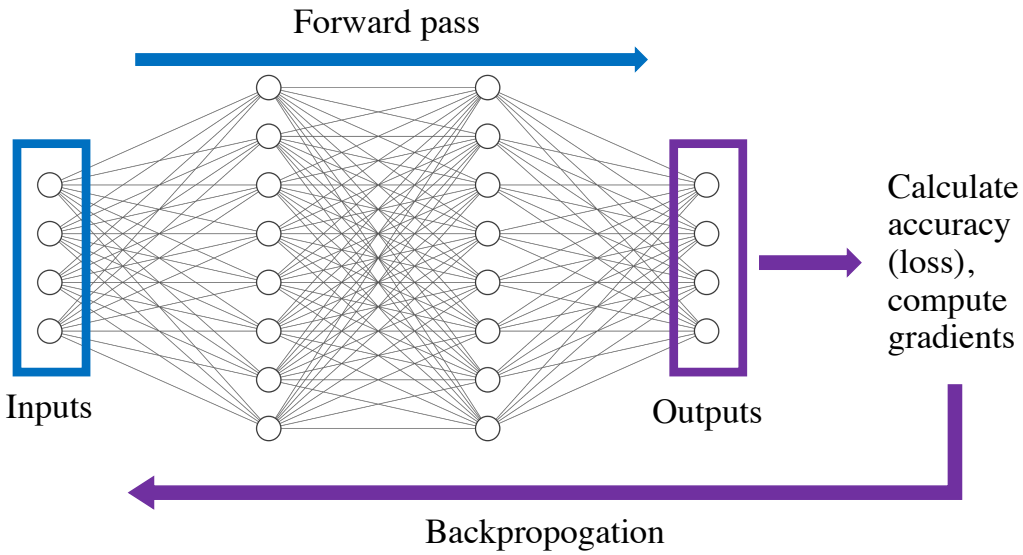

Figure 3.1: An example of a generic neural network model. The input data is fed through the network in the forward pass to produce a predicted output. The loss between the output and the original data is calculated, and the weights in the network are altered according to the gradients of the loss terms via backpropagation.

the series of layers as:

$$\mathbf{y} = f_M\left(\mathbf{W}_M, \cdots, f_2\left(\mathbf{W}_2, f_1\left(\mathbf{W}_1, \mathbf{x}\right)\right)\cdots\right) \tag{3.42}$$

where $\mathbf{W}_M$ and f_M are the matrix of weights and the activation function for the Mth layer respectively. The weight matrices are iteratively updated and optimised as the model learns patterns from a set of training data. This optimisation procedure is performed using backpropagation. After a forward pass through the network, the similarity of the output to the desired ground truth is calculated with a user-defined loss function; the mean-squared error is a typical choice for regression tasks. Consider a single node, single layer network such that:

$$\mathbf{y} = g(f(\mathbf{x}, a), b) = g(\mathbf{z}, b) \tag{3.43}$$

with weighting constants a and b. The loss might be calculated as a mean square

error (MSE):

$$\text{MSE} = \frac{1}{2}(\mathbf{y_0} - \mathbf{y})^2 \tag{3.44}$$

where $\mathbf{y_0}$ is the ground truth. Backpropagation minimises the loss via an iterative, gradient descent update rule:

$$\begin{aligned} a_{k+1} &= a_k + \delta \frac{\partial E}{\partial a_k} \\ b_{k+1} &= b_k + \delta \frac{\partial E}{\partial b_k} \end{aligned} \tag{3.45}$$

where δ is a user-defined learning rate, and the rates of change of error E with the parameters can be found via the chain rule:

$$\frac{\partial E}{\partial a} = \frac{d\mathbf{y}}{da} \times \frac{dE}{d\mathbf{y}} = (\mathbf{y_0} - \mathbf{y}) \frac{d\mathbf{y}}{d\mathbf{z}} \frac{d\mathbf{z}}{da} = 0. \tag{3.46}$$

Consider the linear activation function

$$f(c, \alpha) = g(c, \alpha) = c \times \alpha \tag{3.47}$$

then:

$$\begin{aligned} z &= ax \\ y &= bz \end{aligned} \tag{3.48}$$

and the gradients can be computed as:

$$\begin{aligned} \frac{\partial E}{\partial a} &= (\mathbf{y_0} - \mathbf{y}) \frac{d\mathbf{y}}{d\mathbf{z}} \frac{d\mathbf{z}}{da} = (\mathbf{y_0} - \mathbf{y}) \cdot b \cdot \mathbf{x} \\ \frac{\partial E}{\partial b} &= (\mathbf{y_0} - \mathbf{y}) \frac{d\mathbf{y}}{db} = (\mathbf{y_0} - \mathbf{y}) \, \mathbf{z} = (\mathbf{y_0} - \mathbf{y}) \cdot a \cdot \mathbf{x}. \end{aligned} \tag{3.49}$$

For a network with M layers, this can be generalised as:

$$\frac{\partial E}{\partial a} = (\mathbf{y_0} - \mathbf{y}) \frac{d\mathbf{y}}{d\mathbf{z}_m} \frac{d\mathbf{z}_m}{d\mathbf{z}_{m-1}} \cdots \frac{d\mathbf{z}_2}{d\mathbf{z}_1} \frac{d\mathbf{z}_1}{da}. \tag{3.50}$$

The backpropagation algorithm is run until the network reaches convergence, which typically determined by monitoring the errors over time.

3.3.8 Convolutional neural networks

Convolutional Neural Networks (CNNs) are a class of deep neural networks specifically designed for processing structured grid data, such as images [204]. Therefore, they are well-suited for analysing PIV flow fields which can have velocity measurements at regularly-spaced pixels [71]. In CNNs, features can be extracted from data using convolutional layers, in which the input data are convolved with a learnable filter matrix of a user-defined fixed shape. The convolution operation is conducted by sliding the filter along each pixel in the input data and performing element-wise multiplication followed by a summation [210]. This is denoted mathematically as:

$$\text{conv}(\phi, F)_{i,j} = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} \phi(i+m, j+n) \cdot F(m,n) \tag{3.51}$$

for input data ϕ, a filter F, filter dimensions M and N, and spatial coordinates i and j. This operation produces abstracted feature maps, helping the model to extract and detect important spatial features such as edges and textures [210]. While generic fully-connected networks could be used to analyse grid-based data, CNNs learn patterns and features more efficiently and are less prone to over-fitting [211].

3.3.9 Autoencoders

As previously mentioned, autoencoders are a type of neural network that can reduce the dimension of a set of data, and can be thought of as a generalisation of linear dimensionality reduction techniques such as the POD [121]. This is done by gradually reducing the number of neurons in each layer, reaching a minimum known as a bottleneck, then decompressing the data until the original shape is returned. An example

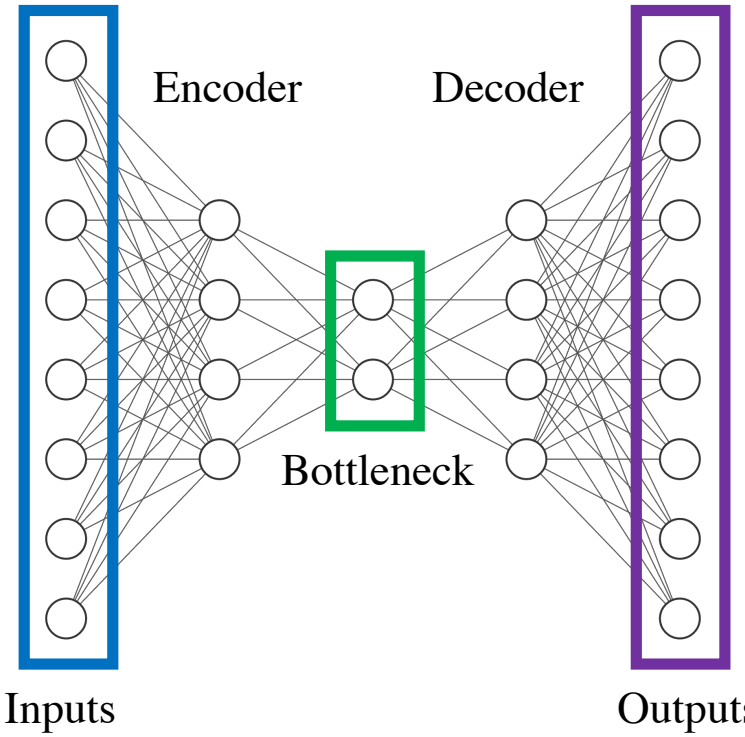

Figure 3.2: An example autoencoder neural network that can learn reduced subspaces that represent the data via the latent space (also known as the bottleneck).

shape of an autoencoder is shown in Figure 3.2. The first section of the network is known as the encoder, and extracts dominant features from the data. This function is akin to the creation of POD modes, but without the inherent orthogonality and linearity constraints. The latter section is the decoder, which builds a full state-space prediction given the dominant features presented in the bottleneck. Convolutional autoencoders are used extensively in Chapter 6 in order to extract dominant patterns from PIV data and use that information to fill large gaps in the images.

In 2015 Ronneberger *et al.* [127] introduced a ground-breaking neural network architecture based on a convolutional autoencoder design, named the UNet. The key novelty of the design was the inclusion of so-called skip-connections, which copy the outputs of each layer in the encoder and concatenate them to the inputs of each layer in the decoder. In a typical autoencoder, each contracting layer in the encoder

extracts more general and global features of the data, with the most general features being passed to the decoder through the bottleneck [212]. By concatenating the original features extracted at each layer of the encoder to the decoder layers via the skip connections, more detailed and localised features can also be preserved and considered in conjunction with the global features. Although the UNet was originally developed for medical image segmentation, this multi-scale treatment of image data has proven to be extremely effective for a range of tasks such as image denoising [213, 214], super-resolution [129, 215], and image reconstruction [216].

3.4 Summary

As discussed in §1.6, this thesis is concerned with the creation of validation targets and the enhancement of validation data. With this in mind, there are two properties of the numerical methods previously discussed that are of particular note, with summaries provided in Table 3.1. Firstly, the expressivity of these methods with regards to their ability to capture complex non-linear patterns in data is important due to the inherently non-linear characteristics of turbulent flows. All of the modal decomposition algorithms considered in this thesis rely on the SVD, which finds new coordinate transforms via linear combinations of the original variables. This is sufficient for a range of tasks, as many phenomena that are globally non-linear may be well-approximated by linearisation at a local level [121]. However, for extrapolative challenges in validation data enhancement such as predicting turbulent flow motion outside of the field of view (see Chapter 6), there are fewer nearby data points that can be used to inform the predictions, and more expressive models that can capture more complex patterns between the data points that are available might be required. Widely-used neural network architectures employ non-linear activation functions between multiple data-processing layers, enabling them to capture complex non-linear

behaviours [206].

Secondly, the mapping between the inputs and outputs, such as many-to-many or many-to-one, is important for the creation of validation targets. In this thesis, these mappings are defined with respect to the number of instances provided in the inputs and outputs. For example, a many-to-one mapping would take any number of realisations (such as 300 PIV images) as the input, and produce a single realisation as the output. Many-to-one methods are often desired in order to collapse a large dataset onto a single representative validation target, thereby simplifying the validation process. Although the POD technique produces a set of modes that could be considered one at a time, POD-based reconstructions of the flow are dependent on the specific instant in time given by the right-singular vector in question. This means that for a set of 300 PIV images from an engine, the POD-based reconstructions of these images will differ depending on which cycle is being reconstructed, resulting in 300 validation targets. This effect is demonstrated in Chapter 4, and for this reason the POD-based reconstructions are labelled as many-to-many in Table 3.1. On the other hand, the temporal decompositions (SPOD, DMD and SPDMD) produce flow reconstructions that operate at specific frequencies, with each spatio-temporal mode representing the behaviour of the snapshots across the whole dataset. There is therefore no time-dependency, and the image reconstructions can be considered either one frequency at a time or as many frequencies taken together, with this option noted in Table 3.1. Finally, neural networks are considered to be any-to-any, as their modularity means that they can be constructed to fit input and output data types of any shape.

Advantages of the POD and GPOD methods are that they are widely used and in many cases considered to be current best-practice [54, 88, 117], so they serve as useful baselines in this thesis. A drawback of all POD-based methods is that

POD-reconstructed data requires a threshold for the number of included modes to be specified, and it is challenging to find objective and broadly-applicable criteria for this threshold [106, 217]. GPOD is an exception to this, where the number of modes is incremented until convergence, but this comes at the expense of making the GPOD method computationally intensive to run, as discussed in §6. Regarding the temporal decompositions, although the orthogonality of the SPOD modes enables the calculation of mode amplitudes via the construction of optimal mode hierarchies, this orthogonality is deemed to be an unnecessary constraint for the investigations in this thesis, as the SPDMD method is able to optimally calculate the mode ampltiudes with respect to the dynamics shown in the individual flow fields, which makes SPDMD particularly suitable for the creation of validation targets as shown in Chapters 4 and 5. Lastly, drawbacks to neural network models include often being computationally intensive to train (typically requiring at least one GPU), requiring carefully considered datasets dedicated to the training process, and being prone to over-fitting due to their data-driven nature. In order to develop validation process methods for creation of validation targets that can be used for a wide range of different datasets, it would therefore be more straightforward to use the non-parametric POD-based methods. However, the expressivity of neural networks becomes a significant advantage when enhancing validation data, as shown in Chapter 6.

Table 3.1: Summary of widely-used dimensionality reduction methods and their relative strengths and weaknesses with regards to how they apply to the objectives of this thesis.

Technique	**Expressivity**	**Mapping**	**Advantages**	**Disadvantages**
POD reconstruction	Linear	Many-to-many	Widely used	Subjective mode threshold
GPOD reconstruction	Linear	Many-to-many	Widely used	Slow inference
SPOD reconstruction	Linear	Many-to-many or many-to-one	Temporal patterns	Orthogonality Subjective mode threshold
DMD reconstruction	Linear	Many-to-many or many-to-one	Temporal patterns	No amplitude hierarchy Subjective mode threshold
SPDMD reconstruction	Linear	Many-to-many or many-to-one	Robust amplitudes	Subjective mode threshold
Neural network	Non-linear	Any-to-any	Non-linear patterns Fast inference Transfer learning	Computationally intensive to train Prone to over-fitting

Chapter 4

Creating validation targets for PIV and RANS

4.1 Overview

As discussed in §1.5.1, creating validation targets by averaging ensembles of vector data can lead to a diminished magnitudes effect. Prior studies have attempted to resolve this with the use of dimensionality reduction techniques such as the POD [54] and KPCA [106]. Although these techniques are able to retain higher vector magnitudes with the use of POD-based reconstructions, these reconstructions utilise many-to-many mappings, as demonstrated later in § 4.4.4. Therefore, it is not possible to collapse the data onto a single validation target with these methods, which makes it more challenging to analyse and interpret large datasets. This chapter hypothesises that temporal modal decomposition methods are more suitable for validation target creation due to having many-to-one mappings, and investigates the use of DMD and

SPDMD to this end.

The Oxford PIV dataset introduced in §2.5.1 is used for the analysis in this chapter, which has a corresponding RANS dataset for comparison, and is subject to CCVs in the form of intake jet flapping. Intake jet flapping refers to the change in general orientation of the coherent high-velocity jet structure between different realisations (cycles) at the same phase (crank angle) [59]. It occurs when separate streams from the two intake valves collide in the engine cylinder with varying strengths, influencing the overall direction of the resultant combined jet. Jet flapping is of interest as variations in the intake jet are thought to be a major source of subsequent CCVs, contributing to abnormalities in the resulting flow fields and inhibiting engine efficiency and power output [79, 80]. For example, Zeng et al. [83] investigated the variability of the early flame kernel growth in a propane-fuelled SI engine, and concluded that variations in the macro-scale flow were a primary contributor to undesirable flame structures. More recent studies have also investigated the effects of CCVs on the resultant burning rates [82, 218, 219]. Abraham et al. [89] suggest three causes of cyclic variability in the intake jet; CCV in the intake port and pressure boundary conditions, physical oscillations of the intake valve during valve opening and closing, and variations in engine speed. The CCVs in the Oxford optical engine (in the form of intake jet flapping) are expected to make this PIV dataset a good test case for examining different validation target creation methodologies, as the diminished magnitudes effect will be made more apparent.

The remainder of this chapter is structured as follows. Firstly, two quantitative metrics are introduced in order to quantify the similarity of different vector fields. Following this, initial comparisons are made between the ensemble of PIV data and the RANS data, and the ensemble mean (EM) is evaluated regarding its ability to

fairly represent the velocity magnitudes in the individual PIV images. The diminished magnitudes effect is quantified for the first time with the use of velocity histograms. The POD method is implemented in order to explore a variety of characteristics such as its ability to retain the vector magnitudes in the POD-reconstructions, the subjectivity in the mode cut-off definition, and the many-to-many mapping. The standard DMD algorithm is then investigated, which is shown to be capable of many-to-one mappings, although spectral analyses reveal that the simplistic amplitude definition results in outcomes that are not robust across different subsets of the data. Finally, this issue is alleviated with the SPDMD, which is shown to produce modes that are significantly more representative of the vector magnitudes seen in the individual snapshots. The SPDMD is therefore proposed as the recommended method for creating validation targets from vector data.

4.2 Metrics

Two key metrics are used in this chapter to quantify the similarity of vector fields, in terms of both vector directions and magnitudes. There are various methods of quantitatively comparing vector directions used in the turbulent flow diagnostics literature, such as using the root mean square [220, 221], point-to-point metrics [222], as well as more detailed weighted indices [223]. The relevance index (RI), also known as the cosine similarity, is perhaps the most widely-used in engine flow research [59, 61, 88], by virtue of producing a single number that represents the overall alignment of a vector field, scaled between 1 and -1 for convenience and generalisability. The RI is implemented in this work, allowing quantitative results to be compared to previous works more easily.

There are also numerous options regarding the vector magnitudes, such as point-

wise [222], root-mean square [224], distribution comparisons [134], and even neural network-learned metrics [225]. Histograms are a convenient and visual method of analysing image data, long used in image processing [226], and the similarity between two histograms can be computed with the histogram distance (HD) [227]. In the context of the present work, a ground-truth is also desired that can represent the true velocity magnitude distributions in the individual images without the dimishing effects of averaging. Working with velocity magnitude histograms provides such an opportunity, as the frequency counts from histograms of individual images can be averaged without the risk of diminishing the magnitudes. This is demonstrated in § 4.4.3. Vector magnitude histograms along with the HD are therefore propsed for the assessment of validation target creation in this work.

4.2.1 Relevance index

To use the RI for vector field data, each field is firstly re-organised into a column vector, such that the RI is given for two fields $\mathbf{q}_A$ and $\mathbf{q}_B$ as:

$$\mathrm{RI} = \frac{\langle \mathbf{q}_A, \mathbf{q}_B \rangle}{\|\mathbf{q}_A\|_2 \cdot \|\mathbf{q}_B\|_2} \tag{4.1}$$

where $\langle \cdot, \cdot \rangle$ is the inner product, and $\|\cdot\|_2$ is the $\mathbb{L}^2$ norm. By arranging the vector fields into columns prior to the calculation, the RI returns a single value that measures the overall alignment of the fields, ranging between $+1$ for a perfectly aligned field and -1 for a perfectly opposite field.

4.2.2 Histogram distance

The histogram distance (HD) quantifies the degree of overlap between two histograms. It is defined as:

$$\mathrm{HD} = \frac{\sum_i \left(\min\left(h_1(i), h_2(i)\right)\right)}{\min\left(\|h_1(i)\|, \|h_2(i)\|\right)} \tag{4.2}$$

for two histograms h_1 and h_2 evaluated at each bin i. The value of HD varies between 1 and 0 for histograms that overlap perfectly and do not overlap at all respectively.

4.3 DMD implementation for cyclic PIV data

Before presenting the results, a note is needed regarding the interpretation of DMD results based on in-cylinder PIV data, due to the cyclic nature of such measurements. Procedure 1 outlines the steps taken in conducting the DMD analysis on the data in this study. In this work, a similar view is taken to Qin et al. [97], who rely on the fact that the PIV snapshots were generated by the same system in order to conduct phase-dependent DMD. Sampling the dynamics of a system between cycles at a constant phase is known as taking a Poincaré section, and the mapping that relates the sample points on the section is known as the Poincaré map [228, 229]. Phase-dependent DMD then linearises the dynamics on the Poincaré section. In the present phase-dependent DMD case, DMD models the evolution of flow structures across cycles in order to spot patterns and variations in the data, rather than through consecutive time to investigate the physical growth and decay of these structures. The frequencies associated with this DMD analysis therefore describe how the flow structures oscillate among flow fields measured at the same crank angle, rather than relating to physical frequencies in the engine. The difficulty in physically interpreting these phase-dependent frequencies is one reason why the subsequent analysis is limited to the study of the 0 Hz 'average' modes.

Procedure 1 Implementation of the DMD method for flow field data.

Description	▷ **Key equations/parameters**
1: For PIV data at the analysis crank angle over M total cycles, split the dataset into blocks of size m, such that each block contains consecutive data. The size of m affects the DMD convergence properties, as discussed in §5.2.5.	▷ Eg. at -285 CAD: ▷ $M = 300$, $m = 100$ ▷ $n =$ No. locations in PIV grid
2: For each block, reshape the PIV data as columns in the data matrices Φ and Φ', with each column representing PIV data from a new cycle, and each row representing the velocity at a specific location in the PIV window.	▷ $n \times (m-1)$ matrices Φ, Φ' ▷ (See Equations 3.21a and 3.21b)
3: Perform DMD on the data matrices and produce the DMD modes. The modes, Φ, represent flow structures that 'oscillate' together from cycle to cycle at a given crank angle.	▷ $\Phi' = \mathbf{A}\Phi$ ▷ $\text{eig}(\mathbf{A}) = \Xi = \frac{1}{\lambda}\Phi'\mathbf{R}\Sigma^{-1}\Psi$
4: Extract the dominant 0 Hz DMD mode, giving the shape of the background flow field.	▷ $\Psi^{(j)} \mid \text{Im}(\Lambda^{(j)}) = 0$
5: Complete the flow field reconstruction by scaling the 0 Hz mode with the relevant amplitude.	▷ $\Psi^{(j)}\Lambda^{(j)}\alpha_{\text{opt}}^{(j)}$
6: Repeat for the remaining blocks of PIV cycles.	

4.4 Results

4.4.1 Initial validation

For the initial validation of the RANS model, the simulated pressure trace was plotted against the experimental average pressure for both the tumble plane (TP) and cross-tumble plane (CTP) experiments, as shown in Figure 4.1. Recall that each experiment yielded slightly different boundary conditions for the CFD model. Regarding Figure 4.1, a good overall agreement between the model and the experiments can be seen. There is a slight difference in the peak pressure, with the CFD model under-predicting the experimental average by 2.8% for the tumble plane experiment, and over-predicting the experiments by 0.8% for the cross-tumble plane. The only difference between the two CFD cases is in the boundary conditions, and as the simulated peak pressures sit within the experimental range, the CFD results are deemed to be a satisfactory match.

After looking at the global pressure, the goal would be to validate the simulated in-cylinder flow fields against experimental data from PIV to ensure that the predictions of local variables are also reliable. However, this is not a straightforward task, as previously discussed. It is hypothesised that significant CCVs in internal combustion engines, such as the flapping intake jet in this case, can cause the ensemble mean to become unrepresentative of the original PIV dataset. This casts doubt as to the suitability of using the ensemble mean as a validation target for RANS simulations. As a result of the intake jet flapping, the crank angles during the intake stroke while the intake valves are open (between -360 and -260 CAD aTDCf) are of interest. -285 CAD aTDCf is chosen as the analysis crank angle in this study as the intake jet is fully established, and the piston has moved far enough away from TDC to provide a larger field of view for the PIV measurements.

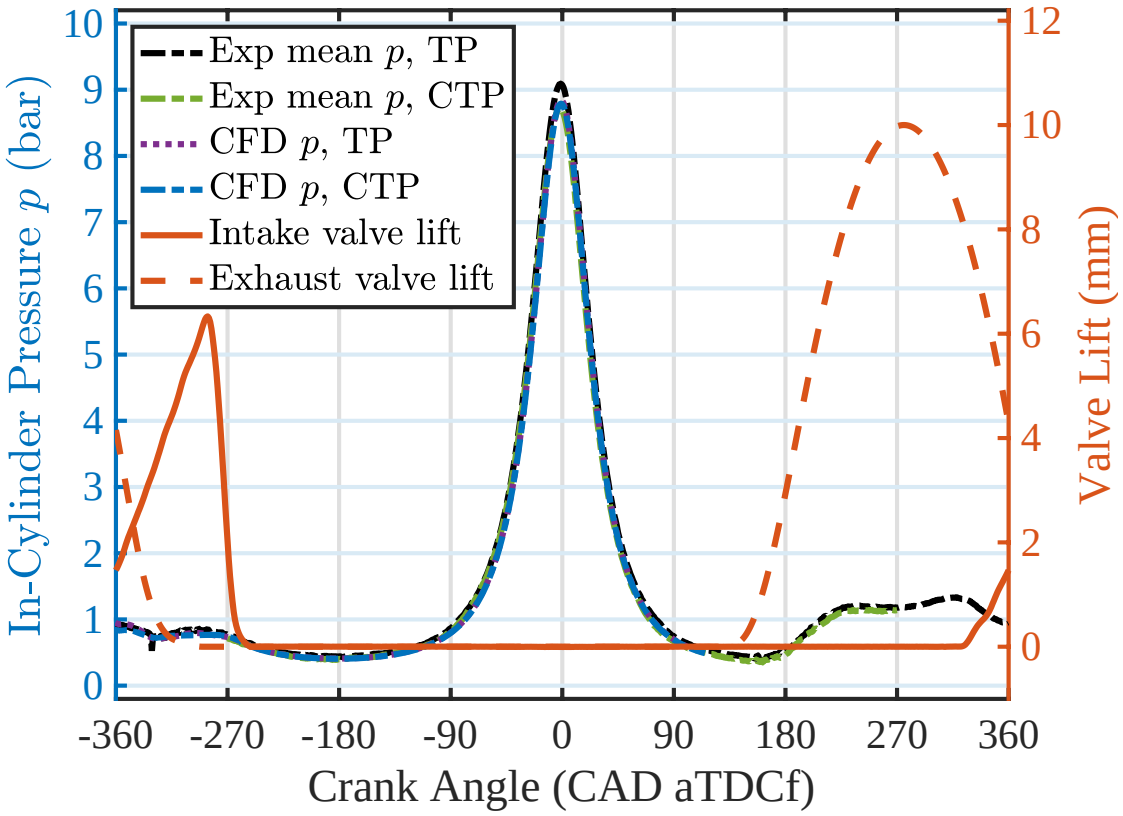

Figure 4.1: Comparison of in-cylinder RANS CFD and mean experimental pressure traces for both the tumble plane (TP) and the cross-tumble plane (CTP) experiments. The relevant boundary conditions corresponding to each experiment were used in the CFD set-ups. Experimental traces are plotted as dash-dotted lines, and CFD traces as solid lines. Valve lift profiles are also given in red.

4.4.2 Initial flow field analysis

Firstly, the adequacy of using 300 cycles for the PIV ensemble mean at the analysis crank angle of -285 CAD aTDCf is tested using the RI and HD, as shown in Figure 4.2, for both the tumble plane (left) and the cross-tumble plane (right). The two similarity metrics were used to compare consecutive ensemble-average fields composed of incrementally-increasing numbers of PIV snapshots, increasing from the first snapshot. For example, the RI and HD between the average of the first 49 PIV snapshots and the first 50 were 1.00 and 0.97, respectively. The HD is first calculated by converting each successive EM field into a histogram of the velocity magnitudes. From the figure it can be seen that the RI converged more quickly than the HD,

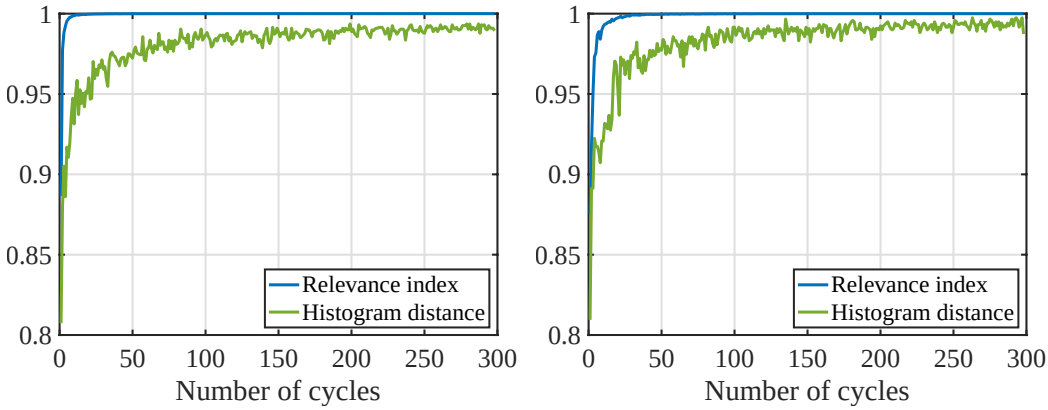

Figure 4.2: Convergence of the PIV ensemble mean for the tumble plane (left) and the cross-tumble plane (right). The plots show the RI and HD values between consecutive incremental ensemble-averages (ie. comparing the similarity of the ensemble mean with the first m cycles to the ensemble mean with the first $m + 1$ cycles).

suggesting that the RI is an easier metric to satisfy in this case. This is consistent with other reports which noted that the non-linear behaviour of the RI causes the metric to reach higher values more quickly [44, 88, 230].

A visual inspection of the plot suggests that the ensemble average only changes by small amounts when adding more cycles beyond the first 100, implying that 300 cycles is sufficient for this application. Figure 4.3 illustrates some example ensemble averages for the cross-tumble plane, showing that the EM indeed does not change significantly with the addition of more cycles after approximately 100 are included. The EM of the 300 cycle dataset is therefore satisfactory for the purposes of this study. Note that 0 mm in the flow field images corresponds to the centre of the cylinder in the radial directions, and the firing deck in the vertical direction.

The view of the tumble plane at -285 CAD aTDCf is shown in Figure 4.4. The top row consists of arbitrary consecutive PIV cycles, and the bottom row consists of the PIV ensemble mean (left) and the RANS CFD results (right). Along the top row, differences in the PIV images can be seen due to turbulent fluctuations, but core

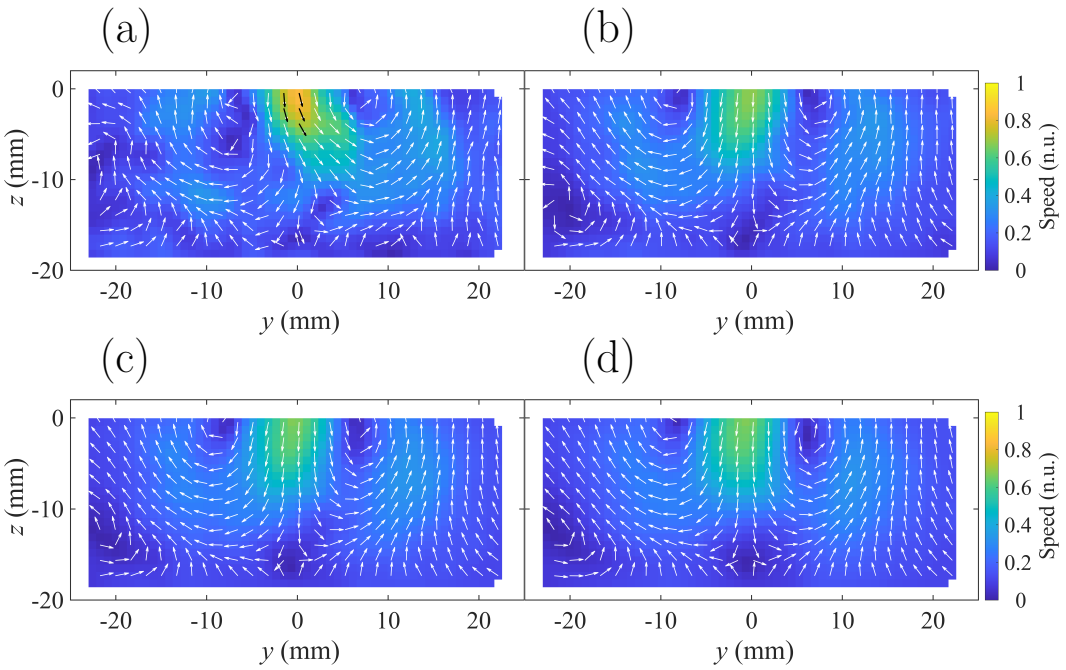

Figure 4.3: Example ensemble-average flow fields from the cross-tumble plane at −285 CAD aTDCf. The figure consists of averages containing: (a) the first 10 cycles; (b) the first 50 cycles; (c) the first 100 cycles; (d) the full 300 cycles. Note that flow speeds are in normalised units 'n.u.'.

flow features can still be identified. In both cycles, the flow is dominated by a strong cross-flow along the top of the cylinder. This creates an anti-clockwise tumble vortex, which has a centre located at approximately ($x = -10, z = -22$ mm) in both cycles. The tumble vortex structure does not vary much from cycle to cycle as the intake jet is always aiming downwards and away from the intake valves. The PIV ensemble mean in Figure 4.4 (c) can therefore be a good representation of the individual cycles, constructing a clear tumble vortex and preserving realistic velocity magnitudes, while smoothing over more fluctuating structures. Although the RANS result predicts a higher flow speed along the left of the cylinder and a slightly lower tumble vortex centre, it still presents a good match with the PIV ensemble mean, with RI= 0.91.

However, the flow structures are more complex on the cross-tumble plane. In the

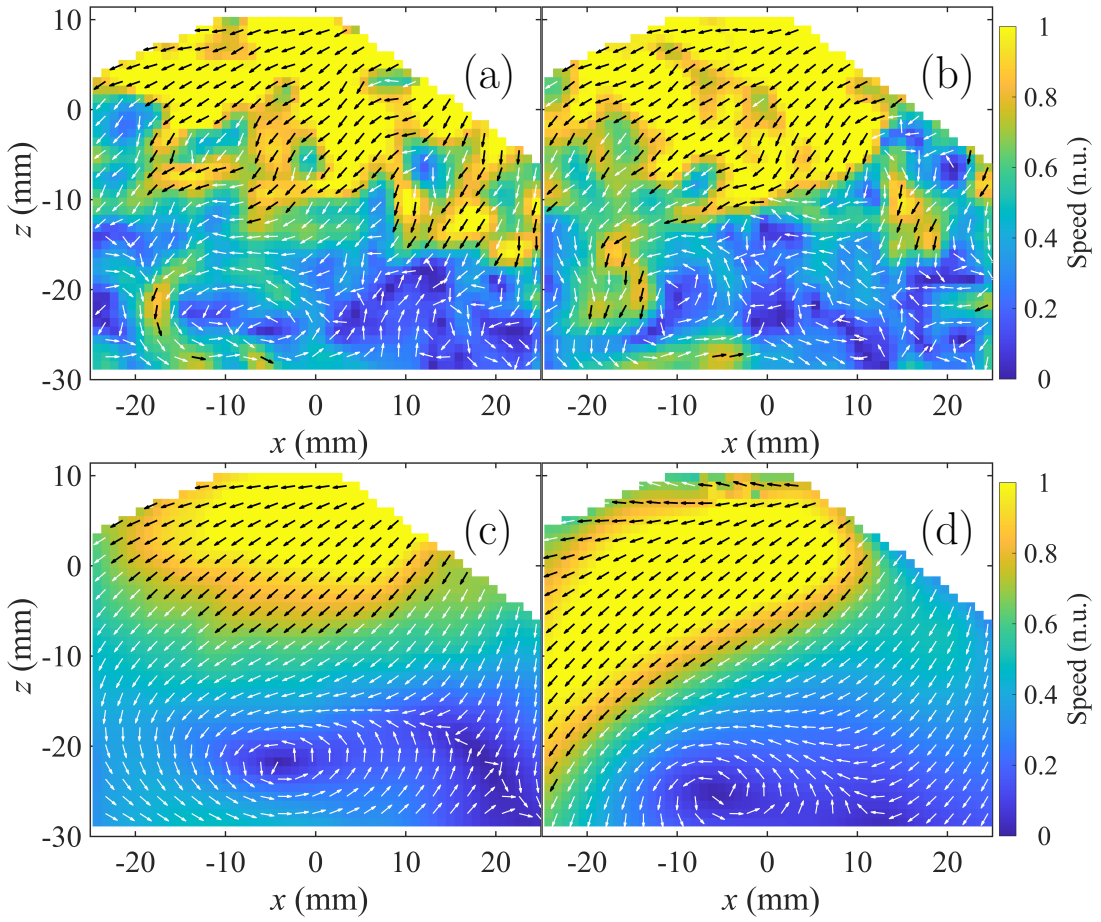

Figure 4.4: Flow fields on the tumble plane at −285 CAD aTDCf. The figure consists of: (a) single cycle A; (b) single cycle B; (c) PIV ensemble mean; (d) RANS CFD.

optical engine cylinder, separate air streams from the intake valves collide with varying strengths, resulting in jet flapping. The jets collide near the cylinder's central line of symmetry ($y = 0$), so this behaviour can be observed more clearly on the cross-tumble plane, illustrated in Figure 4.5 at −285 CAD aTDCf. The top row of the figure consists of arbitrary consecutive PIV cycles A (left) and B (right), where a high-speed intake jet can be observed in both flow fields. The motion of the jet flapping phenomenon can be seen, with the jet pointing in different directions between cycles. The ensemble mean is shown in the bottom-left of the figure, with a centrally-located intake jet and a vortex on either side. However, a qualitative comparison to the individual cycles reveals that the averaging process has diminished the magnitude of

the intake jet. This motivates an investigation into whether the ensemble averaged flow field is a fair representation of the original PIV dataset. In the RANS CFD flow field, a central intake jet is also predicted, though the magnitude of the RANS CFD jet is larger than that of the ensemble mean.

Comparing the CFD results to the ensemble average PIV field in isolation would naturally lead one to the conclusion that the simulation over-predicted the speed of the intake jet, with a potential recommendation that the CFD model should be adjusted to result in a slower jet speed. Conversely, when looking at some of the individual PIV snapshots, it would seem that the CFD slightly under-predicted the intake jet speed. This opposite conclusion would imply that the average PIV flow field does not faithfully represent the nature of the physics in the cylinder for any given cycle in this case. This hypothesis, which was initially proposed in a previous publication [59], is explored further in §4.4.3.

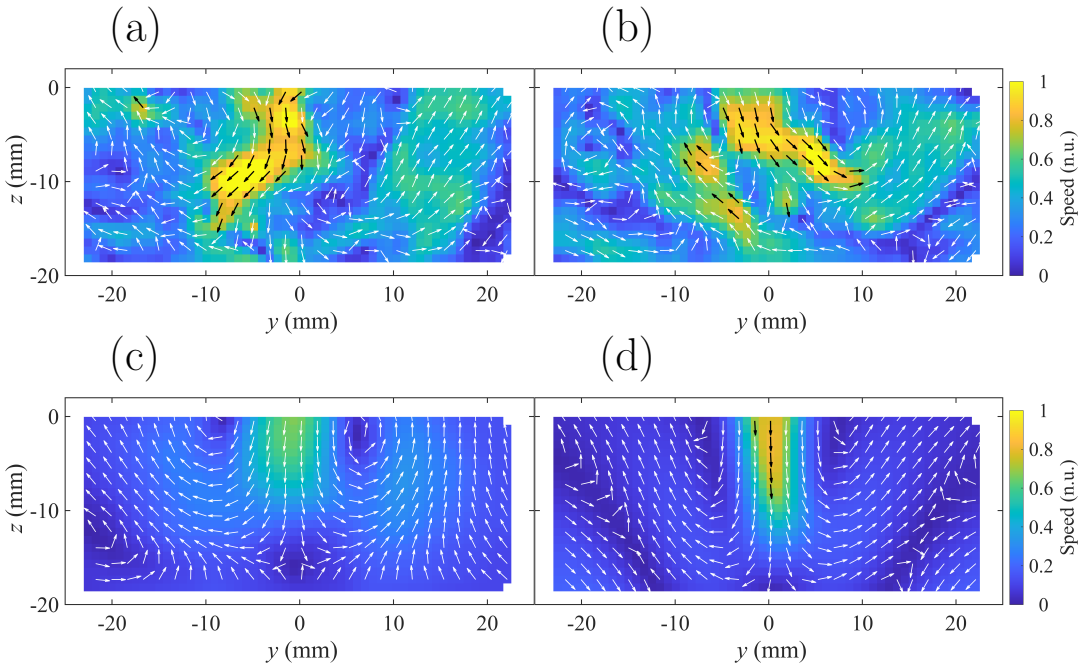

Figure 4.5: Flow fields on the cross-tumble plane at −285 CAD aTDCf. The figure consists of: (a) single cycle A; (b) single cycle B; (c) PIV ensemble mean; (d) RANS CFD.

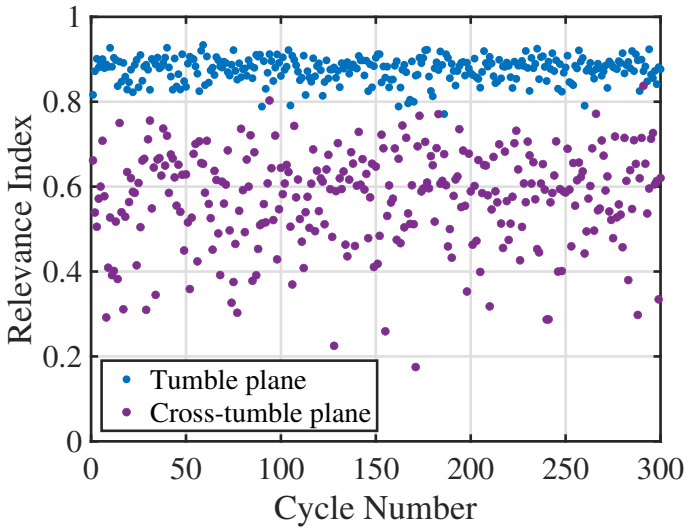

Figure 4.6: Relevance index between the PIV ensemble mean and each of the 300 PIV cycles for both the tumble and cross-tumble planes at −285 CAD aTDCf.

4.4.3 Representation quality of the ensemble mean

To quantify the directional similarity between the 300 PIV cycles and their ensemble mean, the RI was calculated between the ensemble mean and each cycle for both the tumble and cross-tumble planes, shown in Figure 4.6. The RI is consistently high on the tumble plane, but it is substantially lower and more variable on the cross-tumble plane. Further information can be obtained by looking at the velocity magnitudes. To represent the overall distribution of velocity magnitudes for each flow field, the magnitudes at each point in the field are plotted as a histogram. The histogram for the ensemble mean is then quantitatively compared to the ground-truth histogram using the HD. The ground-truth histogram is taken here to be the average of the 300 histograms representing each of the individual PIV cycles; note that this average histogram does not suffer from artificial diminishing of velocity magnitudes, as the numbers of instances for each of the velocity bins (which are always non-negative) are being averaged, rather than the velocity vectors themselves. This average histogram therefore contains contributions from each of the individual cycles without any cancellation or vector magnitude dimishing. The histogram averaging process

is shown in Figure 4.7. The plots comparing the ground-truth average histogram to the histogram of the ensemble mean are shown in Figure 4.8 for the tumble plane (a) and cross-tumble plane (b). There is a marked difference between the two planes, where the tumble plane has HD = 0.88, as opposed to HD = 0.57 for the cross-tumble plane.

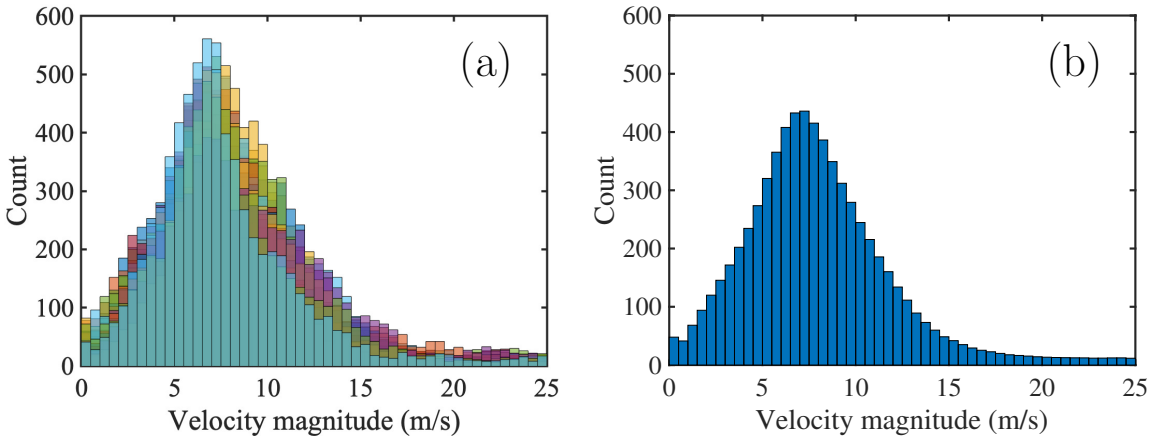

Figure 4.7: Velocity distributions plotted as histograms for **(a)**: 20 PIV snapshots, where the histogram for each individual snapshot is plotted in a different colour and super-imposed atop one another, and **(b)**: the average of 250 PIV histograms .

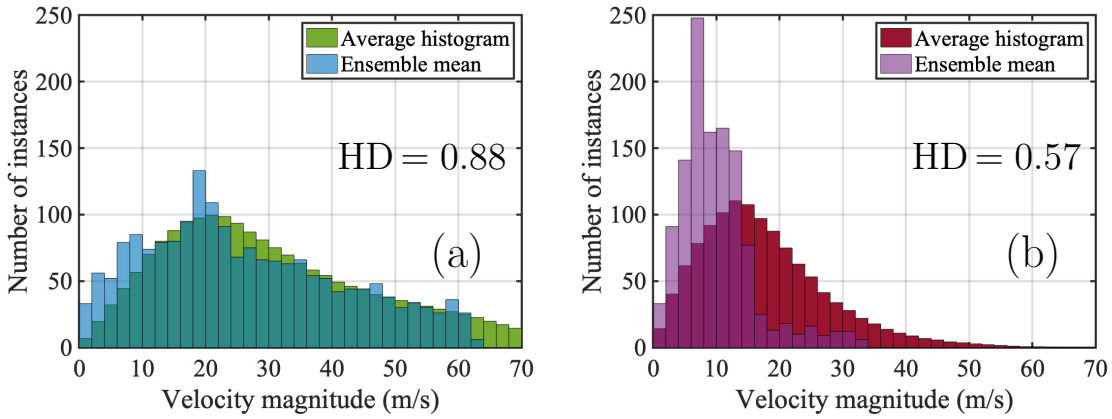

Figure 4.8: Area intersections between the ground truth histogram (taken as the average of the 300 histograms representing each PIV cycle) and the histogram for the ensemble mean, for both (a) the tumble plane and (b) cross-tumble plane. Histogram distances (HD) for each case are also reported.

This shows that while the ensemble mean is capable of fairly representing the flow on the tumble plane, it does not capture the flow dynamics as well on the cross-tumble

plane with regards to both velocity direction and magnitude. The reason for this is due to the additional cyclic variability on the cross-tumble plane. A statistical analysis on the horizontal velocity components in both planes was conducted to illustrate this. Figure 4.9 shows histograms of horizontal velocity components taken from cycle to cycle at a representative point in each plane:

- $(x = 0, z = -22)$ on the tumble plane, corresponding to one side of the ensemble mean tumble vortex,

- $(y = 0, z = -1)$ on the cross-tumble plane, corresponding to a location in the intake jet.

The velocities in these histograms are normalised with reference to the maximum absolute velocities at those points on the corresponding planes. For the tumble plane, the horizontal velocity component is close to a normal distribution, with a large number of cycles at the centre (near-zero horizontal velocity component, as the vector is mostly vertical at the vortex edge). The dominance of the average velocity at this point explains why the ensemble mean is able to provide a good representation of the ensemble of cycles. On the cross-tumble plane, the horizontal velocity component is more indicative of a bi-modal distribution, and the fitted normal distribution is a poor approximation of the ensemble. This represents the fact that in a large number of cycles, the intake jet points to the left or the right rather than straight downwards. A simple ensemble average is therefore expected to be a poorer representation of this dataset, as the mean velocity is less prevalent in the ensemble. There is therefore potential for more advanced numerical methods to provide a more suitable validation target for RANS results on the cross-tumble plane, and two such methods are explored in the following sections.

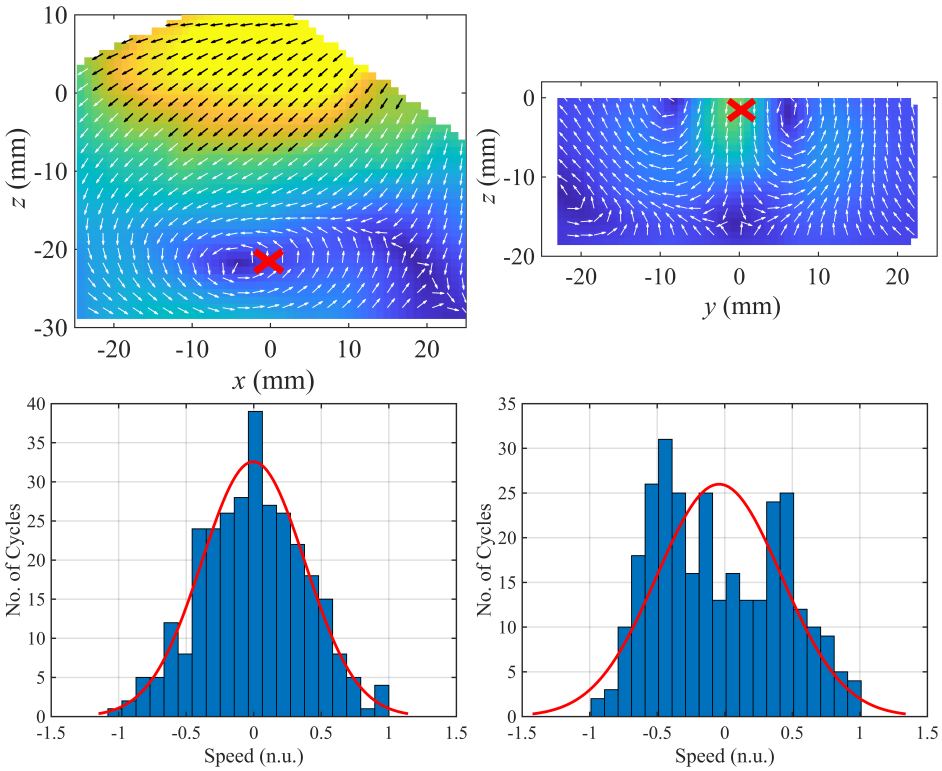

Figure 4.9: Cycle to cycle distributions of horizontal velocity components sampled at $(x = 0, z = -22)$ for the tumble plane (left), and $(x = 0, z = -1)$ for the cross-tumble plane (right). The red crosses in the top row of images illustrate the sample locations used to create the cycle-to-cycle velocity distributions. The velocity distributions are then shown as histograms on the bottom row, along with fitted normal distributions.

4.4.4 POD analysis

As previously discussed, the POD method offers a statistical approach for analysing the flow fields. In this section, the ability of POD to extract core flow features for the analysis of RANS simulations is explored. Figure 4.10 shows the POD-reconstruction of PIV cycles A (left column) and B (right column) on the cross-tumble plane, with the number of retained POD components increasing from one (at the top row) to 299 (at the bottom row). As there are 300 cycles of PIV measurements, and the data were centred prior to the POD analysis, there are a total of 299 POD modes. The reconstruction containing all 299 POD components is therefore an identical recreation of the original cycle.

Beginning at the top of the figure, the first row shows the first POD component, which captures the most amount of energy in the flow field at each cycle. In both cycle A (left) and cycle B (right), a broad central intake jet structure can be seen with a vortex to either side, producing a flow field that is reminiscent of the ensemble mean. Already at this level, there are signs of jet flapping between cycles A and B. Moving down the figure, the influence of CCVs can be seen as successively higher-order POD components are included in the reconstructions.

For cycle A, there is a large increase in the velocity magnitude of the central intake jet when the largest five POD modes are retained (third row). Cycle B has retained a higher jet velocity on the first mode, but doesn't show a large increase until the first ten modes are used in the construction (fourth row). There are two points worthy of note here.

1. The velocity magnitudes and directions in the POD reconstructions are dependent on the choice of cycle.

2. There is no clear 'cut-off' for how many POD components should be included in the flow reconstructions for RANS validation.

Cycle-dependency can pose a problem when attempting to validate RANS simulations, which only produce flow fields representing average structures. For a 300 cycle dataset using POD, even if an objective cut-off point for the number of components can be identified and fairly applied to each cycle, there will be 300 validation targets for the RANS results. This is why, in the context of validation target creation, POD-based reconstructions are labelled as many-to-many mappings in Table 3.1. If the measured flow fields are significantly different from one another, such as an intake

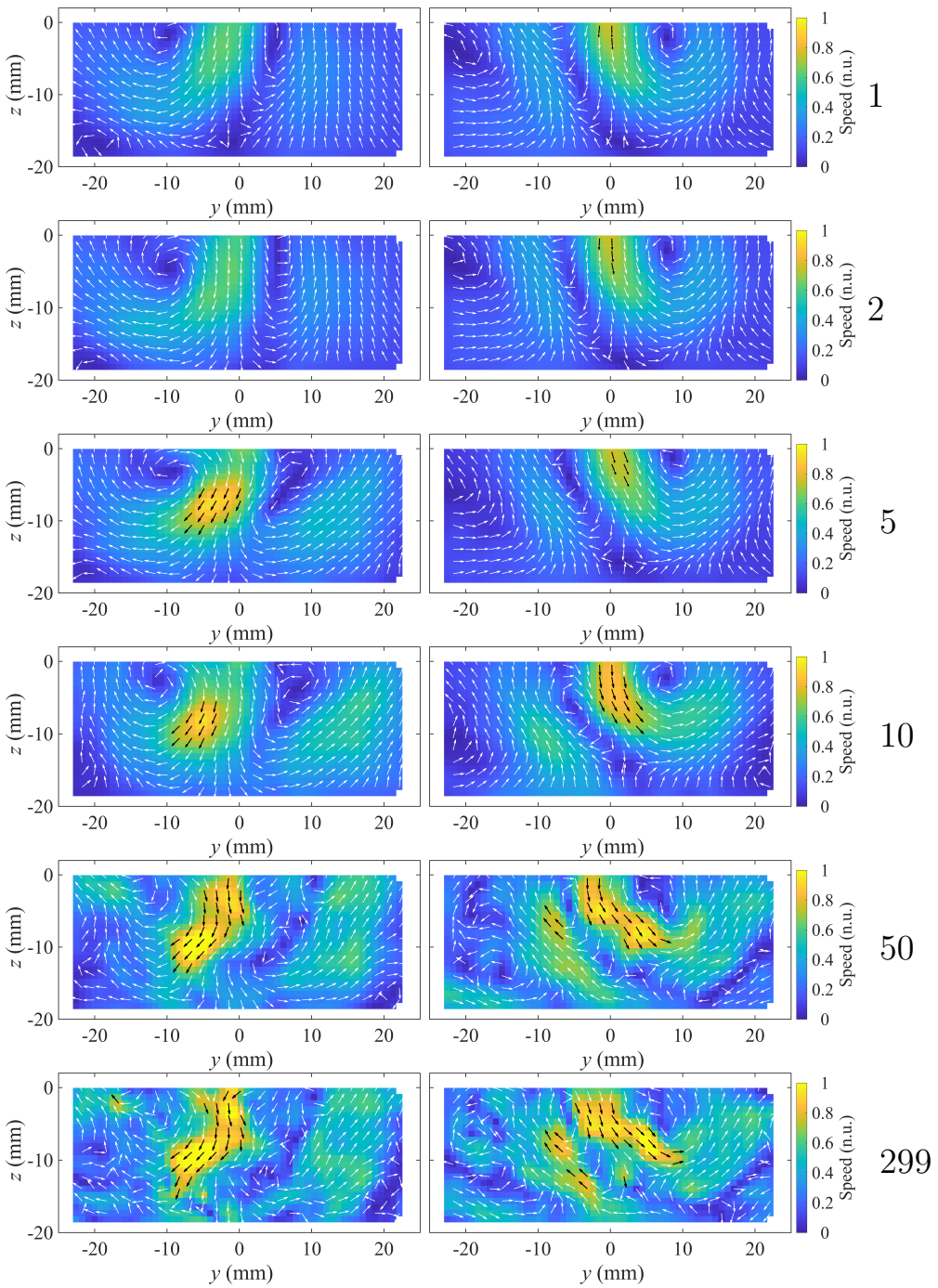

Figure 4.10: POD-reconstructed flow fields on the cross-tumble plane at -285 CAD aTDCf. Left column: cycle A, right column: cycle B. Each row consists of images containing a different number of POD components for the reconstruction, as labelled by the numbers to the right of each row. Top row: the first POD component, second row: the first two POD components, third row: the first five POD components, fourth row: the first ten POD components, fifth row: the first 50 POD components, sixth row: all 299 POD components (original cycle).

jet pointing in different directions, then the range of acceptable validation parameters may be excessively broad, making it excessively easy for the RANS model to match against [59]. The choice of cut-off is also an issue in itself; while the inclusion of more components allows for higher jet velocities to be retained, it may also take the reconstruction further away from the dominant flow structures, further increasing cycle-dependency. Deciding on cut-off criteria to optimally balance these aspects is a challenging task, and prone to subjectivity.

The POD-reconstructions are cycle-dependent because each space-only POD mode is linked to a specific point in time given by the corresponding right-singular vector. One way to collapse this time dimension is to model how the flow structures change with time; frequency-based modal decompositions such as the DMD can achieve this by producing modes that represent patterns in the data that oscillate at specific frequencies and grow or decay together in time. This concept is explored in the next section.

4.4.5 DMD analysis

To summarise so far, the following observations have been made about the PIV measurements in the cross-tumble plane that present challenges for interpreting and using the data.

- There are strong CCVs in the dataset, which cause velocity magnitudes to be diminished in the average flow.

- The CCVs are akin to a bi-modal distribution, where the average flow is less prominent in the ensemble.

Space-only, phase-dependent POD was used to investigate the flow fields by analysing the high-energy flow structures. While POD is capable of producing dominant flow fields that retain a higher intake jet velocity, the following limitations were discussed.

- Large number of validation targets with an excessively broad validation range due to cycle-dependency.

- Difficulty in deciding the number of POD components to retain in flow field reconstructions.

- A mixture of frequencies contained in each POD mode, so steady structures cannot be easily separated from fluctuating ones.

This motivates an investigation into Dynamic mode decomposition (DMD), which may be thought of as an ideal combination of spatial and temporal dimensionality reduction techniques [103]. DMD modes are coherent in both space and time, with a single frequency attached to each mode, and the results do not exhibit cycle-dependency. The DMD technique, therefore, can produce target flow fields across a narrower validation range, which can be provide more objectivity to the RANS validation process, as shown in this section.

The plot of DMD mode frequencies against amplitudes is known as the DMD spectrum, which illustrates the dominant frequencies in the dataset. The spectra for each of the three 100 cycle subsets on the cross-tumble plane are plotted on the top row of Figure 4.11, where the orange circles indicate the modes with the highest amplitude for each subset. However, as discussed previously, high-amplitude modes may rep-

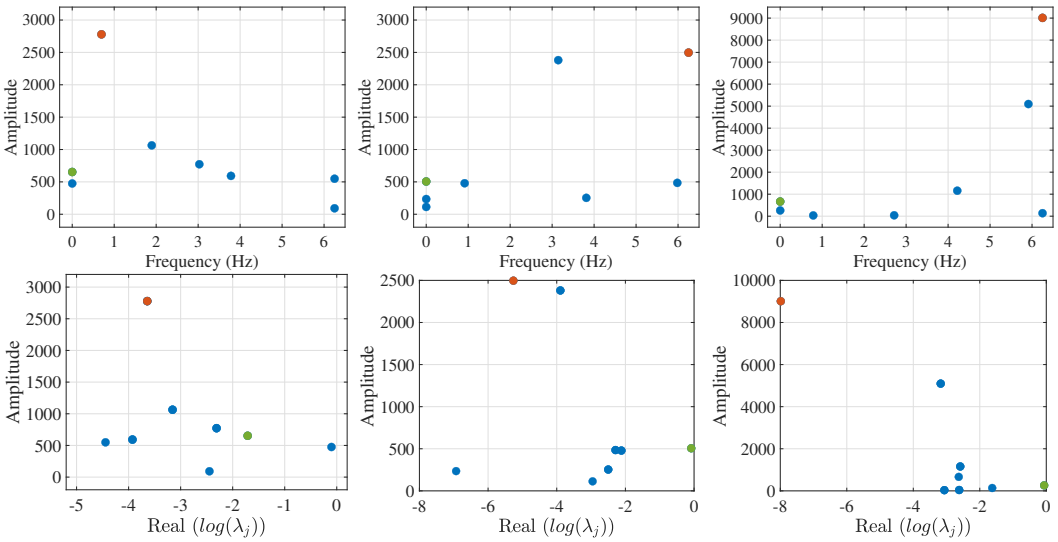

Figure 4.11: Top row: DMD spectra for the first, second, and third 100 cycle subsets of PIV (left, middle, and right plots respectively). Bottom row: DMD mode decay rates plotted against mode amplitudes for the first, second, and third 100 cycle subsets of PIV (left, middle, and right plots respectively). The dominant modes for each subset are given in orange, and the leading background (0 Hz) modes in green.

resent outlier flow structures with fast decay rates, not necessarily representing the dataset as a whole [194, 196]. This can be investigated further by plotting the decay rates, given by the real parts of the DMD eigenvalues, against the mode amplitudes. This is shown on the bottom row of Figure 4.11; modes towards the top-left of these plots indicate high amplitudes and fast decay rates. The high-amplitude modes are directed to the left of the figure in all three cases, with the third 100 cycle set (right) being the most extreme example with the largest amplitude and fastest decay rate relative to all the other modes.

The vector fields corresponding to each of the highest-amplitude modes are plotted along the top row of Figure 4.12. Each field lacks a coherent structure, with the extreme case for the third cycle set (top-right) also saturating the colourmap due to the extreme amplitude. These fast-decaying modes are of less relevance to the purpose of this work, which is to construct a suitable validation target for RANS simulations. As RANS simulations cannot predict the stochastic nature of turbulence explicitly,

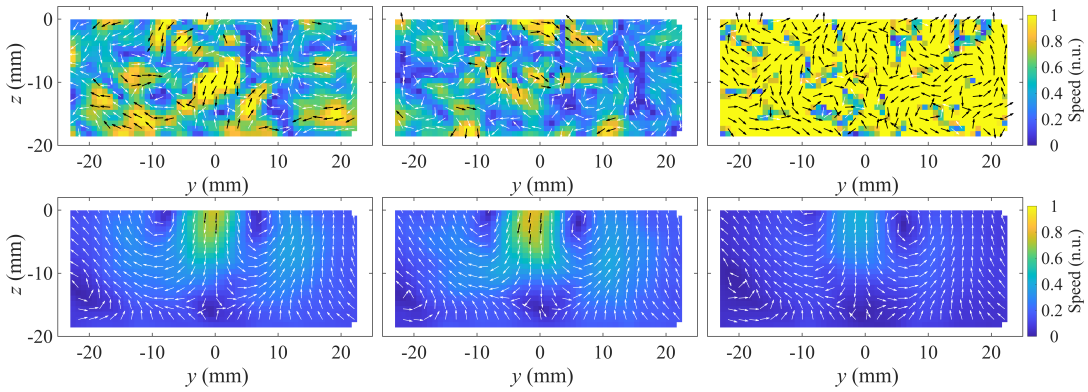

Figure 4.12: Flow fields showing the highest amplitude modes plotted in orange in Figure 4.11 (top row), and the leading 0 Hz modes plotted in green in Figure 4.11 (bottom row).

it may therefore be more relevant to consider the 0 Hz background DMD modes and compare those steady features to the RANS results.

The 0 Hz modes with the highest amplitudes are plotted as green circles in Figure 4.11 and are presented as vector fields in the second row of Figure 4.12. The 0 Hz modes for the first two cycle sets (left and middle) appear to be very alike; the overall structure is similar to the ensemble mean, but a higher intake jet velocity has been retained which is more representative of the individual cycles. The third cycle set, however, has an intake jet that is even more diminished than the ensemble mean. In addition, while there is a clear anti-clockwise rotational structure to the right of the intake jet, the left-hand counter-part appears to have shifted upwards slightly out of the field of view. An explanation for this can be given by observing some of the single cycle PIV snapshots, plotted in Figure 4.13. In these cycles, the intake jet from the right-hand valve appears to be stronger than the left, causing the intake jet to swing round and up to the top-left. As the right-hand rotational structure is fairly consistent among the three 0 Hz modes, it would seem that it is the flow from the left intake valve that is more variable, periodically delivering less air flow into the chamber, contributing to a more extreme left-swing of the jet in some cycles.

However, the large drop in the intake jet velocity for the third cycle set presents a problem when attempting to fairly represent the individual PIV cycles with DMD modes. Due to the nature of comparing snapshots taken at the same crank angle across engine cycles in phase-dependent DMD, the linear mapping between the snapshots may only be approximately true, which calls for an alternative definition of the mode amplitudes that does not rely so heavily on this assumption and the initial conditions. The sparsity-promoting DMD (SPDMD) introduced by Jovanovic et al. [196] addresses this by solving for an optimal set of mode amplitudes that recreates the original dataset as closely as possible with as few modes as possible. Amplitudes defined in this way prioritise DMD modes that have the largest contributions to the entire dataset and resist being skewed by fast decay rates caused by outliers, which may be a more appropriate method of representing the full set of individual PIV cycles in an engine.

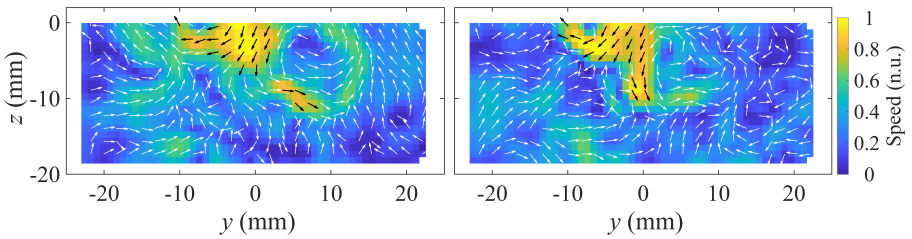

Figure 4.13: PIV snapshots of individual cycles showing the intake jet swinging around to the top-left of the flow field.

In order to use the SPDMD, the user-defined γ needs to be chosen. The γ parameter controls the trade-off between sparsity and the reconstruction error, with a larger γ leading to a more sparse solution and a higher reconstruction error. Jovanovic et al. [196] recommend conducting $\gamma-$value sweeps against the reconstruction error, as defined in Equation 3.40. The performance losses against various choices of γ for the present PIV dataset are shown in Figure 4.14. Like the POD, thresholds can also be chosen for the number of DMD modes to be included in the flow field reconstructions, and the γ sweeps are found to be sensitive to these thresholds, as

shown in the Figure.

Separate sweeps are shown for hard thresholds of 99, 98, and 12 (left, middle, and right respectively) to highlight the influence of changing the truncation cut-off. The hard threshold of 99 (left) represents no change from the original 100 cycle dataset, so the reconstruction error decays to zero for small values of γ up to 1000. Then as expected, sparsity is induced into the dataset with increasing γ, and the reconstruction error increases to over 60%. The extreme effect that truncation has on this dataset is evident when considering the sweep for a threshold of 98, as shown in the middle plot. Again, the reconstruction error converges for $\gamma < 1000$, but this time to a large error of 46%. This shows that the compression of the data onto a set of modes just one fewer than the original dataset causes large differences by this performance metric. This likely reflects the extreme non-linearities that map one snapshot onto the next in this dataset due to the phase-dependent implementation, with behaviours that are unable to be captured by the linear DMD-based model.

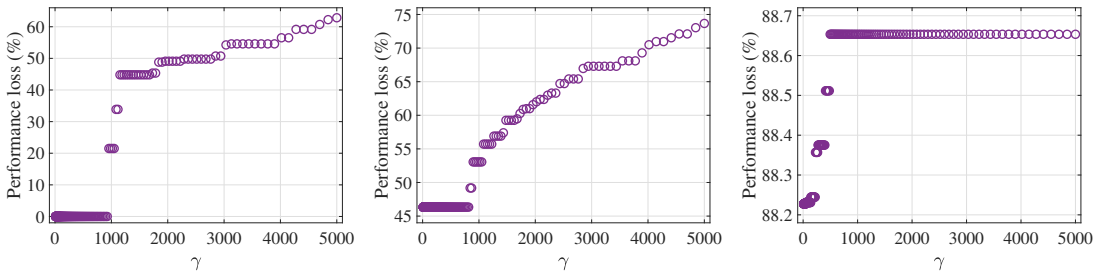

Figure 4.14: Computational sweeps for determining the optimal value of γ in the SPDMD algorithm for hard thresholds of 99, 98, and 12 (left, middle, and right plots respectively).

The γ sweep for a truncation to 12 modes is shown to the right of Figure 4.14. The value of 12 arises from the hard-thresholding criterion proposed by Gavish and Donoho [217]. For a rectangular matrix assumed to be comprised of entries containing a mixture signal and unknown levels of white noise, the optimal hard threshold is given

by:

$$p = \tau(\beta)\sigma_{\text{median}} \tag{4.3}$$

where β is the aspect ratio of the input matrix, τ is the hard threshold coefficient which can be evaluated numerically following Gavish and Donoho [217], and σ_{median} is the median singular value of the input matrix. The hard threshold coefficient generates singular values for a matrix that represents Gaussian white noise, scaled by the median singular value of the original data matrix. The largest 'noise singular value' is reported; all the original singular values larger than this are retained as signal, and smaller singular values are discarded as noise. Note that in the context of engine PIV data, a clean separation between Gaussian noise and signal might not be the most relevant threhsolding criterion, as other factors such as larger-scale CCVs might also be obscuring the flow features of interest. However, a theoretical basis for thresholding and noise removal in practical flows is still an open question [54, 106, 120], but a sparse representation of the data is still desired in order to form a validation target. The Gavish-Donoho criterion therefore serves as a principled place to begin the subsequent analysis. Using Equation 4.3, the hard threshold was given as 12, and this value was used to supply SPDMD modes for the forthcoming discussion of results.

Returning to Figure 4.14, the retention of just 12 modes already leads to a very sparse dataset, with a reconstruction error of over 88%. There is not much scope for introducing more sparsity via the γ parameter at this level of truncation; indeed, over this range of γ, the reconstruction error only varies within 0.5%. Although somewhat arbitrary due to the reduced impact of γ on the reconstruction error at this level of truncation, $\gamma = 1$ was chosen for the subsequent analysis. The spectra for the SPDMD modes for each set of 100 cycles are given in the top row of Figure 4.15. Compared to the standard DMD spectra, the SPDMD spectra are much more similar

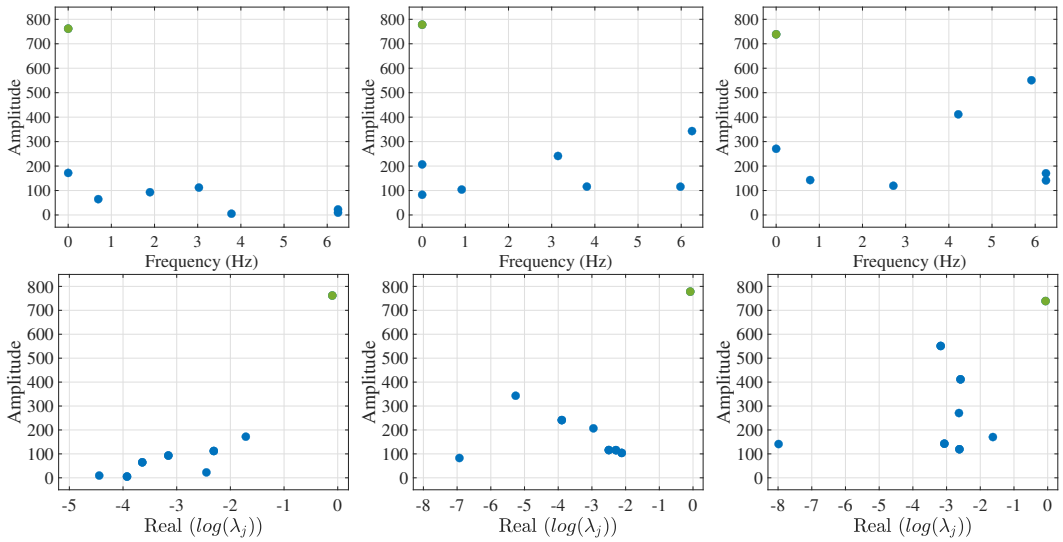

Figure 4.15: Top row: SPDMD spectra for the first, second, and third 100 cycle subsets of PIV (left, middle, and right plots respectively). Bottom row: SPDMD mode decay rates plotted against mode amplitudes for the first, second, and third 100 cycle subsets of PIV (left, middle, and right plots respectively). The dominant modes for each subset are also the leading background (0 Hz) modes, plotted in green.

across the three sets of cycles, with the dominant background 0 Hz modes plotted in green giving similar amplitudes. The plots on the bottom row of Figure 4.15 show that these dominant background modes also have near-zero decay rates, and contribute to the full datasets as a result.

Finally, the dominant SPDMD modes are plotted as flow fields in Figure 4.16. The velocity magnitudes are now much more consistent across the three cycles sets, indicating the robustness of the SPDMD amplitude re-scaling. These flow speeds are also more representative of the individual PIV cycles, with the histogram intersections for each of the three background DMD modes given in Figure 4.17. Note that the average ground-truth histogram is slightly different in each case, containing the relevant sets of 100 PIV cycles for the DMD analysis. The histogram distances for the first, second, and third PIV subsets are 0.89, 0.87 and 0.89 respectively, showing consistent improvement from the histogram distance of 0.57 given by the ensemble mean. These 0 Hz SPDMD modes are therefore recommended for use as validation

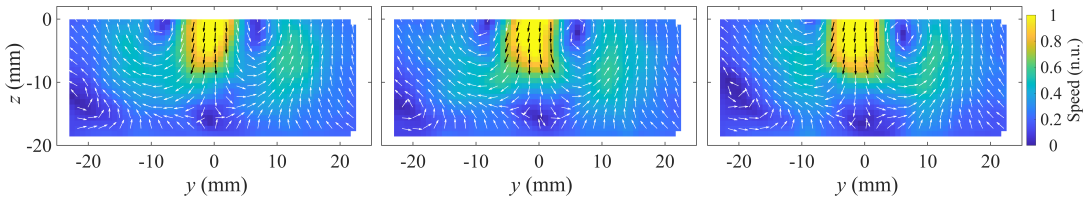

Figure 4.16: Flow fields showing the dominant SPDMD modes plotted in green in Figure 4.15.

targets.

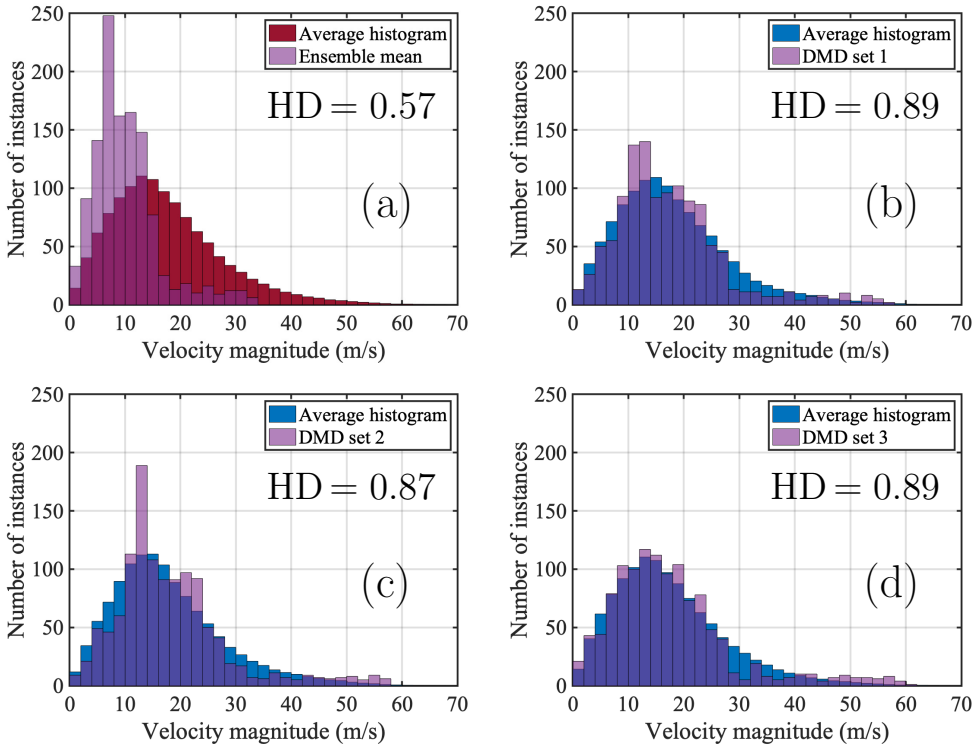

Figure 4.17: Histogram distances between the 'base case' average histograms and the ensemble mean (top-left), and the three dominant SPDMD modes for each of the 100 cycle subsets shown in Figure 4.16.

4.5 Conclusion

This chapter has investigated the creation of validation targets from PIV data, with a focus on the diminished magnitudes problem due to ensemble averaging. The diminished magnitudes effect has been quantified for the first time here using the histogram distance. In addition, the extent of the diminishment has been linked to

the degree of variability within the flow, with the HD between the ensemble mean and the ground-truth histograms giving HD= 0.88 and 0.57 for the tumble plane and cross-tumble plane respectively. A key difference between the flow on these two planes is the amount of CCVs experienced, which was observed by calculating the RI between the ensemble mean flow field and each of the individual PIV snapshots for each plane.

Previous studies have suggested that POD-reconstructed flow fields may be more suitable validation targets they allow for the retention of higher velocity magnitudes. This technique is investigated here, and more realistic vector magnitudes are indeed possible, depending on the threshold chosen for the number of POD modes included. However, each POD-based reconstruction is unique to the specific point in time (in this case, the engine cycle) dictated by the right-singular vectors. This means that there is no collapsing of the data, such that the present PIV dataset of 300 images gives 300 validation targets. This many-to-many mapping between the PIV dataset and the POD-based reconstructions means that there is little benefit from using the POD fields as validation targets over the original PIV images, which is undesirable for the comparison to a single RANS flow field.

The exact DMD algorithm is then explored as it is a spatio-temporal decomposition method, capable of removing the subjectivity associated with POD's time-dependency by focussing on modes that are defined by specific frequencies. However, the validity of the standard amplitude definition is questioned in this context of phase-dependent DMD, with the high-amplitude modes also corresponding to high decay rates, indicating that these flow structures do not influence the entirety of the PIV datasets, and may in fact represent outlier motion. SPDMD is therefore proposed as an alternative solution, capable of more faithfully representing the true dynamics of the

system by assigning high mode amplitudes to flow structures that can most accurately recreate the original PIV datasets via the ADMM optimisation routine. The dominant SPDMD modes are also the 0 Hz background modes, and they resemble the average flow fields but with velocity magnitudes that are more representative of the individual cycles.

Although the DMD methods appeared to be inadequate models of the phase-dependent dynamics due to the high reconstruction errors reported in the γ sweeps, the SPDMD was shown to be capable of effectively re-scaling the 0 Hz modes to faithfully represent the velocity magnitudes in the individual PIV snapshots. This was quantified using the HD, where the SPDMD 0 Hz modes were shown to provide the best matches to the ground-truth histograms over all three subsets of the PIV data, yielding HD of up to 0.89 on the cross-tumble plane. Therefore, it is suggested that the SPDMD method can make the RANS validation process more objective and accurate, by collapsing the data onto a single representative snapshot while retaining the original vector magnitudes. Despite this improvement over the POD and exact DMD methods, some subjectivity still remains with the SPDMD method regarding the choice of threshold criteria. It is likely that in order to truly rectify this specific issue, focus should be moved away from SVD-based methods and towards optimisation algorithms such as the ADMM, which is the key component of the SPDMD method that is able to re-scale the velocity magnitudes. Closer study of optimisation algorithms are therefore suggested for future work in validation target creation.

Chapter 5

Creating validation targets for PIV and LES

5.1 Overview

The previous chapter introduced the DMD and SPDMD techniques and explored their ability to create validation targets from PIV data. The SPDMD 0 Hz modes were recommended as validation targets by virtue of being able to collapse the data onto a single target while retaining accurate vector magnitudes. That work considered the processing of PIV data in isolation, motivated by a comparison to a single snapshot from a RANS model. However, in the case of LES data, both the CFD and experimental datasets typically contain ensembles of images, such that the SPDMD modes representing each dataset will interact with one another. The research in this chapter therefore builds upon the previous work by extending the methodology to an LES validation pipeline, and seeks to provide robust validation targets for each

dataset alongside a discussion of their interpretations.

In particular, this chapter is concerned with PIV data from the well-known Darmstadt engine [171, 231], and velocity data from large-eddy simulations (LES) of the same engine [44, 174]. The previous work on this LES data developed validation methodologies that utilised ensemble averaging and POD; the use of the SPDMD in the present study enhances this prior work by facilitating a closer consideration of the velocity magnitudes in the data. Therefore, new insights are revealed about how to validate LES simulations against experimental data, quantify the similarity of the simulations to the PIV data, and locate sources of error in the datasets. Firstly, the representation quality of the ensemble mean (EM) flow fields is assessed for both the LES and PIV data. The SPDMD 0 Hz modes are also explored, and this method is shown to yield more representative velocity magnitudes in both cases. The impact of using the SPDMD 0 Hz modes to represent the flows rather than the EM is then explored. The effect of two important parameters in the DMD algorithm, namely the mode threshold number and the total number of snapshots, are investigated in this application to aid and inform practitioners in the implementation of the SPDMD method in engine-relevant flows.

5.2 Results

5.2.1 Ensemble mean analysis

The ensemble mean (EM) flow fields from the Darmstadt engine PIV and LES at 470 CAD aTDCf are shown in Figure 5.1 **(a)** and **(b)**, respectively. In both cases, the intake jet is shown as the high-speed yellow portion of the flow to the top-right of the flow fields. The two intake jets show a qualitative match in terms of length and magnitude; however, the resulting vortex centre is positioned further to the right in

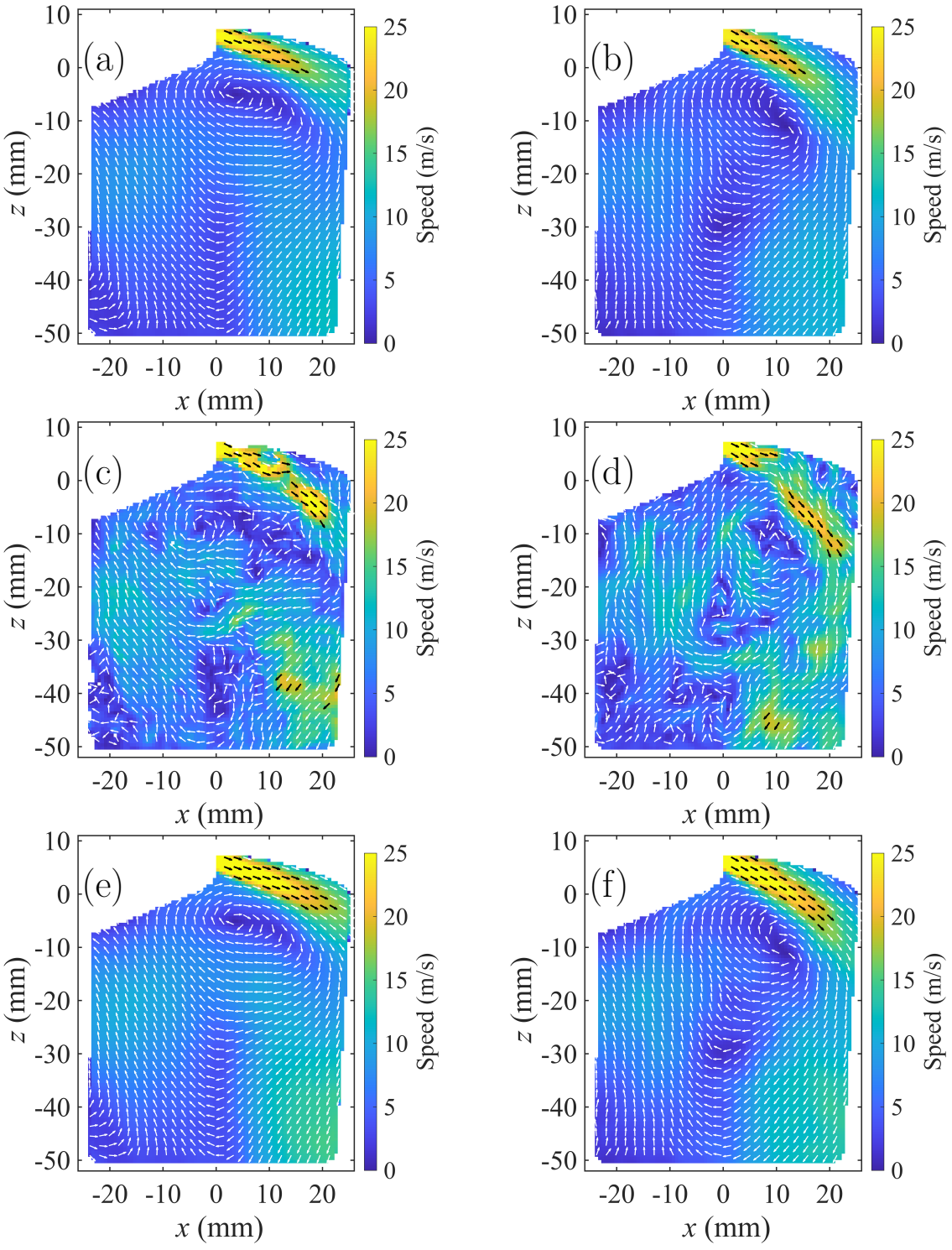

Figure 5.1: Flow fields at 470 CAD aTDCf. The plots show **(a)**: the PIV ensemble mean (EM), **(b)**: the LES EM, **(c)**: an arbitrary PIV snapshot, **(d)**: an arbitrary LES snapshot, **(e)**: the PIV SPDMD 0 Hz mode truncated at 57 modes, **(f)**: the LES SPDMD 0 Hz mode truncated at 14 modes. One in every three vectors are plotted for clarity. The colourmap is scaled by the vector magnitudes, referred to as the flow speed.

the EM LES field. This points to the possible existence of differences in the intake jet that are not immediately made apparent by the EM fields.

In order to further examine the intake jet, single snapshots from PIV and LES are shown in Figure 5.1 **(c)** and **(d)**, respectively. In these images, the intake jets are shown to extend further into the flow fields for both cases. The fact that the intake jets are shorter in the EM fields suggests that the tips of the intake jet are more variable from snapshot to snapshot, causing them to get smoothed over in the EM fields. The single LES snapshot also shows the intake jet pointing further downwards than the jet in the single PIV image, which may contribute to the different vortex centre locations. Finally, a faster recirculation zone can be seen in the bottom-right of both of the single PIV and LES images that is not obvious in either EM field, suggesting that this section is another area of significant variability.

As for the Oxford optical engine previously, these differences between the EM fields and the single snapshots call into question the ability of the EM fields to fairly represent the instantaneous velocity magnitudes present in the original data. Histograms are also used here to quantify how well vector magnitudes from the original data are preserved. Recall that these magnitude histograms measure the number of occurrences of a particular magnitude bin, which is strictly positive, so they can be averaged without risking the cancellation of opposites that comes with averaging vectors directly. The result is a 'cycle set histogram', used as a ground-truth, that represents the velocity magnitudes across the entire dataset.

With this method, the EM velocity magnitude distributions can be compared to their respective cycle set histograms, shown on the top row of Figure 5.2 for both the PIV **(a)** and LES **(b)** flow fields. In both cases, the EM histograms in purple

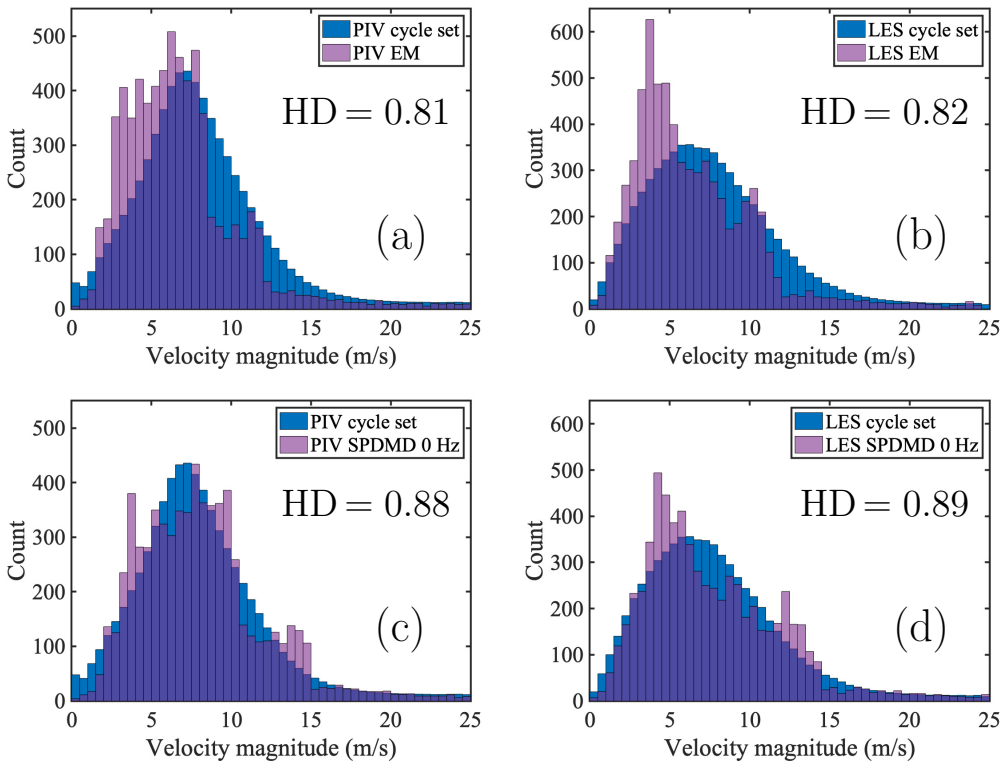

Figure 5.2: Comparisons between the average velocity histograms at 470 CAD aTDCf and the velocity distributions of **(a)**: the PIV EM, **(b)**: the LES EM, **(c)**: the PIV SPDMD 0 Hz mode, and **(d)**: the LES SPDMD 0 Hz mode. Histogram distances (HD) for each case are also reported.

display a left skew (indicating lower velocity magnitudes) when compared to the cycle set histograms in blue. This is because the superposition of vectors in the EM calculation results in vector cancellation and a systematic under-prediction of the true velocity magnitudes. This effect is quantified via the histogram distance (HD), which calculates the relative size of the overlapping area between two histograms. HD = 0.81 between the PIV EM and the PIV cycle set, and HD = 0.82 between the LES data and its ensemble average.

The ability of the EM to represent velocity magnitudes depends on the variability in the dataset, so the HD values are expected to decrease for more variable crank angles and test points. For example, the results for another important crank angle

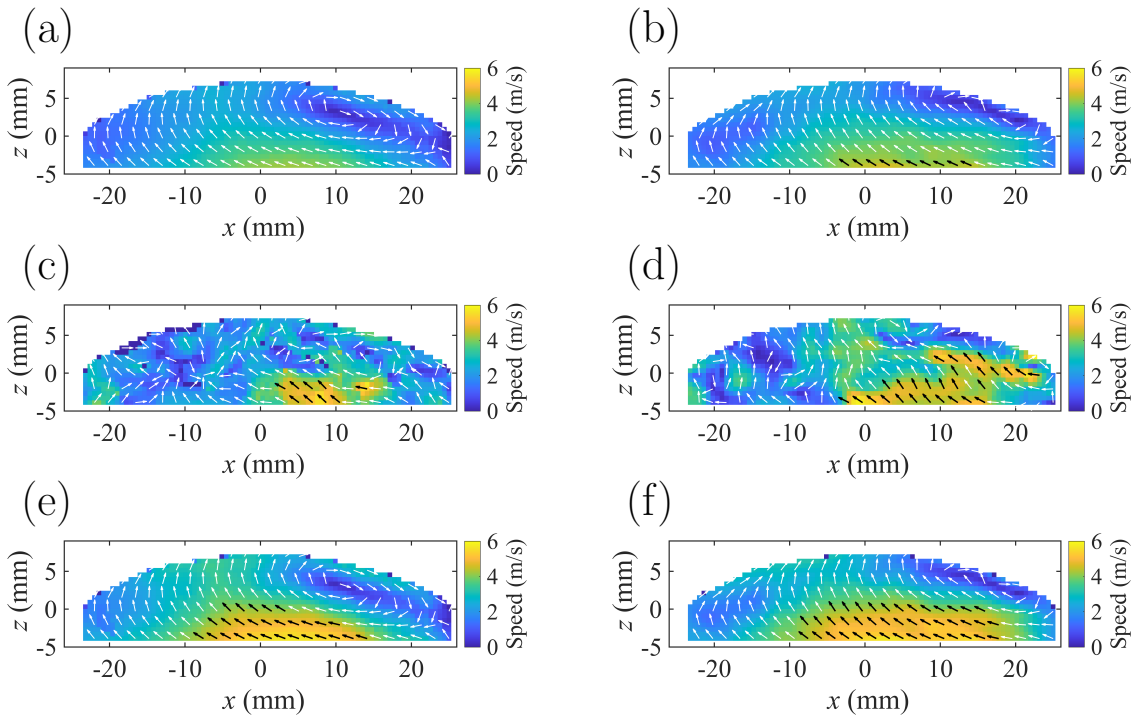

Figure 5.3: Flow fields at 700 CAD aTDCf. The plots show **(a)**: the PIV ensemble mean (EM), **(b)**: the LES EM, **(c)**: an arbitrary PIV snapshot, **(d)**: an arbitrary LES snapshot, **(e)**: the PIV SPDMD 0 Hz mode truncated at 189 modes, **(f)**: the LES SPDMD 0 Hz mode truncated at 10 modes. One in every three vectors are plotted for clarity.

in this engine, 700 CAD aTDCf, are presented in Figure 5.3. For this 700 CAD case, disparities can be seen between the PIV and LES EM fields ((**a**) and (**b**) and single snapshots ((**c**) and (**d**)), particularly towards the bottom of the flow fields where there is a relatively strong recirculation zone off the top of the upward-moving piston. As a result, the HD between the EMs and the respective cycle set histograms is lower, at HD = 0.71 for the PIV and HD = 0.75 for the LES.

5.2.2 SPDMD analysis

As recommended in the previous chapter, the 0 Hz modes from the SPDMD algorithm were then explored as alternatives to the EM. As before, choice of the γ parameter that controls the sparsity−accuracy trade-off in the SPDMD algorithm had a negligible

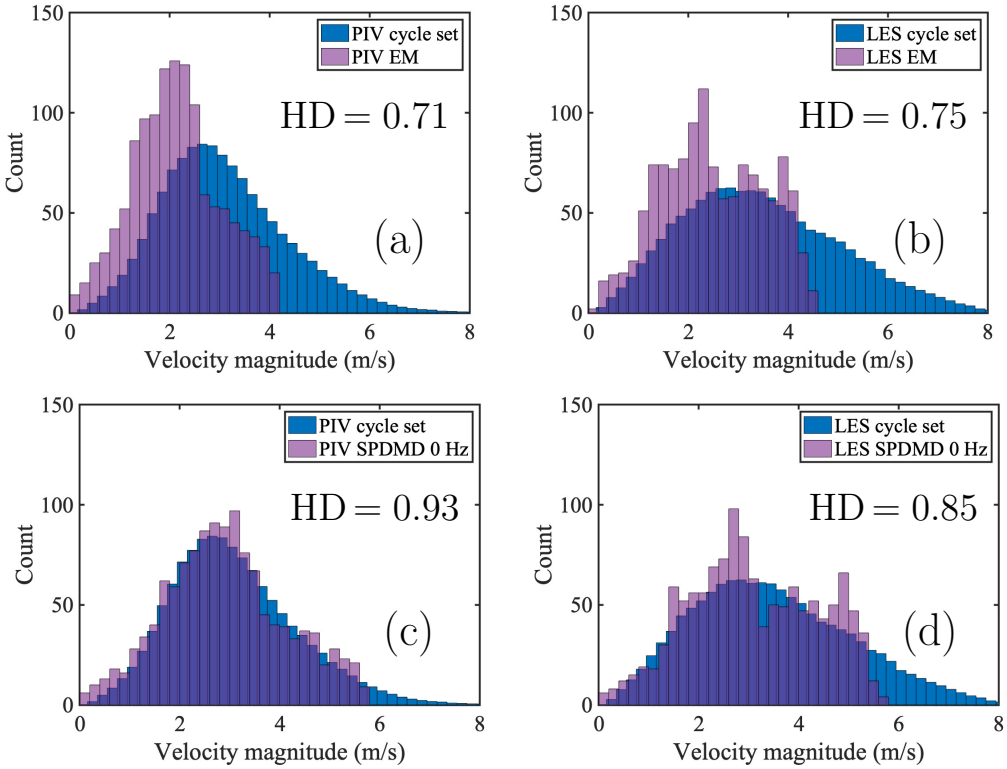

Figure 5.4: Comparisons between the average velocity histograms and the velocity distributions at 700 CAD aTDCf of **(a)**: the PIV EM, **(b)**: the LES EM, **(c)**: the PIV SPDMD 0 Hz mode, and **(d)**: the LES SPDMD 0 Hz mode.

effect on the velocity magnitudes given by the 0 Hz modes for the 250 cycle PIV dataset. Indeed, for a fixed mode threshold number of 50 modes, increasing γ from 0.01 to 10000 only changed the HD between the resultant 0 Hz modes by 0.005. Therefore, γ was simply set to one for the remainder of the study.

As previously discussed, use of the singular value decomposition (SVD) in the DMD algorithm means that the DMD analysis can also be sensitive to the truncation or hard threshold for the number of DMD modes retained in the analysis. A principled approach can be applied to the definition of a hard threshold, but as discussed in §4.4.5, these approaches may not be valid for engine flows where the difference between noise, CCVs, and flow features of interest is not necessarily clear. Therefore, sweeps over the choice of mode threshold number were conducted for both the PIV and the

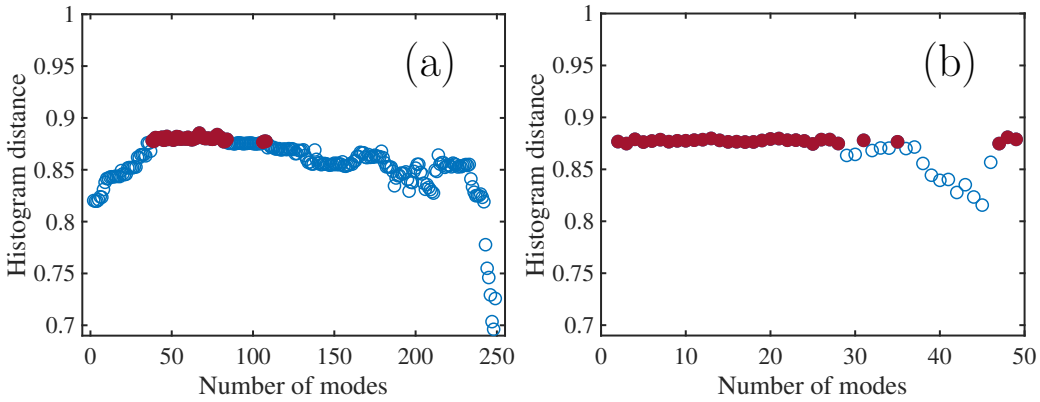

Figure 5.5: Sweeps calculating the HD between the SPDMD 0 Hz mode and the cycle average histograms for varying mode threshold numbers for **(a)**: PIV data, and **(b)**: LES data at 470 CAD aTDCf. The filled red circles represent HD values that are within 1% of the maximum HD value.

LES data, to find the threshold that resulted in the 0 Hz modes with the largest HD against the average histograms.

The results for 470 CAD are shown in Figure 5.5, where the filled red circles represent thresholds with high HDs that were within 1% of the maximum HD found in each sweep. For the PIV data, a continuous streak of high HD values can be found between mode thresholds of 38 and 84. For the LES data, there was a longer continuous streak of high HDs found between 2 and 28 modes. In both cases, therefore, there were many good options for the truncation number. As an attempt to provide consistency to the choice of threshold for the two datasets, the distribution of the singular values for each case was investigated, as plotted in Figure 5.6. The mode thresholds of 38 and 84 correspond to 41% and 61% of the cumulative variance in the PIV dataset, while for the LES data, the thresholds of 2 and 28 modes correspond to a wider range of 22% and 74% of the variance. As a middling value for both cases, a criterion of 50% of the variance was used to select the threshold for each. The thresholds used for the SPDMD analysis at this crank angle were therefore 57 for the PIV data, and 14 for the LES data, as marked in Figure 5.6. Note that this criterion is not intended to

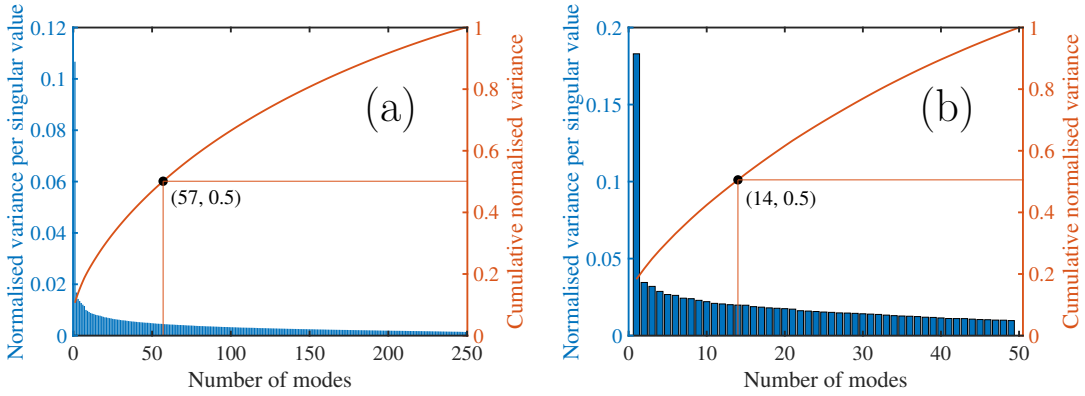

Figure 5.6: Singular value decomposition plots at 470 CAD aTDCf for **(a)**: 250 PIV snapshots, and **(b)**: 50 LES snapshots. The variance captured in each singular value is given by the bar chart on the left axis, and the cumulative variance is given by the line graph on the right axis. The number of modes corresponding to 50% of the cumulative variance is marked for each dataset.

be valid for other datasets or test points, and only serves as a method of selecting a single threshold from the variety of good options in this case. The subjectivity in choosing hard thresholds for SVD-based methods remains a concern, which might be alleviated by considering the ADMM algorithm in isolation from the SPDMD.

The 0 Hz modes at the threshold numbers of 57 and 14 are plotted as flow fields in Figure 5.1 **(e)** and **(f)** for the 470 CAD case. In these fields, the ADMM algorithm has re-scaled the average flow by optimising the velocity magnitudes across each of the PIV and LES snapshots. As a result, the intake jets for the SPDMD 0 Hz fields have longer lengths and faster speeds. In addition, a difference in the angle of the intake jets is made more apparent, which is a potential cause of the different vortex centre locations between the PIV and LES data. Finally, faster recirculation zones have been retained in the bottom-right corners of both SPDMD fields, which is also more representative of the dynamics in the individual PIV and LES snapshots.

For 700 CAD, the HD sweeps for the PIV and LES data are shown in Figure 5.7. Here, the results look vastly more disparate than for 470 CAD, which indicates that

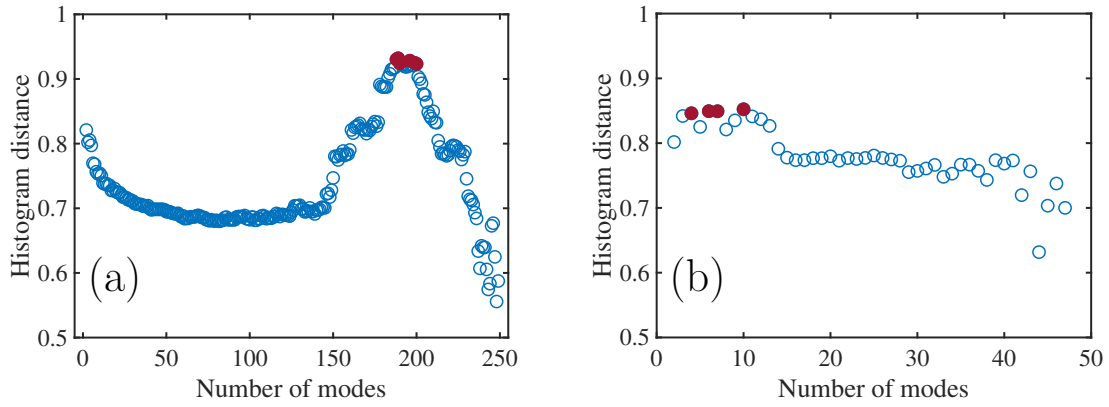

Figure 5.7: Sweeps calculating the HD between the SPDMD 0 Hz mode and the cycle average histograms for varying mode threshold numbers for **(a)**: PIV data, and **(b)**: LES data at 700 CAD aTDCf. The filled red circles represent HD values that are within 1% of the maximum HD value.

the PIV and LES data have different levels of variability at 700 CAD. The optimal thresholds for this crank angle were 189 for the PIV data, corresponding to 93% of the cumulative variance, and 10 for the LES data, corresponding to 50% of the cumulative variance. The threshold for the PIV data at this crank angle is a fairly extreme result, and is potentially symptomatic of significant CCVs, as the low-order DMD modes are not very representative of the data and many modes are needed for an accurate reconstruction. The level of variation in the velocity magnitudes of these datasets is further explored later in Figure 5.9. The SPDMD 0 Hz modes at thresholds of 189 and 10 are plotted in Figure 5.3 **(e)** and **(f)** respectively. The effect of using the modes is similar to the 470 CAD case, with higher velocity magnitudes being retained in the recirculation zones in particular.

The corresponding histograms between the 0 Hz SPDMD fields and the full cycle sets are plotted along the bottom row of Figure 5.2 for 470 CAD. Unlike the EMs, the SPDMD 0 Hz histograms exhibit less overall skew, indicating a more accurate representation of the velocity magnitudes in the dataset overall. A slight left skew for LES SPDMD histograms is observed here which could be an indication that more

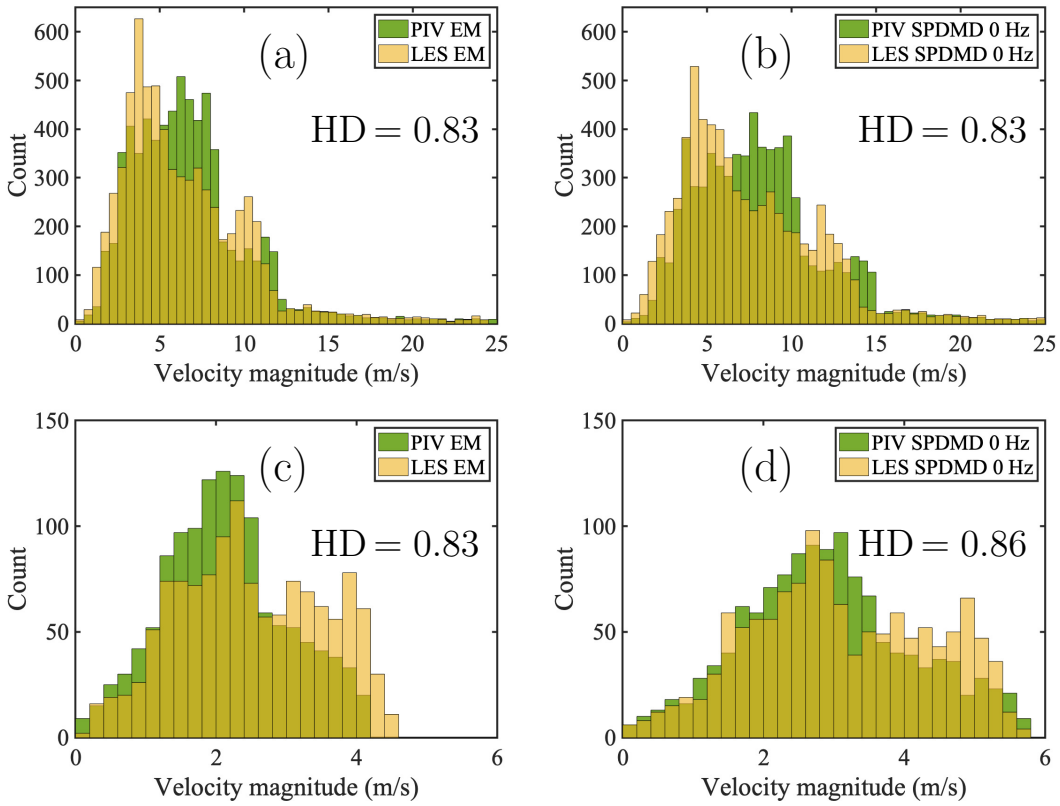

Figure 5.8: Histogram comparisons between **(a)**: the PIV and LES EMs at 470 CAD aTDCf, **(b)**: the PIV and LES SPDMD 0 Hz modes at 470 CAD aTDCf, **(c)**: the PIV and LES EMs at 700 CAD aTDCf, **(d)**: the PIV and LES SPDMD 0 Hz modes at 700 CAD aTDCf

.

realisations are needed for the particular condition to ensure that the LES data is statistically converged [44].

However, for the existing data, SPDMD shows a significant improvement in preserving the flow field magnitude information compared to the ensemble mean. For the PIV data, HD = 0.88, and for the LES data, HD = 0.89. The difference is more pronounced at 700 CAD, where the HDs are increased to HD = 0.93 for the PIV data and HD = 0.85 for the LES, as shown in Figure 5.4. The increases in HD due to the SPDMD 0 Hz modes are consistent with the previous findings on the Oxford optical engine presented in Chapter 4, where it was found that the use of the SPDMD 0 Hz modes

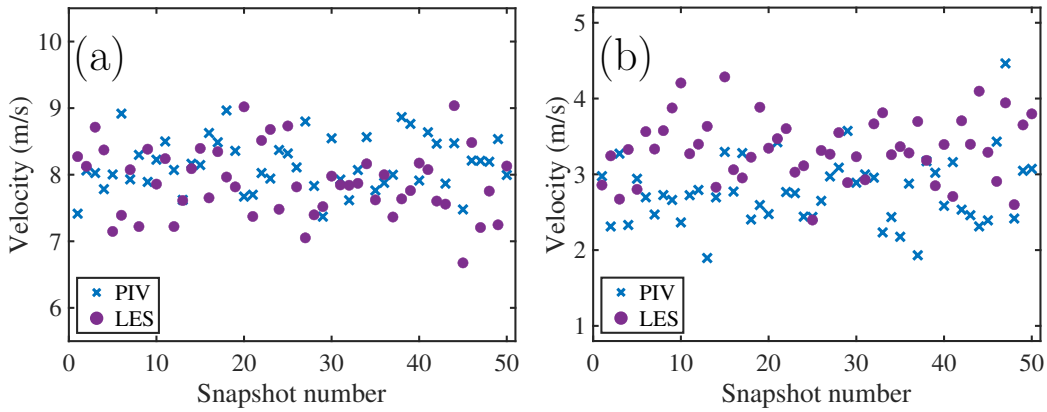

Figure 5.9: Planar-averaged velocity magnitudes for the first 50 snapshots of PIV data (blue crosses), and all 50 of the LES snapshots (purple circles) for **(a)**: 470 CAD aTDCf, and **(b)**: 700 CAD aTDCf. Only 50 PIV snapshots are shown for clarity.

was able to increase the HD from 0.57 with the EM to 0.89 with the SPDMD 0 Hz modes.

5.2.3 Comparison of EM and SPDMD representations

The impact of using either the EM or the SPDMD 0 Hz fields for comparing the PIV and LES data is shown in Figure 5.8. In the case of 470 CAD, the overall shapes are the same regardless of whether the EM or SPDMD 0 Hz modes are chosen; indeed, HD = 0.83 for both cases. This is because the SPDMD increased the HD of both the LES and PIV data by 0.07, so the histograms have been stretched along the $x-$axis by similar amounts in the SPDMD comparison. However, for the 700 CAD case, the SPDMD 0 Hz modes have re-scaled the PIV and LES data by different amounts, leading to the HD being 0.82 when EM fields are compared, but 0.86 when SPDMD 0 Hz fields are considered. This is further evidence that the variabilities of the PIV and LES data are more similar at 470 CAD than at 700 CAD.

To test this hypothesis more directly, planar averages were taken of the velocity magnitudes for each of the single LES and PIV snapshots. The average velocities for

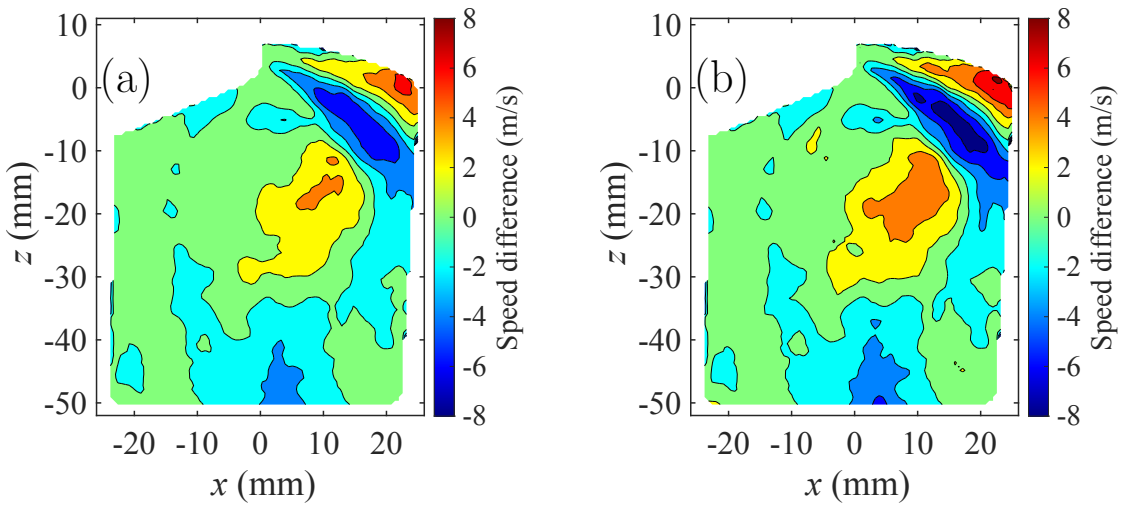

Figure 5.10: Plots showing the differences in velocity magnitude between the PIV and the LES data for **(a)**: the PIV and LES EM fields, and **(b)**: the SPDMD 0 Hz modes at 470 CAD aTDCf. The speed difference is calculated as PIV speeds − LES speeds, so that positive values represent regions where the LES under-predicted the PIV data, and negative values indicate over-predictions by the LES. Each contour line represents 2 m/s of speed difference.

each LES snapshot are plotted alongside the first 50 PIV snapshots in Figure 5.9 for both crank angles. At first glance, the planar average velocity magnitudes appear to have a similar spread for both PIV and LES datasets at 470 CAD. Indeed, a quantitative comparison reveals that the relative standard deviations were similarly low at 4.6% for the PIV data and 6.6% for the LES. This supports the findings of Barbato et al. [44], who found that these experimental and LES datasets had similar levels of variability in the peak cylinder pressures.

For 700 CAD, the relative standard deviations were more disparate, at 26.4% for the PIV data and 12.6% for the LES data. As a result, the SPDMD 0 Hz modes stretched the PIV histogram by a larger amount than the LES histogram, as shown in Figure 5.8. Therefore, the size of the improvement in HD due to the SPDMD does appear to be affected by the level of variability in a dataset; furthermore, if two datasets have velocity magnitudes with different levels of variability, then it could be

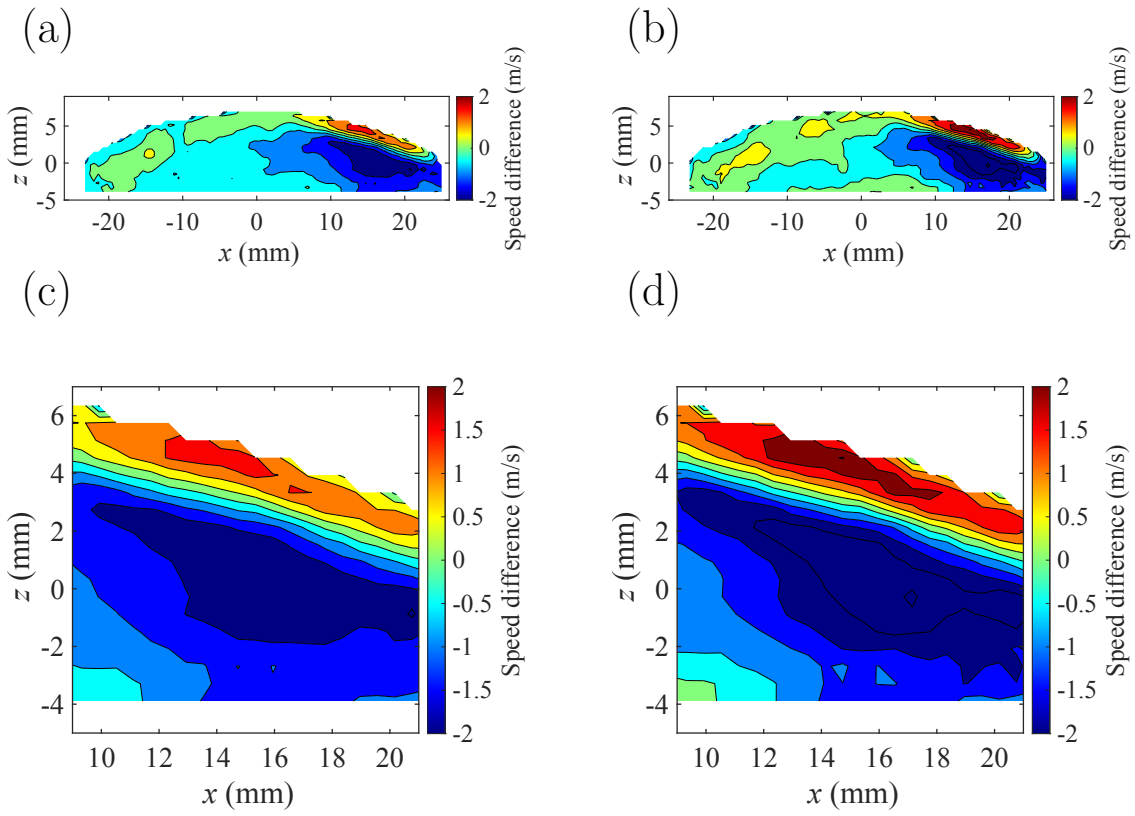

Figure 5.11: Plots showing the differences in velocity magnitude between the PIV and the LES data for **(a,c)**: the two EM fields, and **(b,d)**: the two SPDMD 0 Hz modes at 700 CAD aTDCf. The speed difference is calculated as PIV speeds − LES speeds, so that positive values represent regions where the LES under-predicted the PIV data, and negative values indicate over-predictions by the LES. Each contour line represents 0.5 m/s of speed difference.

expected that the SPDMD would increase the magnitudes of one dataset more than the other, resulting in a change in the HD between the two SPDMD 0 Hz modes relative to the HD between the two EM fields.

The effect of these re-scalings is demonstrated more clearly in Figures 5.10 and 5.11, which show contour plots of the speed differences between the LES and PIV data at each point in the grid. Figure 5.10**(a)** compares the LES EM and PIV EM fields at 470 CAD, while Figure 5.10**(b)** compares the respective SPDMD 0 Hz modes. Positive values indicate that the LES results under-predicted the PIV data, while negative values indicate under-predictions by the LES.

Similar regions of discrepancy are highlighted in both figures, with the largest speed differences being around the intake jet region, towards the top-right of the flow fields. However, the magnitudes of these discrepancies are different, even for a case where the SPDMD 0 Hz modes have rescaled the velocity magnitudes by similar amounts. For example, at 470 CAD, a comparison between EM fields would result in the conclusion that the LES results under-predicted the PIV data by at most 6 m/s in the region $(20, 0)$, whereas the under-predictions reach as high as 8 m/s in the same region when the SPDMD 0 Hz modes are considered. In a similar vein, the EM field comparison shows that the LES over-predicted the PIV by 5 m/s in the region $(15, -10)$, while the SPDMD 0 Hz modes indicate that this over-prediction was actually 7 m/s. This has consequences for the development of LES models, where it is important that sources of variability are correctly diagnosed, and that any discrepancies are accurately quantified in order to ensure that LES models are faithfully capturing the dynamics of the physical experiments. Similar results can be seen for 700 CAD, with the EM and SPDMD 0 Hz comparisons yielding different results for the level of discprepancy in velocity magnitude predictions, particularly in the vicinity of the tumble vortex to the right of the flow field.

5.2.4 Single LES snapshots

The effect of using SPDMD 0 Hz modes to represent data rather than the EM is also apparent when attempting to compare single LES snapshots against an ensemble of PIV data. The velocity histogram for the single LES snapshot shown in Figure 5.1**(d)** is plotted alongside the PIV EM and PIV SPDMD 0 Hz histograms in Figure 5.12. Here, the results look quite different; the comparison between the single LES snapshot and the PIV EM is more disparate with HD = 0.71, while the comparison against the SPDMD 0 Hz mode gives a closer match with HD = 0.84. Therefore, this LES snapshot is more representative of the PIV ensemble than it would have seemed if

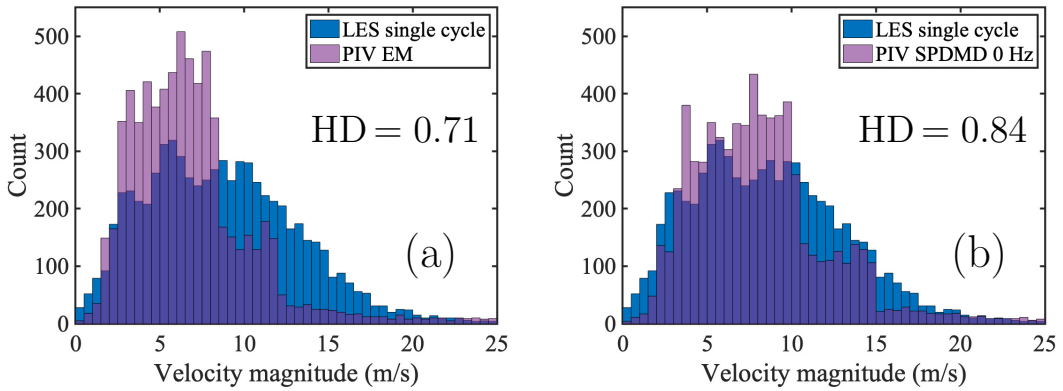

Figure 5.12: Histogram comparisons between **(a)**: a single snapshot from LES and the PIV EM, and **(b)**: a single snapshot from LES and the PIV SPDMD 0 Hz mode, all at 470 CAD aTDCf

.

the EM had been used as the sole validation target. This has implications for how particularly unrepresentative LES snapshots should be identified.

For example, consider a comparison between Figure 5.1**(a)**: the PIV EM, and **(d)**: the LES single snapshot. The low HD of 0.71 between these two flow fields would identify this LES snapshot as being especially unrepresentative of the PIV data. In particular, the length of the intake jet and the higher-speed recirculation zones may be highlighted as specific problem areas. However, if the SPDMD 0 Hz mode in Figure 5.1**(e)** is chosen as the benchmark representing the PIV data, the opposite conclusions about the LES snapshot may be drawn. In fact, this snapshot turns out to be one of the more representative ones within the LES dataset in terms of velocity magnitudes. This highlights the fact that care needs to be taken when using the EM as the validation target for LES results, and when drawing conclusions about the accuracy of velocity magnitude predictions.

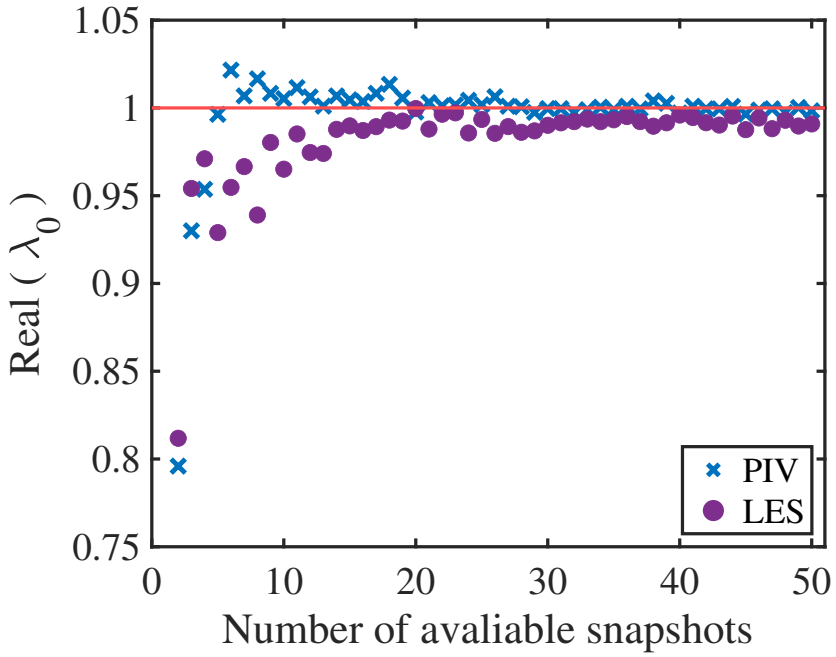

Figure 5.13: Convergence of the real part of the SPDMD 0 Hz eigenvalues (denoted as Real(λ_0)) to unity for an increasing number of snapshots in the dataset at 470 CAD aTDCf. The convergence behaviour for the PIV data is given by the blue crosses, the LES data by the purple circles, and the line Real(λ_0) = 1 is plotted in red.

5.2.5 Effects of dataset size

The effect of the number of available snapshots on the SPDMD 0 Hz modes was also explored, as this was a key difference between the LES and PIV data (consisting of 50 and 250 snapshots respectively). The convergence of the real parts of the DMD eigenvalues and the calculation of the mode residuals are two widely used metrics for analysing the convergence of DMD spectra [99, 232]. For a statistically stationary process, the real parts of the eigenvalues will converge to unity, so this is an appropriate test for assessing the validity of the 0 Hz SPDMD modes. On the other hand, the residual calculation gives a more complete picture of DMD convergence, as it quantifies the difference between the reconstruction of the last snapshot and the actual data [232]. However, this work focusses purely on the 0 Hz SPDMD mode, and a reconstruction of the data outside of the 0 Hz mode is not required. Therefore, the real parts of the DMD eigenvalues are used as the convergence metric instead.

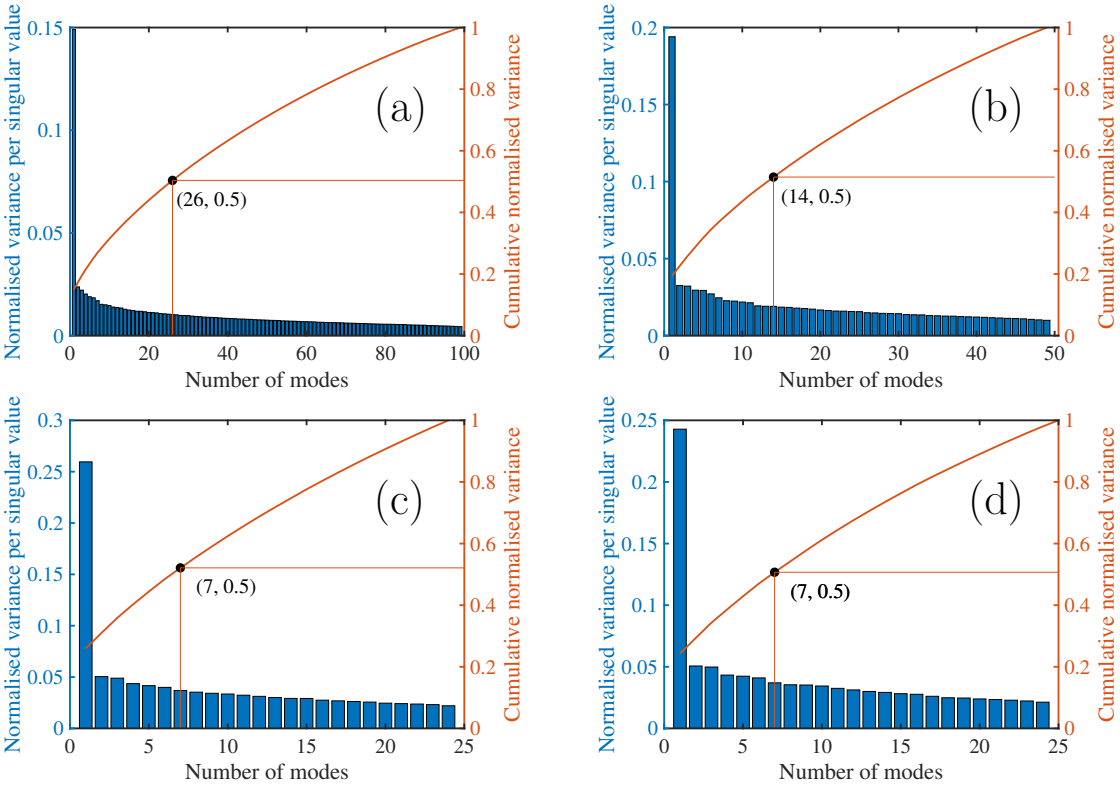

Figure 5.14: Singular value decomposition plots for **(a)**: the first 100 PIV snapshots, **(b)**: the first 50 PIV snapshots, **(c)**: the first 25 PIV snapshots, and **(d)**: the first 25 LES snapshots, all at 470 CAD aTDCf. The variance captured in each singular value is given by the bar chart on the left axis, and the cumulative variance is given by the line graph on the right axis. The number of modes corresponding to 50% of the cumulative variance is marked for each case.

Figure 5.13 plots the real parts of the eigenvalues associated with the 0 Hz SPDMD modes (λ_0) for an increasing number of available snapshots. Specifically, separate λ_0 values were calculated for the first m and $m+1$ snapshots and analysed accordingly. Because the number of snapshots varied for each calculation, it was impractical to fix a constant SPDMD mode threshold number for each case. Instead, a criterion was chosen such that the threshold numbers were set to the number of singular values that contained at least 50% of the cumulative variance in the data. For example, when the first 20 LES snapshots were considered, the 6 largest singular values accounted for 52.8% of the cumulative variance in the data. Therefore, 6 was chosen as the SPDMD mode threshold number for $m = 20$. 50% was chosen as the variance threshold as it

Table 5.1: The impact of sample size on the proportion of the total variance contained in the first singular value, and the number of singular values needed in order to capture 50% of the cumulative variance in the data (denoted as SV50) for 470 CAD aTDCf.

	Variance in first singular value	SV50
PIV 250	0.11	57 (23%)
PIV 100	0.15	26 (26%)
PIV 50	0.19	14 (28%)
PIV 25	0.26	7 (28%)
LES 50	0.18	14 (28%)
LES 25	0.24	7 (28%)

is known to produce reliable SPDMD 0 Hz modes for these datasets over a number of different crank angles; however, different variance threshold criteria ranging from 10-95% were also tested and found to have a negligible effect on the overall convergence patterns.

In Figure 5.13 it can be seen that convergence to unity is achieved with relatively small dataset sizes, indicating that SPDMD can produce reliable 0 Hz modes for datasets containing as few as 20 snapshots for both the LES and PIV data. Convergence to the unit circle indicates that there are enough snapshots to generate a steady state signal, and that the total time spanned by the dataset at approximately 20 cycles is longer than the slowest characteristic time scale in the data [232]. This result is consistent with the findings of Grenga et al. [232], who investigated direct numerical simulation (DNS) results of a turbulent planar premixed hydrogen/air jet flame with DMD. Their results showed that real(λ_0) changed by negligible amounts for dataset sizes varying from 21 snapshots to 401.

Further discussion is warranted regarding the decision to compare 250 PIV snapshots to 50 LES realisations. The full 250 PIV snapshots were utilised for the majority of

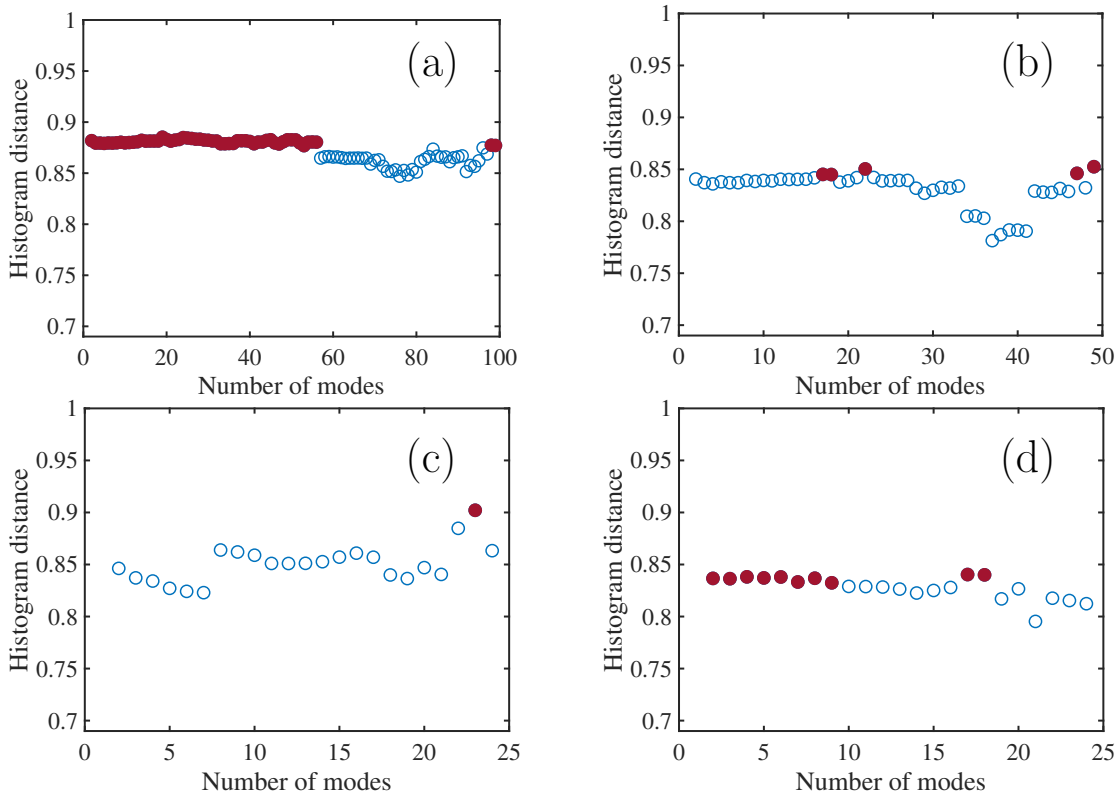

Figure 5.15: Sweeps calculating the HD between the SPDMD 0 Hz mode and the cycle average histograms for varying mode threshold numbers for **(a)**: the first 100 PIV snapshots, **(b)**: the first 50 PIV snapshots, **(c)**: the first 25 PIV snapshots, and **(d)**: the first 25 LES snapshots, all at 470 CAD aTDCf. The filled red circles represent HD values that are within 1% of the maximum HD value.

this study in order to make the most of the available data, as the dataset sizes are relatively small and therefore caution should be taken when discarding information by only considering a smaller subsample. However, the effect of a smaller dataset size was also tested by subsampling the PIV data into the first 25, 50 and 100 PIV snapshots. The singular value decompositions and HD sweeps are shown for each of the PIV subsamples as well as for the first 25 LES cycles in Figure 5.14.

The PIV data behave more similarly to the LES data when 50 snapshots are considered for each. For example, the amount of variance contained in the first singular value and the number of singular values needed to reach 50% of the cumulative variance (denoted here as SV50) are presented in Table 5.1. For the present engine, as

more snapshots are considered, the proportion of the variance that is contained in the first singular value decreases, and the value of SV50 is also reduced as a percentage of the total number of singular values. The reduction in the relative SV50 suggests that some redundancy is entering the dataset, as the raw SV50 values increase at a slower rate than the total number of snapshots. The fact that fewer modes are needed to capture 50% of the variance in the data (relative to the total number of snapshots) may indicate that the data are becoming more converged. The decrease in the variance contained in the first singular value may be due to the fact that larger datasets are more complex overall, and therefore more challenging to capture in a single mode. However, a more thorough investigation into the various possible sampling strategies would be required in order to verify these hypotheses.

The corresponding HD sweeps for varying mode threshold numbers considering subsamples of the PIV and LES snapshots are shown in Figure 5.15. There does not seem to be a pattern in the distribution of the HDs with varying sample size when all of the HD plots are considered together, so it is likely that hard thresholding the SPDMD reconstructions will remain a case-dependent exercise. However, as previously stated, a larger statistical analysis would need to be conducted before a firm conclusion can be drawn.

What are the implications of dataset size on RI / similarity to the experimental fields? * Validation of the CFD is left to Modena. * Variances may be different * If the CFD is more accurate, maybe the behaviour of the SVD plots would be more similar as well. * Fontanesi noted that the LES results were well convereged, but variances were indeed quite different.

5.3 Discussion

The results in this work show that the SPDMD 0 Hz modes can provide more realistic representations of vector magnitudes than the EM for a dataset that is characterised by some degree of variability. For the application of validating LES data against PIV data, this is shown to be useful in a number of ways, listed below.

1. Gaining insight into the dynamics of the physical system, as shown in Figures 5.1 and 5.3.

2. Quantifying differences in vector magnitudes between simulations and experiments, as shown in Figures 5.10 and 5.11.

3. Identifying LES snapshots that are particularly representative of the PIV ensemble, as shown in Figure 5.12.

Point 1 refers to the more representative vector magnitudes that are displayed in the SPDMD 0 Hz modes, for example as shown in Figures 5.1 and 5.3. The figures indicate that an isolated consideration of the EM fields could result in a misleading conclusion about what the speed of the intake jet should be. One could argue that this 'error' cancels out if the EM is used to compare both datasets, as the EM diminishes the intake jets in the PIV and LES data by similar amounts. However, this is not guaranteed to be the case, especially if the variability in one dataset is significantly different from another. For example, for the 700 CAD case, the relative standard deviation in the planar-averaged velocity magnitudes was 12.6% for the LES data, but 26.4% for the PIV data. This difference in variability leads to a more significant difference in the HD between the EM fields and SPDMD 0 Hz modes, as shown in Figure 5.8.

Therefore, the SPDMD 0 Hz modes can provide increased confidence in the velocity magnitudes given by the flow fields that are used to represent the ensembles of data. A note is needed here regarding the use of SPDMD 0 Hz modes and the quantification of variability in the data, or CCVs. Although the size of the increase in the HD to the set of individual snapshots due to the SPDMD is affected by the level of variability in the dataset, other more direct methods for calculating CCV would likely be more suitable, such as quantifying the pressure variation [233] or through a POD analysis [234]. Rather, the primary purpose of the SPDMD 0 Hz modes here is to provide a more reliable representation of a vector field ensemble for use as a validation target.

The contributions of this work also provide evidence for the robustness of the proposed application of SPDMD across different datasets. With the results in this chapter, alongside the previous one, the approach has been shown to yield increased HDs to the sets of individual snapshots for both LES and PIV data, representing a variety of physical behaviours across different engines and phase angles. Within the realm of CFD validation, the most widely-used method in prior works, the POD, is perhaps better suited to the quantification of in-cylinder flow CCVs or the comparison of isolated POD modes (ie. comparing like POD modes across datasets) as conducted by Barbato et al. [44], rather than using the POD-based reconstructions as validation targets. On the other hand, the SPDMD 0 Hz modes offer an objective many-to-one solution for the formation of a validation target from an ensemble of data, that resembles the individual snapshots of the physical flow, and without the diminishing of vector magnitudes that is associated with ensemble averaging.

An ongoing challenge with this method lies in how to objectively define the threshold for the number of modes to retain in the SPDMD analysis. Currently, it appears

that the best practice is to simply test a number of different thresholds and select one with a high HD to the individual snapshots, as shown in Figure 5.5. The thresholding challenge is common to many methods with roots in the singular value decomposition (SVD), including POD. In some cases, a hard threshold can be defined where there is a clear separation between the high- and low-variance modes in the form of an 'elbow' in the plot of singular values [69], but this is rarely the case for turbulent engine flows [106]. Gavish and Donoho [217] provide a theoretical basis for finding the optimal hard threshold for a low-rank matrix with Gaussian noise, but it is unclear how well this translates to a dataset with an unknown level of noise, and other variable factors such as CCVs.

5.4 Conclusion

In this chapter, the diminished magnitudes effect was quantified for the PIV and LES data from a different engine, also using the HD. While the EM fields under-represented the magnitudes seen in both the PIV and LES datasets, the SPDMD 0 Hz modes were able to remove the under-predictions and yield higher HD values to the original data, indicating that the SPDMD 0 Hz modes were more representative of both datasets, showing that the findings in the previous chapter possess a degree of generalisability.

At 700 CAD, use of the SPDMD 0 Hz modes had an impact on the comparison of HDs between the PIV and LES datasets. However, at 470 CAD, the SPDMD 0 Hz modes increased the HD for both the PIV and LES datasets by similar amounts, indicating that these two datasets had similar levels of variability in the velocity magnitudes. As a result, a comparison between EMs gave similar HDs to a comparison between SPDMD 0 Hz modes. Even so, the two methods provided different estimates for

the sizes of the speed differences across the LES and PIV data. In this case, the SPDMD 0 Hz modes showed that the LES data tended to mis-predict the velocity magnitudes around the intake jet region by larger amounts than it would appear from the comparison of EM fields. Accurate quantification of these mis-predictions is necessary in order to correctly diagnose inaccuracies in velocity data and aid in the development of LES models.

The importance of having a validation target that fairly represents the PIV data was also investigated by exploring the different conclusions that could be drawn about the accuracy of a single LES snapshot. For the LES snapshot chosen, a comparison to the PIV EM indicated that the velocity magnitudes in the LES snapshot were fairly unrepresentative of the PIV data, with a relatively low HD = 0.71. However, this same LES snapshot was shown to have more representative magnitudes if compared with the PIV SPDMD 0 Hz mode, where the HD increased to 0.84. The choice of validation target is therefore an important factor in attempting to assess the accuracy of individual LES snapshots and locate sources of error.

A convergence test was run to investigate how the number of snapshots in the dataset affected the SPDMD 0 Hz modes. For both the PIV and LES data, the SPDMD 0 Hz modes converged after approximately 20 snapshots. This suggests that SPDMD is able to produce a reliable 0 Hz mode for relatively small dataset sizes. Finally, a closer investigation of the thresholding issue was undertaken, where a link between the threshold criterion and the proportion of cumulative variance stored in the singular values was attempted. This behaviour was explored across several different dataset sizes, and ultimately no pattern was found between the proportion of variance in the singular values and the ideal SPDMD mode threshold in a HD sense. While the HD focusses on the vector magnitude distributions, the overall cumulative singular value

variance will also include the vector directions, and there might not necessarily be a strong correlation between their respective variabilities. At present, it is suggested that the best strategy for the choice of mode threshold is to treat it similarly to the γ parameter, and conduct a sweep of different thresholds to identify the reconstruction that produces the highest HD to the ground-truth.

To summarise, in this chapter the SPDMD method is shown to be capable of producing a single validation target from ensembles of PIV and LES data that accurately represent the vector magnitudes in the original data. The consistent results for a new engine suggest that this methodology has a degree of robustness for different set-ups. A comparison between SPDMD fields from PIV and LES data has a larger impact on the quantification of the speed differences between the two, rather than the overall magnitude distributions. This SPDMD method is therefore shown to be more objective and accurate in creating validation targets than the POD and EM. Additional insight into the datasets is also provided via an investigation of variability and dataset size. However, like the POD, some subjectivity still remains regarding the choice of mode threshold, which can likely only be alleviated by moving away from SVD-based methods and towards optimsation routines in isolation.

Chapter 6

Enhancing validation data via flow reconstruction

6.1 Overview

Whereas the previous two chapters investigated the creation of validation targets from data, this chapter is mostly concerned with the enhancement of validation data itself. This is because although experimental PIV data are often used as the ground-truth for CFD flow field validation, PIV images of internal flows can often contain gaps which hinder the use of techniques such as modal decomposition and the EM as well as the CFD validation process overall (see §1.5.1). This chapter is therefore focussed on the use of inpainting methods in order to reconstruct the turbulent flow inside such gappy regions in PIV images, which would enable the contstruction of full field-of-view validation targets, facilitating more complete comparisons with CFD data.

In many of the previously-published works that develop inpainting methods for turbulent flows, the findings are restricted to specific datasets that are not open-source, which inhibits reproducibility as well as the fair comparison of different inpainting methods [117, 118, 126]. Moreover, in order to test the accuracy of inpainting models, it is necessary to introduce artificial gaps into clean data so that predictions can be compared to a ground truth. However, there has been little thought given to how well these artificial gaps reflect reality, and therefore how well these inpainting approaches would perform in practice. This motivates an investigation into how artificial gaps that reflect common physical scenarios affect inpainting results, along with an open-source engine-relevant dataset, so that inpainting models can be reproducibly evaluated and used by the engine research community. The work in this chapter contributes towards these aims with the creation of the EngineBench database and the accompanying benchmark performances of five widely-used inpainting models. A website was created to host this work, containing all the necessary data and code: `https://eng.ox.ac.uk/tpsrg/research/enginebench/`.

6.2 Inpainting benchmark method

This section describes the establishment of the new benchmark used to test and diagnose the inpainting models. The components of the benchmark include the open-source dataset, the inpainting task, the inpainting models, and the metrics used to assess them. Each of these components are described in the following subsections.

6.2.1 Data and subset

The proposed EngineBench database consists of PIV data from the TCC-III engine, introduced in §2.5.3. The full database contains over 400,000 PIV images, coming to a size of 31 GB, as listed in Table 6.1. The dataset is stored on Kaggle as a series

of h5 files, as the natively hierarchical format simplifies the chunking of data so that train/validation/test splits can be separated by specific phase angles or test points. Also, h5 files have the capability for lazy loading, and the binary file format allows for efficient data storage. A diagram illustrating the hierarchical structure of each h5 file is given in Figure 6.1.

Table 6.1: Key EngineBench dataset information.

	EngineBench	**EngineBench LSP small**
Type	2D PIV	2D PIV
Engine	TCC−III	TCC−III
Pressures (kPa)	40, 95	40
Engine speeds (rpm)	800, 1300	1300
PIV planes	Lower swirl Tumble Cross-tumble	Lower swirl
Crank angles	40−705	90, 135, 180, 225, 270
# Snapshots	419,334	5,205
Size	31 GB	408 MB

In order to accelerate the training times, as numerous model configurations (44 in total) needed testing for the benchmark, a subset of the EngineBench data was constructed and used to generate the results. The use of a subset also makes the benchmarking results more accessible to researchers with smaller memory computers, and informs practitioners on how the ML models perform with smaller datasets. The subset, named EngineBench LSP small, was constructed solely using data from the lower swirl plane (LSP), as the field of view remains constant with the changing crank angle position, simplifying the analysis. Five crank angles are extracted at phases of interest throughout the engine cycle at one operating point, as presented in Table 6.1. EngineBench LSP small therefore contains 5205 PIV snapshots in total, and is also hosted on Kaggle, accompanied by tutorial notebooks to demonstrate how the data can be interacted with. An example PIV image from the subset is provided in

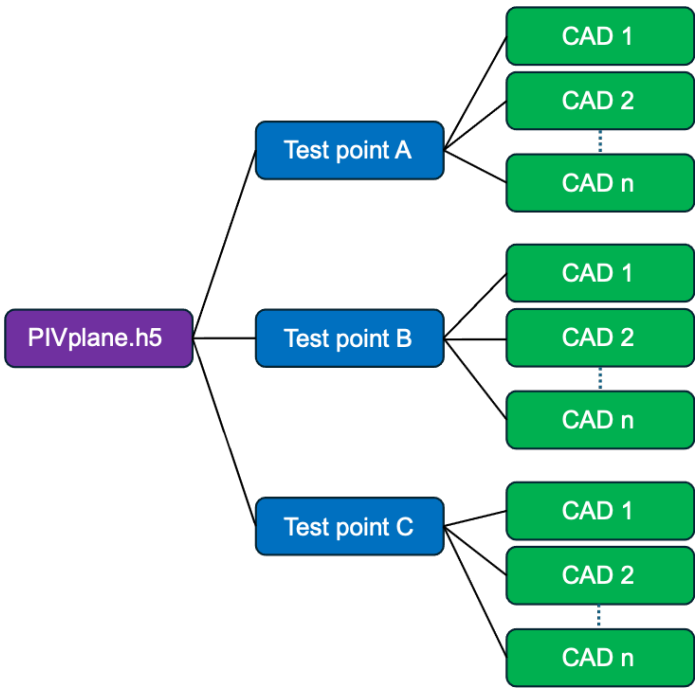

Figure 6.1: Generalised h5 file structure in EngineBench.

Figure 6.2. Finally, the original spatial dimensions for each image are 50 × 49 pixels. Zero padding is therefore added around the edges of the images to 128 × 128 for compatibility with standard ML models.

6.2.2 Target

The goal of the benchmark was chosen to be the inpainting of so-called 'edge gaps'. In this thesis, edge gaps are defined as large blocks of missing data at the edges of the field of view. This type of gap was selected for a number of reasons. Firstly, they are more realistic than other types of gaps such as randomly-located blocks; edge gaps commonly occur in PIV setups that have restricted optical access due to walls [235, 236]. In addition, it is especially challenging to predict the flow inside edge gaps, as there is a limited amount of local information that can inform the models. From the model's perspective, predicting the flow inside edge gaps is therefore akin to extrapolation beyond the field of view. This difficult challenge causes widely-used

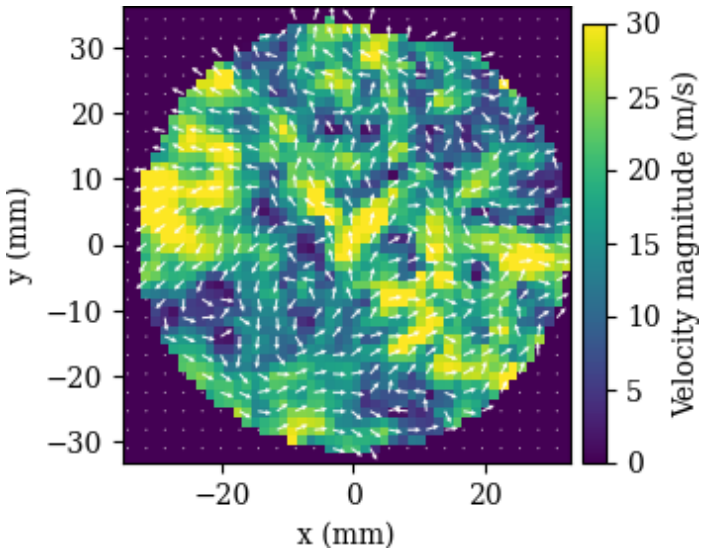

Figure 6.2: Example PIV image from EngineBench LSP small, showing a circular field of view. At each pixel the arrows show the direction of the turbulent flow, and the colourmap shows the velocity magnitude.

methods such as interpolation and gappy proper orthogonal decomposition (GPOD) to fail (see §6.3), and the development of solutions to this challenge is intended to push the boundaries of what is possible with flow field data reconstruction.

A consistent test case is therefore constructed using edge gaps to benchmark the performance of the inpainting models. Two masks of a fixed shape are constructed that each remove the data at a proportion of the pixels at the edges of the field of view. A vertical mask is applied to the first half of the test set, and a horizontal mask to the latter half. In addtition, two gap sizes are tested, consisting of 10% and 25% of the data missing. An example test flow field with 10% of the data missing is shown in Figure 6.3. Within EngineBench LSP small, all the data at 180 CAD aTDCf are held out for the test case in order to assess the generalisability of the models. The flow fields at 180 CAD are notoriously challenging to predict, as the piston is on the point of switching its direction of travel, causing the flow patterns to be highly variable [180].

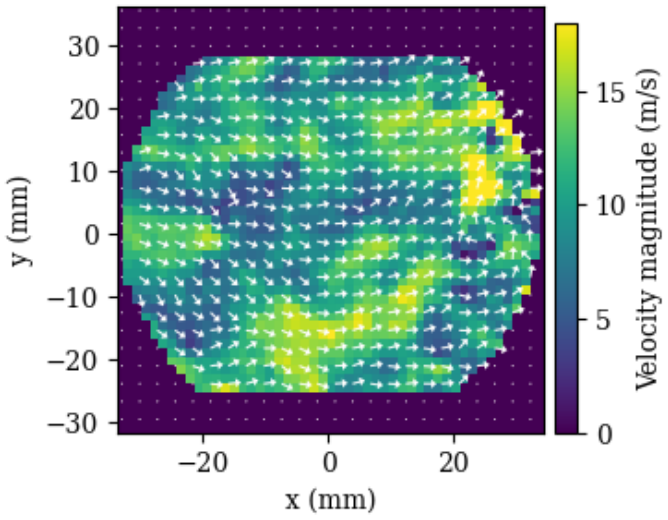

Figure 6.3: Example PIV image from EngineBench LSP small, with the horizontal edge gaps added at the top and bottom of the field of view.

6.2.3 Models and training

The performance of four different model architecures is benchmarked in this study. Firstly, adaptive median filter GPOD (GPOD-MF) is chosen as a best-in-class non-parametric approach, known to outperform interpolation and other GPOD methods [117]. Secondly, the UNet model [127] is chosen due to its wide usage in turbulent flow research. Two loss functions are tested with the UNet: a mean square error (MSE) loss, a huber loss function in order to test the effect of outliers in the PIV data, and a physics-based gradient loss. Further details of the loss functions are given in §6.2.5. Thirdly, the UNet transformer (UNETR) model [237] with a MSE loss is chosen due to the performance enhancements that have been reported due to the transformer module, with the project MONAI implementation [238]. Finally, an adapted version of the context encoder generative adversarial neural network (CE-GAN) [239] with MSE and adversarial losses is implemented due to its high performance in standard inpainting tasks [112, 240] and recent usage in turbulent flows [134, 143]. As the original context encoder was designed for inpainting gaps of a fixed size and location, the network architecture is modified in a similar fashion to the

changes made by Li *et al.* [134]. For the generator, an additional de-convolutional layer is included at the output to return a prediction of the same spatial dimensions as the input, forming a symmetrical autoencoder architecture. To correspond with the generator modifications, an extra convolutional layer is added at the beginning of the discriminator to handle inputs of the same size as the original data. A dropout layer with a probability of 50% is also added at the output of the discriminator, following Li *et al.* [134]. Model summaries implemented here are provided in Table 6.2 for reference.

Table 6.2: ML sizes in millions of parameters and loss functions tested.

Model	# Parameters (M)	Loss functions
GPOD	N/A	N/A
UNet	10.5	MSE, huber, gradient
UNETR	87.3	MSE
CE-GAN	74.0	Adversarial

During the ML model training, the losses between the predictions and the labels are calculated across the entire image, not just inside the gap. This approach provides a number of benefits: to simplify the random gaps training process, retain the context of the broader turbulent flow and field of view, and to provide practitioners with a visual representation of how the network relates the prediction inside the gap to the rest of the field, avoiding edge effects in the output. Performance metrics on the test set predictions are then reported for the central regions as well as the edge gap regions. Regarding the model details, all architectural hyperparameters were retained from their original studies. Training in all cases was run over 300 epochs, where an epoch is one complete pass through the training dataset by the model. For the UNet and UNETR models, the learning rate was 1e−3 and multiplied by a factor of 0.5 every 50 epochs via a step scheduler, in keeping with previous works [127, 147]. For the CE-GAN, the learning rate was 1e−4 and multiplied by 0.75 every 50 epochs. A number of different learning rate and schedule configurations were attempted in order

to find a suitable training dynamic between the generator and the discriminator, but it is challenging to assess the performance of a GAN; the two-player training dynamic typically renders the evaluation of a simple loss metric insufficient, often leading practitioners to rely on human inspection to validate the results [241, 242]. In the present scenario, the GAN was trained until it reached a point where the results looked physically correct, but it is quite likely that improved performance could be achieved with more advanced training techniques such as introducing 'warm-up' training phases and making more bespoke modifications to the model architecture [241, 243]. As the focus of the present work is on the benchmark creation, further GAN modifications are left for future work, but the potential for the GAN results to be sub-optimal should be kept in mind when assessing the results presented here.

Table 6.3: Definitions of phase angle permutations that comprise the training, validation, and hold-out test sets. The different permutations are denoted as **A**, **B** and **C**, and the corresponding phase angles are given in crank angle degrees (CAD).

	A	**B**	**C**
Train	90	135	90
	135	225	225
	225	270	270
Validation	270	90	135
Test	180	180	180

Finally, the training, validation, and testing datasets were split by crank angle. As previously mentioned, 180 CAD was held out for the test set, while different permutations of the other four phase angles are then used to construct the training and validation sets, with three phases for training and one for validation. The training process for each model was run three times with different permutations of training and validation phase angles tests, which tests the sensitivity of the model performances to the specific phases chosen for the analysis and provides the error bars for the results. The crank angle permutations are defined in Table 6.3. Only three permutations

are considered out of the possible four as the spread across permutations was found to be acceptably low on all metrics. The resultant number of images in the train, validation, and test sets are 3123, 1041, 1041.

GPOD implementation

GPOD is performed on a single large data matrix, rather than separate training and testing matrices. Therefore, in the present study, the test data matrix at 180 CAD is stacked alongside three training data matrices according to the permutations defined in Table 6.3. The test edge gaps are added to the test data matrix, and the gap locations are initialised using the ensemble mean from the training data matrices which do not contain any gaps. Convergence checking (CC) gaps are also implemented into the test data in a similar format to the test edge gaps; ie, the test edge gaps are extended to incorporate another 10% of the pixels in the image where the true values are known. The GPOD-reconstructed flow fields at the minimum CC L2 error are retained for analysis in this study.

6.2.4 Metrics

A variety of metrics are used to evaluate the model performances, in order to quantify pixelwise accuracy, vector similarity, and multi-scale phenomena. The relative L2 error is used to quantify pixelwise accuracy, and is calculated for true and predicted velocities $\boldsymbol{u}_{\text{true}}$ and $\boldsymbol{u}_{\text{pred}}$ as follows:

$$\text{L2} = \|\boldsymbol{u}_{\text{true}} - \boldsymbol{u}_{\text{pred}}\|_2 \,/\, \|\boldsymbol{u}_{\text{true}}\|_2 \tag{6.1}$$

where $\boldsymbol{u}_{\text{true}}$ and $\boldsymbol{u}_{\text{pred}}$ are the true and model-predicted velocity vectors respectively. In addition, two vector-based metrics are used to quantify the similarity of the overall

flow structures. The relevance index (RI) previously introduced in §4.2.1 is also reported here for convenience:

$$\mathrm{RI} = \frac{\langle \boldsymbol{u}_{\mathrm{true}}, \boldsymbol{u}_{\mathrm{pred}} \rangle}{\|\boldsymbol{u}_{\mathrm{true}}\|_2 \cdot \|\boldsymbol{u}_{\mathrm{pred}}\|_2} \tag{6.2}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product. The RI varies between 1 for perfectly aligned vectors, and -1 for perfectly opposite vectors. The similarity of the vector magnitudes is given by the magnitude index (MI) [62]:

$$\mathrm{MI} = 1 - \frac{\|\boldsymbol{u}_{\mathrm{true}} - \boldsymbol{u}_{\mathrm{pred}}\|_2}{\|\boldsymbol{u}_{\mathrm{true}}\|_2 + \|\boldsymbol{u}_{\mathrm{pred}}\|_2} \tag{6.3}$$

with the MI varying between 1 for vectors of identical magnitude and 0 for totally disparate vector magnitudes. Finally, in order to capture the multi-scale turbulent flow features, the energy spectrum S for each image is calculated using the Fourier transform:

$$S(\boldsymbol{k}) = \frac{1}{2}(\widetilde{\hat{\boldsymbol{u}}(\boldsymbol{k})\hat{\boldsymbol{u}}^*(\boldsymbol{k})}) \tag{6.4}$$

where $\hat{\boldsymbol{u}}(\boldsymbol{k})$ is the Fourier-transformed velocity vector, $\hat{\boldsymbol{u}}^*(\boldsymbol{k})$ is its complex conjugate, $\boldsymbol{k}$ is the spatial frequency wavenumber vector, and $\widetilde{(\cdot)}$ represents the radial average over the vertical and horizontal frequencies [134]. The Kullback–Leibler (KL) divergence is then used to quantify the similarity between energy spectra:

$$\mathrm{KL}(S_{\mathrm{true}} \| S_{\mathrm{pred}}) = \sum_k S_{\mathrm{true}}(k) \log\left(\frac{S_{\mathrm{true}}(k)}{S_{\mathrm{pred}}(k)}\right) \tag{6.5}$$

ranging from 0 for identical distributions to infinity for a complete divergence.

6.2.5 Loss functions

The widely-used mean-squared-error (MSE) loss l_{mse} between two pixels at location i is given by:

$$l_{mse} = \left(\boldsymbol{u}_{i,\text{true}} - \boldsymbol{u}_{i,\text{pred}}\right)^2$$

.

The Huber loss is a hybrid loss function that reduces sensitivity to outliers by applying an L1 loss to element-wise errors above a certain threshold (delta) and a quadratic loss otherwise to aid convergence. It is defined per pixel i as:

$$l_{huber,i} = \begin{cases} 0.5\left(\boldsymbol{u}_{i,\text{true}} - \boldsymbol{u}_{i,\text{pred}}\right)^2, & \text{if } \left|\boldsymbol{u}_{i,\text{true}} - \boldsymbol{u}_{i,\text{pred}}\right| < \delta \\ \delta * \left(\left|\boldsymbol{u}_{i,\text{true}} - \boldsymbol{u}_{i,\text{pred}}\right| - 0.5 * \delta\right), & \text{otherwise} \end{cases}$$

then averaged over all pixels in the image pairing. A smaller value of the δ parameter increases the influence of the L1 loss; the value of delta was tuned via a grid search of values presented in Table 6.5.

For the CE-GAN, the discriminator was trained using a binary cross entropy (BCE) loss, while the generator utilised a BCE / MSE hybrid. The BCE loss is defined as:

$$l_{bce} = -\boldsymbol{u}_{i,\text{pred}} * \log \boldsymbol{u}_{i,\text{true}} + \left(1 - \boldsymbol{u}_{i,\text{pred}}\right) * \log\left(1 - \boldsymbol{u}_{i,\text{true}}\right)$$

while the combined generator loss is given by:

$$l_{gen} = \lambda_{adv} * l_{bce} + \left(1 - \lambda_{adv}\right) * l_{mse}$$

where the adversarial ratio λ_{adv} controls the relative importance of the MSE and BCE losses. Following Li *et al.* [134], the sensitivity of four different adversarial ratios are

tested with results reported in Table 6.6.

Finally, the physics-based gradient loss is defined as [244]:

$$l_{phys} = \lambda_{grad} * l_{grad} + (1 - \lambda_{grad}) * l_{mse}$$

where the gradient ratio λ_{grad} controls the relative importance of the gradient and MSE losses. The gradient loss l_{grad} calculates the MSE between the gradients of the feature and target maps, and is defined in two dimensions (2D) as:

$$l_{grad} = l_{mse}\left(\frac{\partial \boldsymbol{u}_{i,\text{true}}}{\partial x}, \frac{\partial \boldsymbol{u}_{i,\text{pred}}}{\partial x}\right) + l_{mse}\left(\frac{\partial \boldsymbol{u}_{i,\text{true}}}{\partial y}, \frac{\partial \boldsymbol{u}_{i,\text{pred}}}{\partial y}\right)$$

where x and y are the two coordinate directions. A gradient loss function helps to preserve sharp transitions and edges which can be smoothed over when using the MSE. In addition, in the case of turbulent flow data, preservation of the gradients can encourage the model to align with physical quantities such as vorticity [147].

6.2.6 Data augmentation

One of the key considerations of this work is in how artificial gaps are introduced into the data to train the models. This can be handled via data augmentation at training time. Three different techniques were investigated in this work: introducing fixed horizontal and vertical edge gaps like the test case (fixed edge), blocks of various sizes and locations (random blocks), and edge gaps of random size and orientation (random edge gaps). Some example random block gaps are shown in Figure 6.4, where

the yellow regions indicate areas where data were removed from the snapshots.

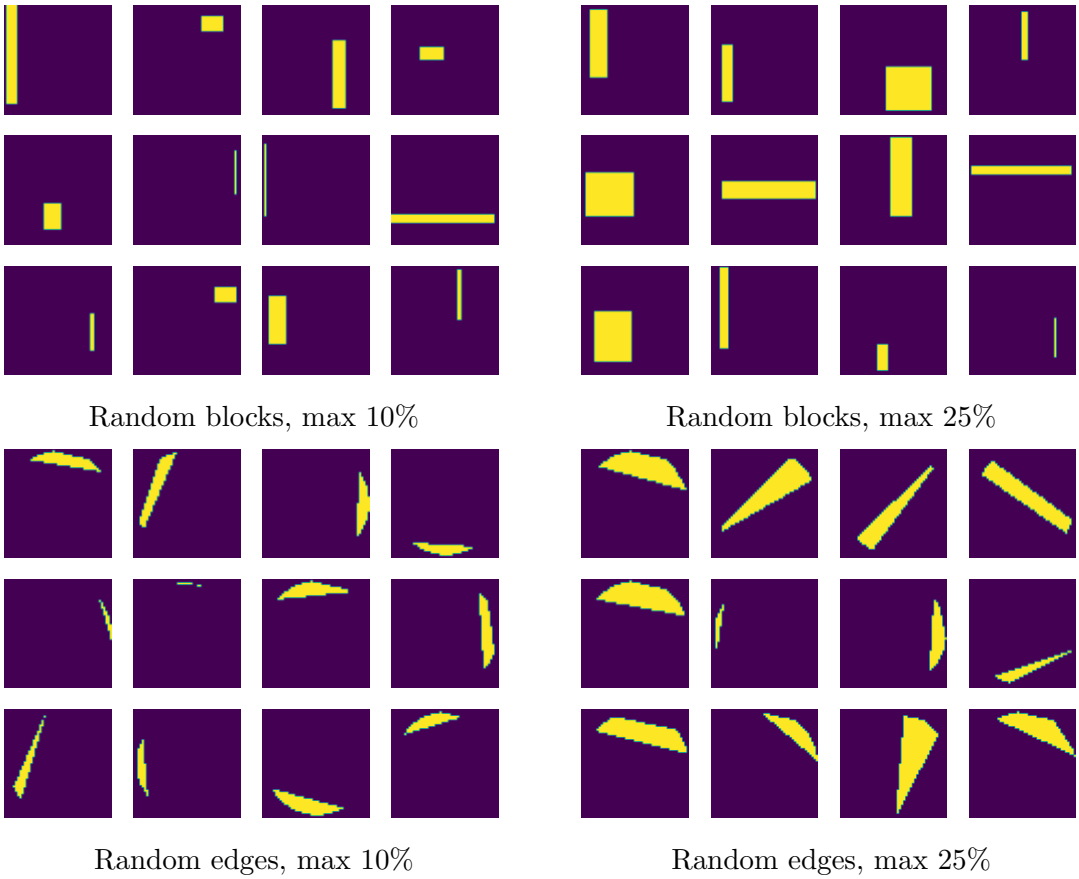

Figure 6.4: Samples of the randomly-generated block and edge gaps used to train the models in this study, for both 10% and 25% gap sizes.

The random edge gaps are constructed by taking four random points along the input image borders, drawing a polygon between the points, and masking out pixels that lie outside of the polygon. There can be a maximum of two points on any one edge. This approach ensures that edge gaps are created with random sizes and orientations, to prevent the models from overfitting to specific gap shapes and locations. A maximum percentage of the pixels are allowed to be removed by the mask; the mask is discarded if it removes more pixels than this, and a replacement mask is generated. This upper threshold for the gap sizes is needed to constrain the training process, prevent the inpainting task from becoming overly challenging, and reflect more realistic physical scenarios. A histogram showing the proportion of pixels removed for

each snapshot in one pass of the training set for 10% gaps is shown in Figure 6.5. A regular PIV snapshot is shown alongside two snapshots with random edge gaps added in Figure 6.6.

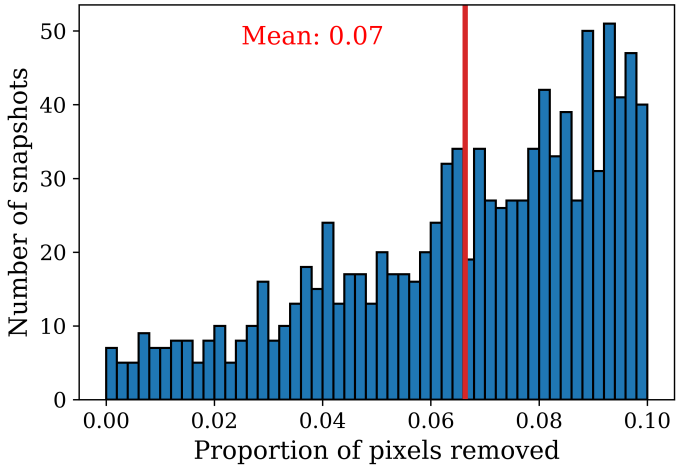

Figure 6.5: Histogram showing the proportion of pixels removed by the random edge masks in one pass through the training set. 7% of the total pixels in the field of view were removed on average.

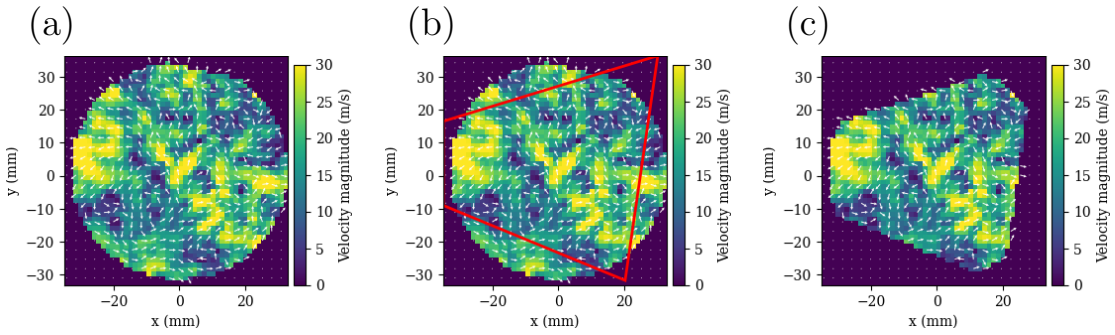

Figure 6.6: Example random edge gap creation. From left to right, (a): original image; (b): image with a random edge gap polygon superimposed in red; (c): edge gaps added to regions outside of the random polygon.

6.3 Results

The inpainting model results are presented in the next section. Optimal configurations for the data augmentation strategy, loss functions, and the GPOD convergence are presented at first. This is then followed by the key benchmark results, and the construction of validdation targets from the predicted flow fields.

6.3.1 Training gaps

Firstly, the different artificial gap generation strategies described in §6.2.6 were tested with the UNet model, in order to establish the optimal training pipeline. The results for the four metrics tested are given in Table 6.4, with separate reports for the central image regions and the gap regions. Overall, the accuracy inside the central regions is very high in all cases, with RI = 0.999 and a pixelwise L2 error of $\approx 3\%$. This shows that the UNets are able to preserve the information provided to it in the input to a very high degree, despite compressing the data through the bottleneck. As expected, the accuracy inside the gap regions is worse, as the UNet is required to extrapolate beyond the field of view that was supplied at the input. However, the results still appear to be passable, with RI up to 0.88 at the 10% gap size and 0.82 at 25%. Further discussion on the utility of these predictions is provided in §6.4.

Training the UNet on fixed edge gaps, which have the same form as the test gaps, generally produced the highest accuracies in the image centres. This simpler training strategy allowed the model to focus more on the global flow patterns provided at the input, as the location of the gaps did not change from image to image. This emphasis on general flow patterns helped to also yield the best KL divergences within the edges at both 10% and 25% gap sizes. However, the weaker RI and L2 scores at the edges indicate that over-fitting the model to the fixed mask shape prevented it from generalising as well to the more specific localised flow behaviour in the unseen crank angle. Conversely, for the random edge gaps, the addition of significant variability to the process made it more challenging for the UNet to learn the global flow distributions as precisely. However, the random edge training did improve the predictions of local details and vector orientations, with the strongest RI and L2 scores at the edges.

Table 6.4: Impact of training a UNet, MSE model on the fixed edges used as the testing gaps, random edge gaps, random block gaps, and a combination of the latter two. One result for each setup using permutation A is reported. The final figure represents the average over the 1041 test images. **Bold** typeface represents the best result.

	Gap size	RI	MI	L2	KL
Central regions:					
Fixed edge	10%	**1.000**	**0.986**	**0.027**	**0.000**
Random edge	10%	**1.000**	0.984	0.032	**0.000**
Random block	10%	**1.000**	0.984	0.031	**0.000**
Random block & edge	10%	0.999	0.983	0.035	**0.000**
Edge gaps:					
Fixed edge	10%	0.878	**0.754**	0.488	**0.009**
Random edge	10%	**0.886**	0.751	**0.463**	0.016
Random block	10%	0.844	0.703	0.528	0.028
Random block & edge	10%	0.879	0.747	0.473	0.010
Central regions:					
Fixed edge	25%	**0.999**	**0.984**	**0.033**	**0.000**
Random edge	25%	**0.999**	0.979	0.042	**0.000**
Random block	25%	**0.999**	0.979	0.043	**0.000**
Random block & edge	25%	**0.999**	0.979	0.043	**0.000**
Edge gaps:					
Fixed edge	25%	0.814	**0.695**	0.609	**0.021**
Random edge	25%	**0.819**	0.688	**0.569**	0.027
Random block	25%	0.803	0.674	0.588	0.035
Random block & edge	25%	0.813	0.683	0.577	0.030

The random blocks and combination of random blocks and edges were used to test whether a broader inpainting training process would help the model to generalise further. However, neither of these strategies produced higher accuracies than the fixed or random edge gaps in isolation. This shows that for this problem, the best performance can be achieved by providing the model with training and testing gaps that are of the same general shape and location; however, some randomisation within these general parameters via the random edge gaps did provide the strongest RI and L2 metrics inside the edge regions for both gap sizes. In addition, it is expected that

models trained on random edge gaps will be able to handle test cases with edge gaps at any orientation, unlike models trained on fixed gap positions. Due to this improved flexibility, combined with strong scores across all four metrics, the random edge gaps method was deemed to have the most practical utility among the data augmentation methods tested here. Therefore, the random edge gaps technique was used in the training pipeline to benchmark the other model configurations investigated in this thesis.

6.3.2 Loss functions

Table 6.5: Huber loss δ tuning results for the 180 CAD test case with 10% edge gaps. One result for each setup using permutation A is reported. **Bold** typeface represents the best result.

	RI	MI	L2	KL
Central regions:				
$\delta = 5$	0.999	0.983	0.033	**0.000**
$\delta = 1$	**1.000**	0.983	0.034	**0.000**
$\delta = 0.5$	0.999	0.983	0.034	**0.000**
$\delta = 0.1$	**1.000**	**0.988**	**0.024**	**0.000**
Edge gaps:				
$\delta = 5$	0.886	0.751	0.462	0.016
$\delta = 1$	**0.896**	**0.765**	**0.445**	**0.013**
$\delta = 0.5$	0.895	0.760	0.449	**0.013**
$\delta = 0.1$	0.858	0.712	0.517	0.026

Table 6.6: Adversarial loss lambda tuning results for the 180 CAD test case with 10% edge gaps. One result for each setup using permutation A is reported. **Bold** typeface represents the best result.

	RI	MI	L2	KL
Central regions:				
$\lambda_{adv} = 1e{-}1$	0.709	0.578	0.709	0.099
$\lambda_{adv} = 1e{-}2$	**0.883**	**0.731**	**0.478**	**0.014**
$\lambda_{adv} = 1e{-}3$	0.709	0.616	0.809	0.082
$\lambda_{adv} = 1e{-}4$	0.773	0.645	0.807	0.024
Edge gaps:				
$\lambda_{adv} = 1e{-}1$	0.575	0.513	0.817	0.160
$\lambda_{adv} = 1e{-}2$	**0.789**	**0.656**	**0.612**	0.043
$\lambda_{adv} = 1e{-}3$	0.695	0.604	0.855	**0.019**
$\lambda_{adv} = 1e{-}4$	0.755	0.616	0.929	0.021

Table 6.7: Gradient loss lambda tuning results for the 180 CAD test case with 25% edge gaps. One result for each setup using permutation A is reported. **Bold** typeface represents the best result.

	RI	MI	L2	KL
Central regions:				
$\lambda_{grad} = 0.5$	**0.999**	**0.979**	**0.043**	**0.000**
$\lambda_{grad} = 0.9$	**0.999**	0.971	0.059	0.001
$\lambda_{grad} = 0.99$	**0.999**	0.971	0.059	0.001
$\lambda_{grad} = 0.999$	**0.999**	0.969	0.062	0.001
Edge gaps:				
$\lambda_{grad} = 0.5$	0.809	0.687	0.581	**0.024**
$\lambda_{grad} = 0.9$	0.825	0.693	0.559	0.028
$\lambda_{grad} = 0.99$	0.821	0.684	0.566	0.027
$\lambda_{grad} = 0.999$	**0.834**	**0.696**	**0.549**	0.043

Use of the Huber, adversarial, and gradient loss functions require the tuning of parameters in order to determine suitable configurations. A grid search was performed in each case, following best practices laid out in previous studies [134, 147]. The Huber and gradient loss parameters (δ and λ_{adv}, respectively) were tuned within a UNet model, while the adversarial ratio λ_{adv} was employed within the CE-GAN. The results are presented in Tables 6.5 and 6.6. For the Huber loss, $\delta = 1$ exhibited the best performance in the edge gaps and was therefore used in the remainder of the study. The increased use of the L1 loss at the smaller $\delta = 0.1$ was less sensitive to large errors more likely to be found inside the gaps, but produced the best accuracies in the image centres where the reconstructions are closer to being correct.

For the adversarial ratio, $\lambda_{adv} = 1e{-}2$ yielded the best scores across three of the metrics in the edges and was chosen for the remainder of the CE-GAN results in this study. In this case, λ_{adv} strikes a balance between reconstructing an accurate output with the MSE loss, and fooling the discriminator with the BCE loss. The stronger RI, MI, and L2 but weaker KL divergence at $\lambda_{adv} = 1e{-}2$ relative to $\lambda_{adv} = 1e{-}3$

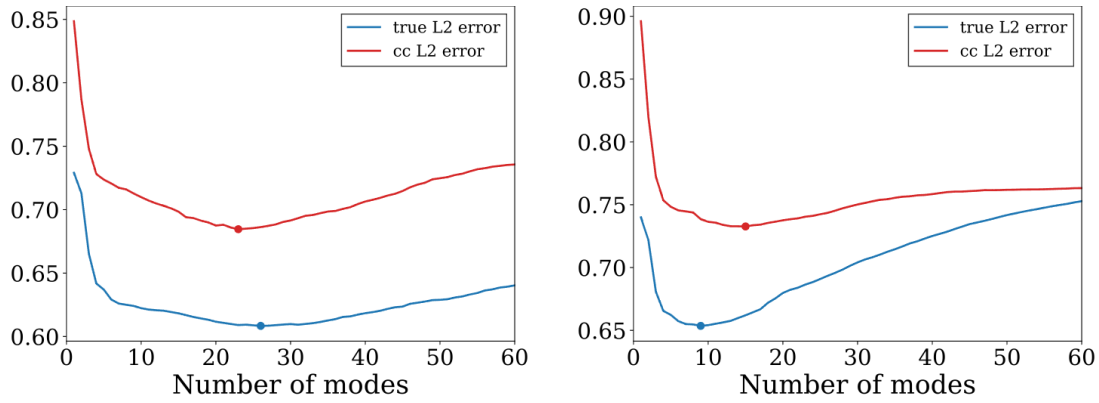

Figure 6.7: GPOD convergence curves for 10% and 25% gaps (left and right, respectively) at permutation A. The minimum errors for both the true L2 error in the edge gaps and the L2 error in the convergence checking (CC) gaps are marked as filled circles.

indicate that capturing finer details is more useful for overcoming the discriminator than the global velocity distribution. For the gradient loss, $\lambda_{grad} = 0.999$ exhibited the best results, as shown in Table 6.7, and was used by the UNet, gradient models for the remainder of the study. The λ_{grad} parameter behaves in a similar fashion to the adversarial ratio, with a trade-off between the KL divergence and the RI, MI, and L2 scores. In this case, more emphasis on the gradient loss helps to preserve global flow patterns, while the MSE is more effective in predicting local details such as vector orientations and pixelwise values.

6.3.3 GPOD convergence

As the final step before each of the model configurations can be benchmarked againsnt one another, the number of modes to be retained by the GPOD prediction is governed by the convergence criteria. Plots showing the GPOD convergence curves for 10% and 25% gaps in permutation A are shown in Figure 6.7. In both cases, the relative L2 error calculated in the convergence-checking (CC) gaps gives an optimal number of modes that is relatively close to the true optimum given by the L2 error in the true gaps. The number of modes retained in the final GPOD reconstructions benchmarked

here considered the true lowest L2 errors, and were 26 for the 10% gap size, and 9 for 25%. The lower number of modes in the latter case indicates that the GPOD algorithm relies on a reconstruction that contains more general flow patterns in order to optimise the error across the larger gaps in the different snapshots.

6.3.4 Main benchmark results

The results for the benchmark performance metrics are given in Tables 6.8 and 6.9, with the best result for each metric presented in bold. Loss curves for each model configuration are provided in Figure 6.8. The UNet and UNETR models exhibit similar performances across all metrics, with the UNet models slightly outperforming UNETR for predictions inside the edge gaps. As shown in Table 6.2, the number of parameters in the UNet architecture is eight times smaller than that of the UNETR model, so the UNet exhibits a better accuracy-complexity trade-off. This indicates that detailed local features and textures may be more predictive of the target outputs than global context in this situation, which runs counter to where UNETR models typically see performance gains [245–247].

The UNet variants each exhibit similar predictive performances in the edge gaps at the 10% gap size, although the gradient loss function demonstrates the best RI, MI and L2 metrics at 25% gaps. This is in line with the results of Chung *et al.* [147] who showed that the gradient loss provided persisting benefits for a super-resolution task of increasing difficulty from 8× to 32× magnification. On the other hand, in the present work, the UNet, gradient model yields higher KL divergences in the edges, especially at 25% gaps, as shown in Table 6.9. This shows that the gradient loss function emphasises local regions with large velocity gradients at the expense of the overall energy distribution in the flow. As with the investigation on data augmentation strategies and loss function parameters, this is another example of how

Table 6.8: Results for the 180 CAD test case at 10% gaps. The mean and standard deviations are reported from the three permutations of training data defined in Table 6.3. **Bold** typeface represents the best mean in each category separated by the horizontal lines. GPOD-MF metrics are not given in the central regions as adaptive GPOD methods only update values inside the gaps.

	Gap size	RI	MI	L2	KL
Central regions:					
UNet, MSE	10%	**1.000 ± 0.000**	0.984 ± 0.000	0.033 ± 0.001	**0.000 ± 0.000**
UNet, huber	10%	0.999 ± 0.000	0.983 ± 0.000	0.035 ± 0.001	**0.000 ± 0.000**
UNet, gradient	10%	0.999 ± 0.000	0.974 ± 0.001	0.052 ± 0.001	0.001 ± 0.000
UNETR	10%	**1.000 ± 0.000**	**0.985 ± 0.001**	**0.030 ± 0.001**	**0.000 ± 0.000**
CE-GAN	10%	0.884 ± 0.003	0.745 ± 0.010	0.470 ± 0.008	0.019 ± 0.004
Edge gaps:					
GPOD-MF	10%	0.797 ± 0.001	0.666 ± 0.001	0.610 ± 0.002	0.105 ± 0.001
UNet, MSE	10%	0.890 ± 0.004	0.759 ± 0.007	0.456 ± 0.009	**0.013 ± 0.002**
UNet, huber	10%	0.892 ± 0.003	**0.760 ± 0.003**	0.452 ± 0.005	**0.013 ± 0.001**
UNet, gradient	10%	**0.894 ± 0.002**	0.758 ± 0.004	**0.451 ± 0.004**	0.016 ± 0.001
UNETR	10%	0.884 ± 0.002	0.755 ± 0.002	0.467 ± 0.005	0.014 ± 0.001
CE-GAN	10%	0.784 ± 0.008	0.661 ± 0.010	0.622 ± 0.007	0.032 ± 0.008

Table 6.9: Results for the 180 CAD test case at 25% gaps. The mean and standard deviations are reported from the three permutations of training data defined in Table 6.3. **Bold** typeface represents the best mean in each category separated by the horizontal lines. GPOD-MF metrics are not given in the central regions as adaptive GPOD methods only update values inside the gaps.

	Gap size	RI	MI	L2	KL
Central regions:					
UNet, MSE	25%	**0.999 ± 0.000**	0.978 ± 0.001	0.044 ± 0.002	**0.000 ± 0.000**
UNet, huber	25%	**0.999 ± 0.000**	0.978 ± 0.001	0.045 ± 0.002	**0.000 ± 0.000**
UNet, gradient	25%	**0.999 ± 0.000**	0.972 ± 0.003	0.057 ± 0.005	0.001 ± 0.000
UNETR	25%	**0.999 ± 0.000**	**0.983 ± 0.001**	**0.034 ± 0.003**	**0.000 ± 0.000**
CE-GAN	25%	0.885 ± 0.006	0.739 ± 0.006	0.470 ± 0.011	0.020 ± 0.001
Edge gaps:					
GPOD-MF	25%	0.762 ± 0.008	0.629 ± 0.007	0.654 ± 0.009	0.144 ± 0.011
UNet, MSE	25%	0.817 ± 0.005	0.691 ± 0.006	0.571 ± 0.006	0.029 ± 0.001
UNet, huber	25%	0.822 ± 0.001	0.691 ± 0.004	0.565 ± 0.002	0.028 ± 0.001
UNet, gradient	25%	**0.826 ± 0.008**	**0.692 ± 0.003**	**0.559 ± 0.010**	0.040 ± 0.009
UNETR	25%	0.800 ± 0.005	0.680 ± 0.004	0.598 ± 0.004	**0.027 ± 0.001**
CE-GAN	25%	0.735 ± 0.005	0.620 ± 0.004	0.677 ± 0.005	0.055 ± 0.004

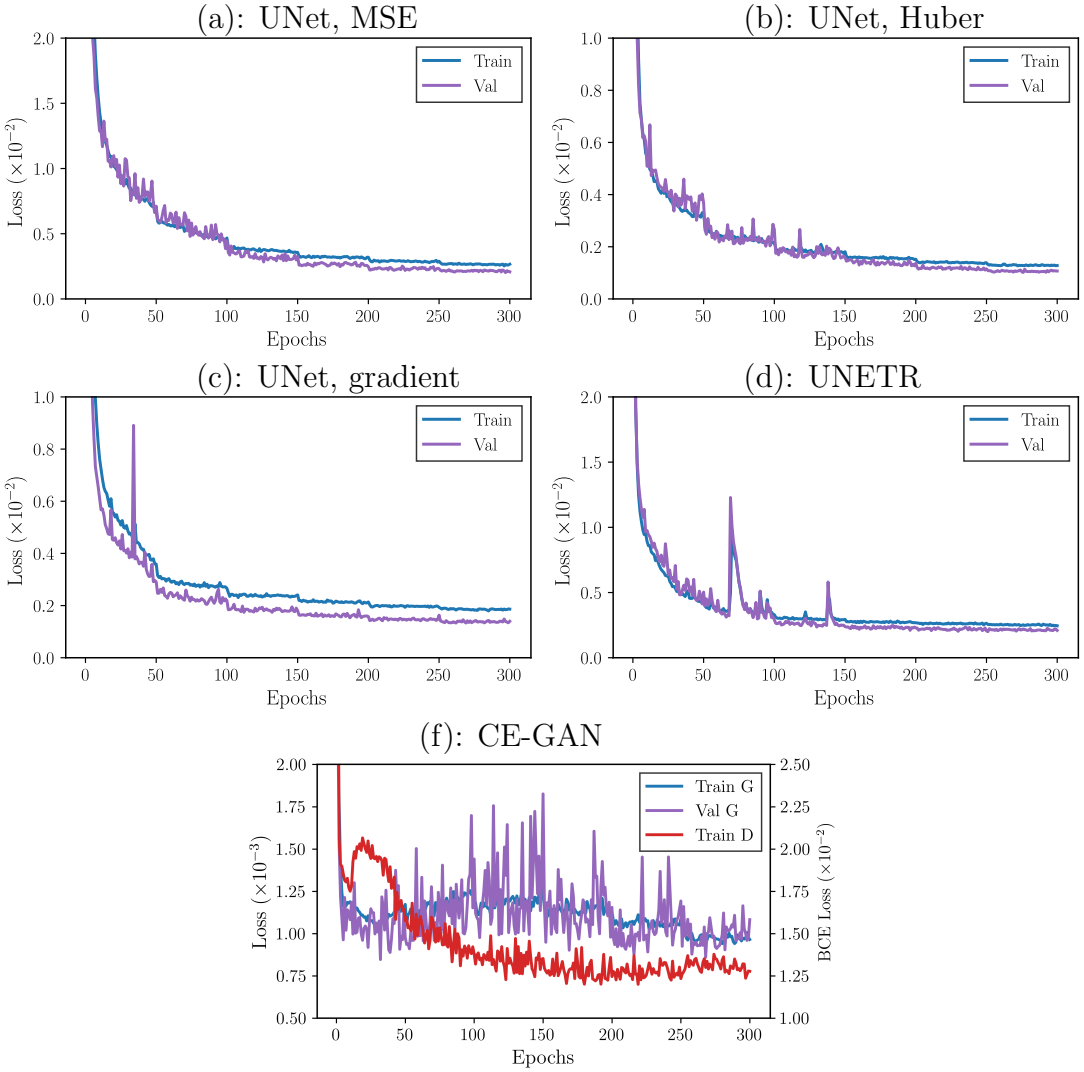

Figure 6.8: Training loss curves for each of the parametric models for permuation A at 10% gaps. In the CE-GAN plot in part **(f)**, note that the generator 'G' was trained using a combined MSE and BCE loss, while the discriminator 'D' was trained solely with the BCE loss.

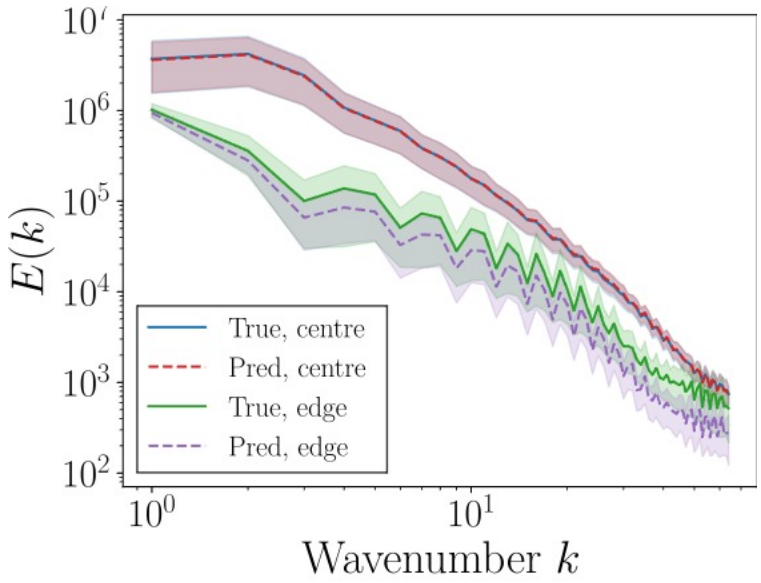

Figure 6.9: Energy spectra comparing the ground truth test set images to the UNet, MSE predictions at a 10% test gap size. Ensemble mean spectra are given by solid or dashed lines, with the shaded areas representing one standard deviation from the means.

the models seem to face something of a trade-off between accurate KL divergences, and RI and L2 errors.

The accuracy of all UNet-based models is very high in the image centres, with KL divergences that round to zero at a three decimal place tolerance, showing that the original flow structures across all scales are being well-preserved. Ensemble averaged energy spectra for the UNet, MSE model predictions at 10% gaps are shown in Figure 6.9, and there is a near line-on-line match between the true and predicted spectra in the image centres. Note that the energy spectra are challenging to compute in the gappy regions in isolation, as sharp edges and discontinuities are prevalent, contributing to the Gibbs phenomena observed in the edge spectra in Figure 6.9. However, overall trends can still be seen, and the UNet edge prediction follows a downward trend that is similar to the ground truth.

Regarding the other metrics in the edge gaps, the L2 errors of both UNet and UNETR models are relatively high at between 45-47%. This is within the range of values reported by Li *et al.* [134] for large gap sizes, but about twice as high as other results reported by Morimoto *et al.* [126] for the reconstruction of a turbulent flow in a fixed gap shape. The reasoning behind this is discussed in §6.4. For the RI and MI, values of between $0.9-0.95$ are commonly taken to represent self-similarity between vector fields [88]. The average RIs for the UNet and UNETR predictions at 10% gap sizes approach this criterion in the edge gaps, and meet it in the central regions. The MI values are systematically lower, which is consistent with other reports that the MI is a stricter metric to satisfy, as it follows a linear relationship rather than the sinusoidal RI [44, 88, 230].

Example flow field predictions from the UNet, MSE model at 10% gaps are shown in Figure 6.10. In the top row of the figure, the regions masked out by the horizontal mask are relatively uniform and easy to predict with no large variations in velocity magnitude. This allows the UNet to predict the flow inside the gaps to a fair degree of accuracy. On the other hand, for the bottom row, turbulent motion inside the edge gap regions is more complex, with the flow directions switching to point outwards just inside the edge gap regions. There are few obvious indicators for this motion in the centre of the image, and the UNet struggles to fully predict this complexity. The scarcity of spatially local information due to the edge gaps highlights the challenge presented by this inpainting task; it is likely that more knowledge of the out-of-plane motion would be needed in order to predict such complex behaviour. To provide a clearer picture of the differences between these two flow fields, the point-wise L2 errors are shown in Figure 6.11. Plots showing example outputs from each of the models at 10% and 25% gap sizes are provided in Figures 6.12 and 6.13.

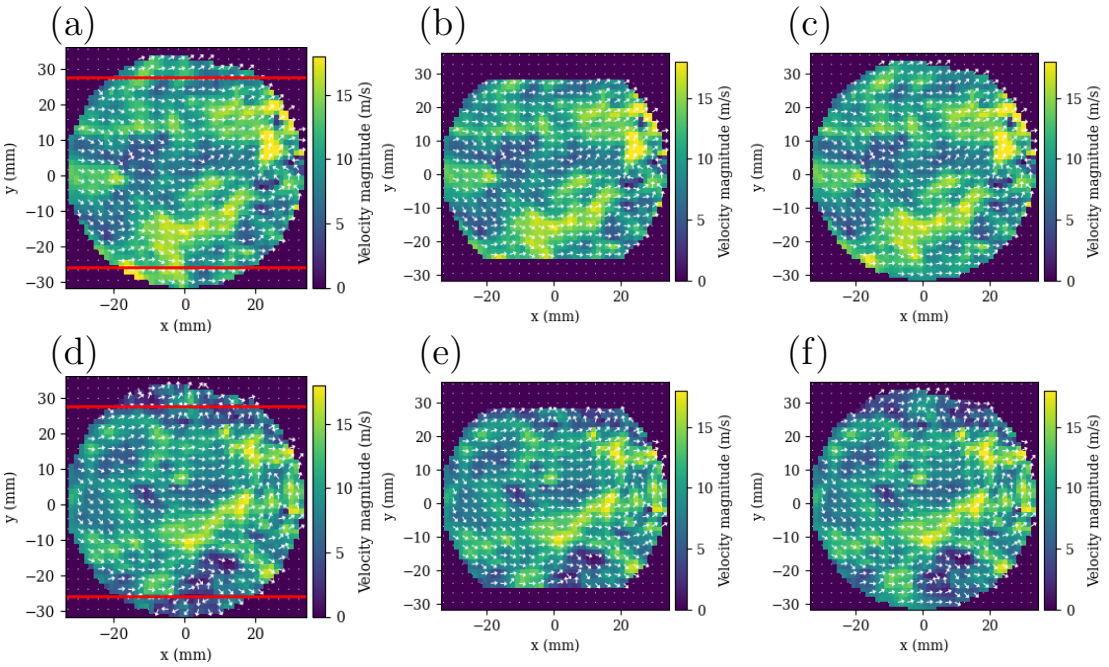

Figure 6.10: Sample flow fields from the 10% gaps test set. Top row: best UNet, MSE prediction (L2 = 0.225); bottom row: worst UNet, MSE prediction (L2 = 1.026). Part (a) and (d): original snapshot with the test mask shown as horizontal red lines; (b) and (e): gappy input; (c) and (f): prediction.

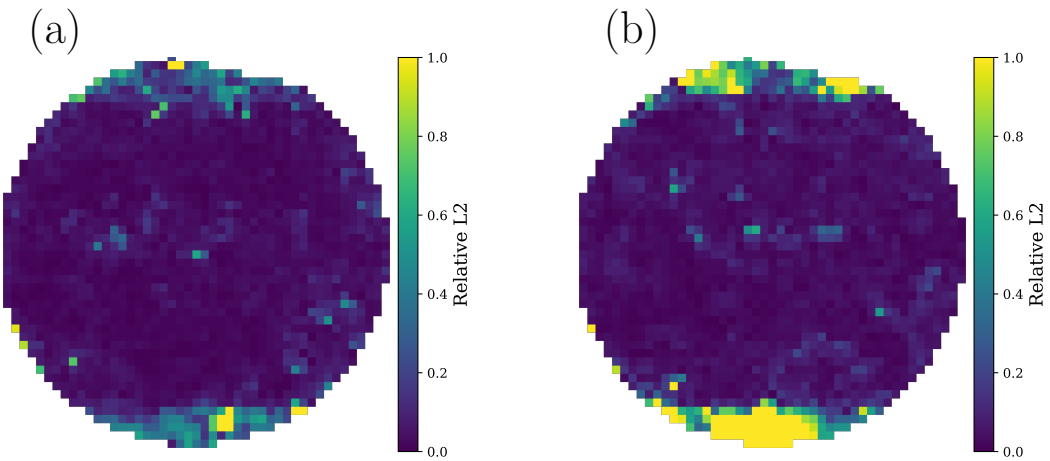

Figure 6.11: Pixel-wise L2 errors between the UNet predictions and the original images for (a): the best (top row of Figure 6.10), and (b): the worst predictions (bottom row of Figure 6.10).

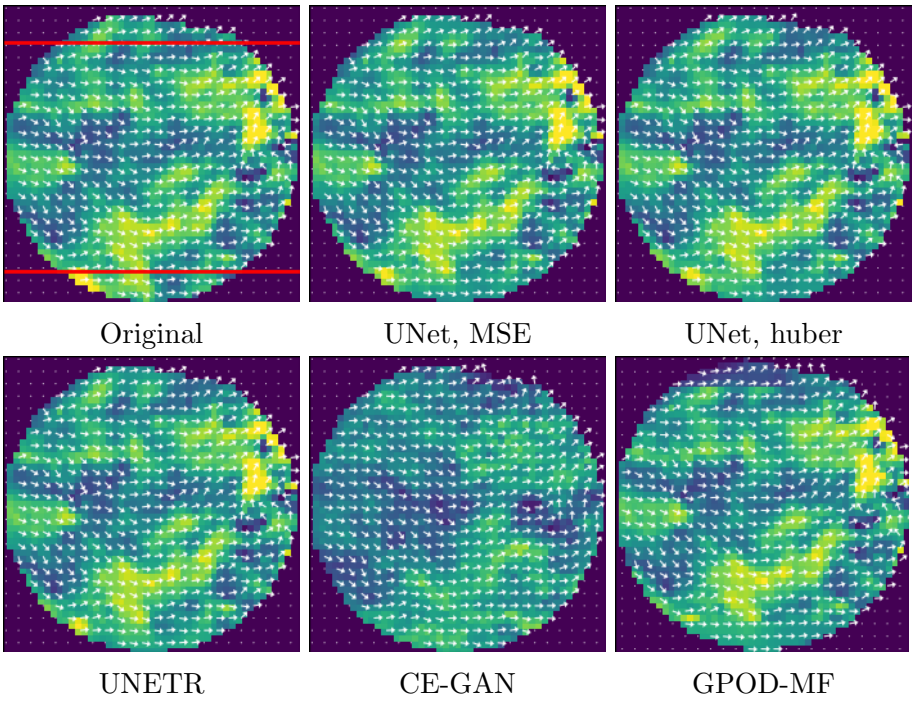

Figure 6.12: Comparison of different model predictions for a single test snapshot at 10% gaps. Gappy images formed by removing data outside of the red lines in the original image are fed into the models.

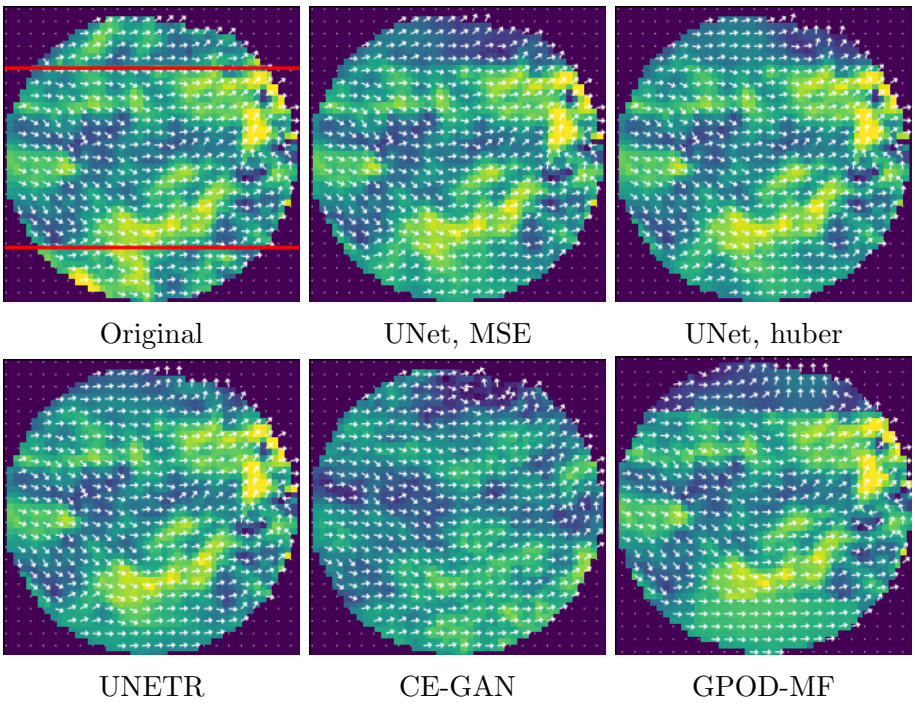

Figure 6.13: Comparison of different model predictions for a single test snapshot at 25% gaps. Gappy images formed by removing data outside of the red lines in the original image are fed into the models.

The CE-GAN demonstrates relatively poorer performance across the board. Li *et al.* [134] also reported relatively low pixel-wise accuracies for the CE-GAN in an inpainting task on PIV data, but better performance than GPOD in terms of predicting multi-scale properties. These findings are supported in the present study; however, Tables 6.8 and 6.9 show that the CE-GAN results are worse than UNet-based models across all metrics, especially in the central image regions. Finally, the GPOD-MF method yields the lowest performance, as GPOD-MF is sensitive to the limited amount of spatial information available near the gap regions. Difficulties arise because the algorithm initialises the gaps with ensemble mean vectors calculated from the training set, then iterates on these guesses using the dominant flow features from POD-based reconstructions. However, for highly variational flows, the mean can be a poor approximation of the full dataset [54, 230]. GPOD-MF can typically overcome this by utilising the local spatial information to update the guesses, but this is not so effective for large block gaps, and errors can be compounded instead. Figure 6.7 shows that the algorithm converges at a relatively small number of modes, representing the dominant flow structures. While these dominant structures do not fare as badly on the global vector-based metrics, they are overly smoothed and differ vastly in terms smaller-scale flow structures, as shown by the large KL divergences between the GPOD-MF predictions and the true vectors in the edge gaps.

6.3.5 Validation targets

Finally, the ability of the ML-reconstructed flow fields to serve as validation targets for CFD data is explored. As one of the highest-performing models, the outputs of the UNet, gradient model on the test set were compared to the true test set images. Each set of images was collapsed onto a representative validation target using both the ensemble mean (EM) and the sparsity-promoting dynamic mode decomposition (SPDMD). The SPDMD modes were constructed using a consistent threshold of 140,

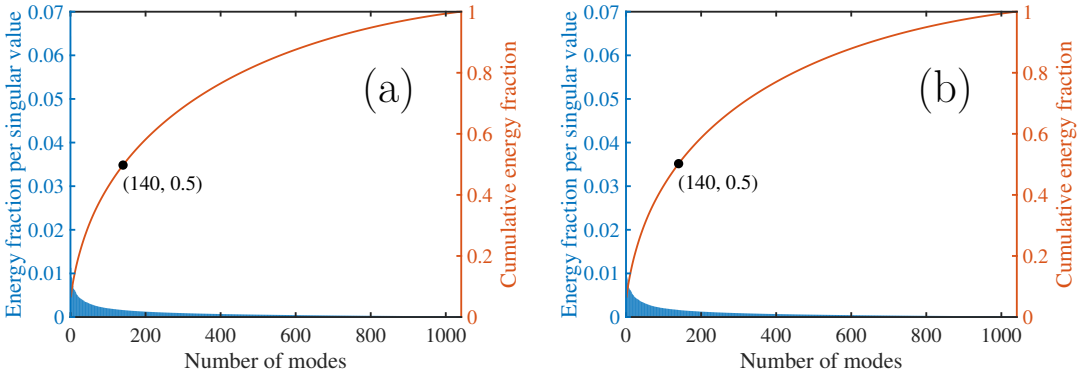

Figure 6.14: Singular value decomposition plots for the TCC-III test set data, for both **(a)**: the set of true images, and **(b)**: the set of UNet, gradient predictions. The variance captured in each singular value is given by the bar chart on the left axis, and the cumulative variance is given by the line graph on the right axis. The number of modes corresponding to 50% of the cumulative variance is marked for each dataset.

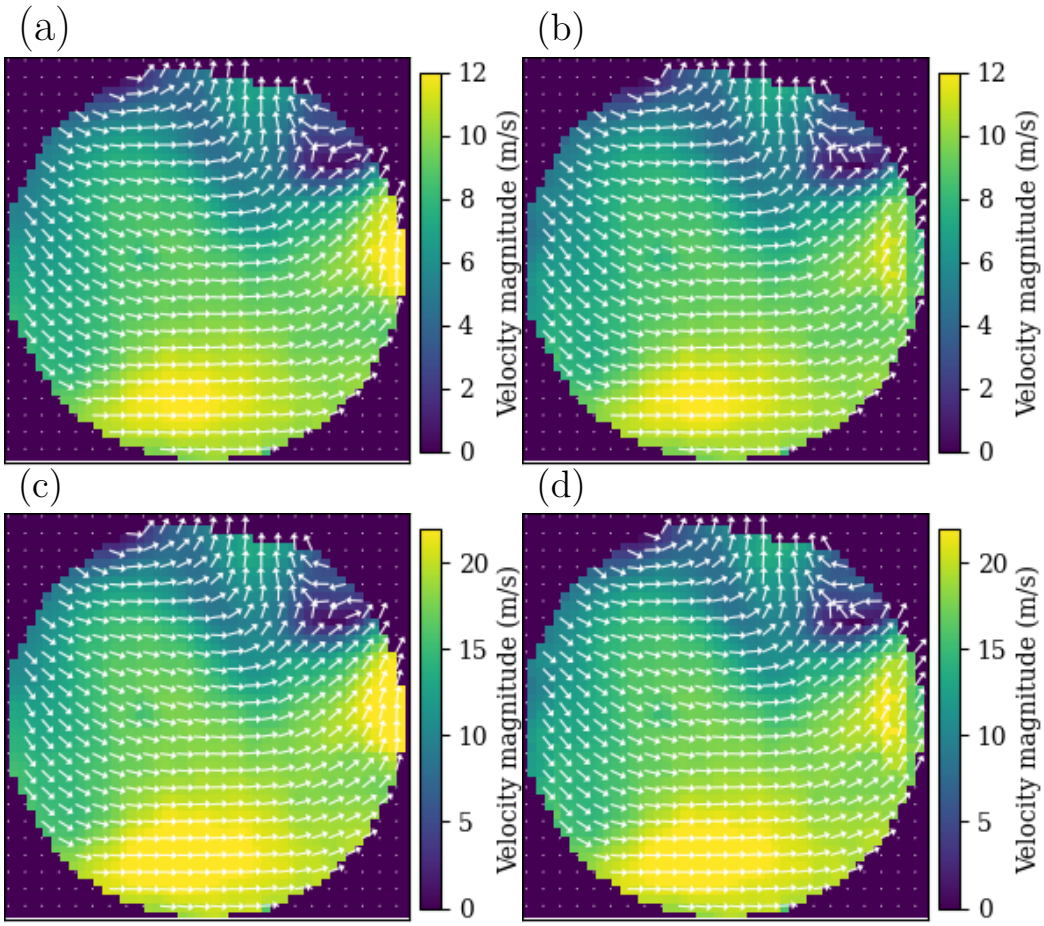

Figure 6.15: The original test images and UNet, gradient predictions as validation targets, with (a): the true EM; (b): the predicted EM; (c): the true SPDMD 0 Hz mode; (d): the predicted SPDMD 0 Hz mode.

corresponding to 50% of the cumulative variance in the data, as shown in Figure 6.14. The EM and SPDMD 0 Hz modes for each dataset with the vertical edge gaps are then provided in Figure 6.15. The predicted validation targets qualitatively appear to be close matches with the true ones, albeit with some slight under-predictions of the vector magnitudes at the left and right edges. Quantitatively, the similarity metrics between the true and predicted validation targets inside the edge gap regions were RI=0.964, MI=0.853, L2=0.278, KL=0.000. The metrics were the same for the comparison of the EM fields and the SPDMD 0 Hz modes. These figures support the observation that although general vector alignment is captured well in the predicted validation targets, the point-wise prediction of vector magnitudes is less accurate, leading to weaker MI and L2 scores.

6.4 Discussion

These results have shown that UNet-based models are capable of extrapolating beyond the field of view by reconstructing the flow inside edge gaps to a reasonable degree of accuracy, significantly out-performing GPOD. At 10% gap size, all three UNets achieved an RI of at least 0.89 on average for the unseen crank angle, showing that vector alignments can be well-predicted in general. However, the 25% gap size presents a much harder challenge, with RIs falling to approximately 0.82. The raw metrics at 10% and 25% gap sizes might not seem too disparate at first, but a visual inspection of Figures 6.12 and 6.13 reveals that the differences between these scores has significant consequences in the predicted flow fields. While the predicted flow fields at the 10% gap size appear to be reasonable in general, the predictions at the 25% gap size are unreliable, with large inaccuracies in the predicted flow motion.

This poses an interesting question as to what level of accuracy should be expected from

an inpainting model, and whether it is possible to accurately predict the flow inside edge gaps as large as 25% of the total pixels. The process of inpainting in this case relies on there being a strong correlation between the flow at the centre of image and the flow inside edge gaps. For the larger gap size, it is less likely for such a correlation to exist, as the distance between the outer gap regions and the nearest data-containing pixel is increased. If there is no strong correlation between these different regions of the flow, then this becomes an ill-posed problem, with many possible options for the flow behaviour inside the edge gaps [239]. Future work should investigate how the internal correlations within the flow relate to an inpainting model's performance, to further inform what is and is not possible within this task.

Comparisons between the neural network models here show that UNet-based models exhibit similar performances, while the CE-GAN accuracies are markedly worse. In particular, the low accuracies inside the central regions indicate that the CE-GAN is not retaining as much of the information in the image centres as the UNet-based models are. Indeed, although both models incorporate autoencoder-like structures, while the CE-GAN generator has an AlexNet-like architecture [144], UNet-based models utilise skip connections that are designed to preserve contextual information at each stage of the autoencoder. This allows the UNets to simultaneously preserve information in the image centre to a very high degree of accuracy, and yield gap predictions that seamlessly integrate with the rest of the image. This has also correlated with better edge gap predictions for the UNet-based models in this case. It should be noted that this particular application pushes the CE-GAN beyond its initial design intention of solely predicting inside the gap region, rather than also reproducing the entire image. In summary, the key structural difference between the CE-GAN generator and the UNets is the inclusion of skip connections, and the CE-GAN reports lower accuracies in the central image regions, which is information that skip connections are

designed to preserve. This points towards skip-connections being key architectural components allowing ML models to reach higher performances in this inpainting task, which should be considered in the future development of ML models for turbulent flows.

A note is needed regarding the relatively high L2 errors, of between 45-47% for the UNet-based models. It is hypothesised that the main reason for the higher L2 errors reported in this benchmark is the significant out-of-plane motion present in the TCC-III engine, which is not accounted for currently. As shown in the bottom row of Figure 6.10, the flow at the image centres reveal few indicators of the sudden change in vector directions inside the edge gaps. However, only two velocity components along a single PIV plane are observed here. It is possible that accounting for out-of-plane motion by gaining access to the third velocity component using techniques such as tomographic PIV [248] or assimilation with CFD data [249] will be required to significantly reduce the L2 errors in this situation, and such investigations will constitute future work.

With the present results as they are, it is recommended that UNet-based models can be used to reconstruct the flow in large block gaps with as many as 10% of the total number of pixels missing to a reasonable degree of accuracy. Such reconstructed flow fields could be used to improve understanding of general flow patterns, and replace ensemble mean-filled or interpolated flow fields as inputs into other data analysis methods like modal decomposition. In addition, this level of performance facilitates the construction of reasonably accurate validation targets from the predicted flow fields, with RI=0.96 between the predicted and ground-truth validation targets. However, despite the well-predicted vector alignments, care should be taken when using the vector magnitudes from the predicted validation targets, as these were

found to under-predict the ground-truth values. Note that the performance of the UNets is expected to improve for easier inpainting tasks such as smaller and more centrally-located blocks of missing data, in which case the prediction of these vector magnitudes would likely improve. Testing the sensitivity of inpainting models to a range of gap types is also recommended for future work. Finally, reliably inpainting edge gaps for the 25% gap size appears to be out of reach at the moment, and as previously discussed, an investigation into how feasible it would be for any model to accurately reconstruct the flow in such scenarios is recommended future work.

6.5 Conclusion

This chapter has introduced the EngineBench database, and used it to establish the first inpainting benchmark for an industrially-relevant turbulent flow, in order to address the limited availability of practical benchmarks on experimental data. The models were tasked with inpainting large edge gaps, a highly challenging problem that pushes the models to the limits of their capabilities. This benchmark was used to provide objective insight into how a range of widely-used models behave in these challenging conditions, identify success and failure modes, and provide recommendations for future work.

Firstly, a number of data augmentation strategies that introduce artifical gaps of different forms into the data are tested to find the optimum strategy for inpainting edge gaps. A novel strategy, named random edge gaps, was created to introduce edge gaps of random sizes, locations, and orientations into the training data. Although fixed gap training yielded the best results in terms of MI and KL divergence, random edge gap training was shown to result in the most accurate predictions of vector orientations and pixel-wise errors due to the improved generalisability. Random edge

gap training is therefore recommended for the creation of flexible and generalisable inpainting models in this case.

Overall, UNet-based models demonstrated the best general performance across the four metrics and two gap sizes tested. Indeed, the UNet-based model predictions in the edge gaps approached self-similarity at the 10% gap size according to the vector-based metrics. This suggests that even gaps as challenging as large edge gaps can be reconstructed to a reasonable degree of accuracy with the use of a UNet, which exhibits significant performance improvements over the GPOD method for this type of gap. However, pixel-wise L2 errors remained relatively high for all model predictions. A visual inspection of the reconstructed flow fields revealed that sudden changes in the flow direction without any obvious indicators in the rest of the flow was a cause of lower reconstruction accuracies. It is therefore hypothesised that acquiring information on the out-of-plane motion, such as through stereo-PIV or data assimilation with CFD, would be needed in order to rectify this issue.

A comparison between the UNet-based models and the CE-GAN showed that the former were capable of preserving the central flow information provided at the input to a very high degree of accuracy, and then leveraging this information to produce better predictions inside the edge gaps. This indicates that skip-connections, contained within UNet-based models but not the CE-GAN generator, are important architectural components that facilitate high accuracies for neural networks training to reconstruct turbulent flow data. This characteristic should be considered in future model development.

With regards to how the present reconstructed flow fields can be used, the strong RI and KL divergences of the UNet reconstructions at 10% edge gaps indicate that these

results are reliable enough to provide insight into general flow patterns, and contribute to other data analysis approaches such as by replacing ensemble mean-filled flow fields in modal decomposition. In addition, the predicted flow fields could be used to construct validation targets from the data, with high vector alignment similarity to the ground-truths, although the vector magnitudes are less well preserved, indicating the need for caution. For the 25% gap size, the inaccuracies are very high and predictions unreliable. Recommended future work also consists of investigating methods of incorporating information on the out-of-plane motion into the inpainting task, exploring how the spatial correlations within the flow impact expected inpainting performance, and testing inpainting models on a variety of different gap types including random noise and other forms of block gaps.

Chapter 7

Conclusion

7.1 Key findings

The aim of this thesis has been to improve methods of validating engine CFD data against PIV vector fields through the use of dimensionality reduction techniques. The specific challenges of validation target creation and validation data enhancement are addressed. Within each of these challenges, focus has been placed upon accurately collapsing ensembles of vector data, and rectifying large gaps in experimental data. The Oxford PIV dataset was firstly investigated within the context of RANS validation, due to the industrial relevance of the datasets and the previously-identified intake jet flapping phenomenon that contributes to vector magnintude diminishing [54]. The Modena/Darmstadt LES/PIV pair of datasets was then explored, as LES validation involves the comparison of two different ensembles of data. Therefore, this scenario facilitated a deeper analysis of the proposed SPDMD method, with explorations into the effects of comparing SPDMD modes rather than EM flow fields, diagnosing in-

dividual LES cycles, and the dataset sizes. Finally, for the enhancement of validation data, a benchmark was established using open-source TCC-III PIV data and a new edge-gap inpainting task, in order to fairly compare different inpainting methods, identify success and failure modes, and propose pathways for future work. The nascency of inpainting in fluid mechanics research as well as the lack of reproducibility in previously-published works motivated the establishment of such a benchmark, which was named EngineBench.

Firstly, an investigation was conducted into the diminished magnitudes effect. A number of issues were also identified with previously-proposed remedies to this phenomenon, with key contributions and findings outlined below.

- The diminished magnitudes effect that results from taking an ensemble average of vector data was robustly quantified for the first time. The histogram distance (HD) technique [227] was implemented for this task. In addition, the degree of vector magnitude diminishment was linked to the level of variability within the flow in question, by calculating the RI between the individual images and the EM flow fields, as well as the standard deviations of image-wise velocity magnitudes.

- A framework was introduced for the qualitative comparison of different validation target creation methods. It was used in order to class the EM and DMD methods as having many-to-one mappings between the input data matrix and the reconstructed validation target, while the POD was classed as many-to-many. This difference between the POD and DMD methods arises because the former exhibits temporal dependancy, causing POD-based reconstructions to differ depending on which of the right-singular vectors is being used in the re-

construction. This property is categorised as a drawback for the POD method in this context, as many-to-one mappings reduce the data complexity and produce more parsimonious representations of the dataset.

- Although the exact DMD algorithm was able to produce a many-to-one mapping by considering flow field reconstructions that operate at a single frequency, the method was found to not be robust across different subsets of the PIV data. This is because the naive definition of the DMD mode amplitudes in the exact algorithm leaves the amplitudes highly sensitive to random noise and outlier turbulent flow behaviour.

As a result of this analysis, the SPDMD algorithm was proposed due to being a temporal decomposition technique that also re-scales the mode amplitudes such that reconstrustruction errors to the original image data can be minimised. The following improvements to vector data analysis and validation target creation were therefore found as a result.

- Significantly improved HD scores with the ground-truth histograms were produced by the SPDMD 0 Hz modes relative to the other methods tested. In addition, the results were found to be consistent across the different engines, PIV planes, and analysis crank angles considered. This led to the conclusion that the SPDMD method is more effective at creating validation targets from vector data than the EM, POD, and exact DMD.

- The impact of using SPDMD-generated validation targets over the EM was also investigated, providing further insight into the data and the techniques. A comparison of two SPDMD 0 Hz fields was found to more accurately quantify

the speed differences between the LES and PIV data than the use of two EM fields. In addition, use of the SPDMD 0 Hz mode facilitated more accurate analyses of the similarity of individual LES snapshots to the ensemble of PIV data.

- The size of the datasets was also found to be an important factor in determining how the variance in the dataset is distributed among the singular values. The PIV data exhibited more similar characteristics to the LES data in this respect when the same number of snapshots were considered for each, explaining some of the reason for differences in variability between the two datasets. However, this investigation also revealed that there was no strong pattern between the optimal SVD threshold (in a HD sense) and the cumulative variance captured by the modes. Therefore, an objective definition of mode thresholding remains an open question, and is recommended for future work.

Finally, with the proposal of the SPDMD technique for validation target creation, consideration was given to the reconstruction of data inside gaps present in a validation dataset. A new inpainting benchmark was created for engine flows, with the target being the challenging task of inpainting large block gaps at the edges of the field of view in the PIV images. Six different inpainting model configurations were implemented for this task, with performance being evaluated across four different metrics for two gap sizes, consisting of data missing in 10% and 25% of the total number of pixels. A number of contributions were made as a result.

- A fully reproducible, open-source benchmark was created, named EngineBench. This allows researchers to build upon the initial results presented here by testing different modelling approaches and data analysis tasks. In addition, a series of

quick-start tutorials were created to accompany the benchmark, in order to minimise the barrier to entry for engine researchers.

- By using EngineBench, it was demonstrated that it is possible to reconstruct the flow inside edge gap regions at the 10% size to a reasonable degree of accuracy (RI=0.89) for the unseen crank angle used for testing. However, gaps as large as 25% remain out of reach at present. By benchmarking the model performances on a challenging edge gaps task, it is intended that new upper limits can be set for what is possible with ML-enhanced vector data processing.

- The UNet-based models exhibited the strongest performances, reporting RI=0.89 inside the edges, while GPOD scored below 0.80. The GPOD method typically produces strong performances with more localised gaps of smaller sizes, highlighting the increased challenge presented by the edge gaps. The key architectural reason behind the strong performances of the UNets was accredited to the skip-connections present in these neural networks, which allow for more accurate preservation of contextual information, leading to improved learning of the relationship between the image centres and edges.

- Although the pixel-wise L2 errors between the UNet predictions and the ground-truth images remained relatively high inside the edges, it was found that the UNet predictions at 10% edge gaps could be reliably used as validation targets for CFD data, as long as the analysis is constrained to the vector orientations: RI=0.96 inside the edges between SPDMD 0 Hz modes constructed using the predicted and ground-truth data. However, pixel-wise L2 errors were higher at 28%, caused by a general under-prediction of vector magnitudes by the UNet models. This would need to be improved upon before full validation could be

conducted using UNet-reconstructed flows inside edge gaps.

- A new data augmentation method was also proposed for use with edge gap inpainting, named random edge gaps. This technique, involving the removal of data outside of randomly-drawn polygons, led to a 4% improvement in the L2 error inside the edges over directly training the models on the test mask, due to the improved generalisability of the models.

7.2 Recommendations and future work

Based on the findings of this thesis, the following recommendations are made as current best-practice.

- SPDMD 0 Hz modes should be used instead of the EM or POD for the creation of validation targets from ensembles of vector data, particularly when the data exhibit significant variability. In this case, the SPDMD technique leads to improved representations of the vector magnitudes in the data, leading to more accurate validation processes and diagnoses.

- ML models, particularly UNets, can reasonably reconstruct the turbulent flow behaviour inside engine PIV images with edge gaps as large 10% of the total number of pixels. The accuracy is high enough that these reconstructed flow fields can be used as validation targets (via the SPDMD or EM methods) when considering the vector orientations. However, the vector magnitudes are currently less reliably predicted.

There is significant scope for future work based on these findings. Firstly, the is-

sue of objectively defining mode thresholds for SVD-based methods remains an open question, and impacts the use of the SPDMD method. The key to the success of the SPDMD approach was the underlying ADMM optimisation algorithm, which handled the vector magnitude re-scaling. It is therefore possible that the subjectivity in choosing the number of DMD modes to be included in the analysis could be bypassed by using a more direct implementation of the ADMM algorithm on vector field data. The use of optimisation algorithms such as the ADMM with vector data ensembles should therefore be further explored.

Regarding the outcomes of the inpainting benchmark, future work should explore methods of improving the reconstruction accuracy inside the edge gaps. In this case, it is suggested that significant performance gains could be made with the inclusion of out-of-plane information, such gaining the third velocity component by assimilation with 3D CFD data. Future work should also investigate the practicality of attempting to inpaint edge gaps as large as 25% of the data by considering the degree of correlation within the flow on the PIV plane in question. In addition, a systematic effort to quantify and characterise the different types of gaps that exist in PIV data would aid researchers in deciding how to add artificial gaps to clean data in order to test new inpainting methods. Edge gaps is perhaps one of the more challenging inpainting tasks, with performance improvements expected for the inpainting of smaller and more centrally-located gaps. Furthermore, the EngineBench results indicated that skip connections are an important feature of neural network architectures for accurately inpainting large edge gaps, and it would be valuable to investigate this further during the development of improved inpainting models.

References

[1] H. Ritchie and M. Roser, "CO_2 and Greenhouse Gas Emissions," 2017, Accessed: 2023-10-30. [Online]. Available: https://ourworldindata.org/co2-and-other-greenhouse-gas-emissions

[2] Statista, "Estimated worldwide automobile production from 2000 to 2019," 2020, Accessed: 2023-10-30. [Online]. Available: https://www.statista.com/statistics/262747/worldwide-automobile-production-since-2000/

[3] D. Banister, "Transport for all," *Transport Reviews*, vol. 39, no. 3, pp. 289–292, 2019.

[4] D. A. King, M. J. Smart, and M. Manville, "The Poverty of the Carless: Toward Universal Auto Access," *Journal of Planning Education and Research*, 2019.

[5] F. Leach, G. Kalghatgi, R. Stone, and P. Miles, "The scope for improving the efficiency and environmental impact of internal combustion engines," *Transportation Engineering*, vol. 1, p. 100005, 2020.

[6] E. Emilsson and L. Dahllöf, "Lithium-ion vehicle battery production: Status 2019 on energy use, co_2 emissions, use of metals, product environmental footprint, and recycling," IVL Swedish Environmental Research Institute, Tech. Rep. C 444, November 2019. [Online]. Available: https://www.ivl.se/download/18.34244ba71728fcb3f3fda0/1574923989017/C444.pdf

[7] M. Chordia, A. Nordelöf, and L. A.-W. Ellingsen, "Environmental life cycle implications of upscaling lithium-ion battery production," *The International Journal of Life Cycle Assessment*, vol. 26, pp. 2024–2039, 2021.

[8] Q. Qiao, F. Zhao, Z. Liu, S. Jiang, and H. Hao, "Cradle-to-gate greenhouse gas emissions of battery electric and internal combustion engine vehicles in china," *Applied Energy*, vol. 204, pp. 1399–1411, 2017.

[9] A. Elgowainy, J. Han, J. Ward, F. Joseck, D. Gohlke, A. Lindauer, T. Ramsden, M. Biddy, M. Alexander, S. Barnhart *et al.*, "Current and future United States light-duty vehicle pathways: cradle-to-grave lifecycle greenhouse gas emissions and economic assessment," *Environmental Science & Technology*, vol. 52, no. 4, pp. 2392–2399, 2018.

[10] A. Joshi, "Review of vehicle engine efficiency and emissions," *SAE International Journal of Advances and Current Practices in Mobility*, vol. 4, no. 2022-01-0540, pp. 1704–1733, 2022.

[11] P. Senecal and F. Leach, "Diversity in transportation: Why a mix of propulsion technologies is the way forward for the future fleet," *Results in Engineering*, vol. 4, p. 100060, 2019.

[12] U. E. P. Agency, "The 2019 EPA Automotive Trends Report," 2020, Accessed: 2023-10-30. [Online]. Available: https://www.epa.gov/automotive-trends/highlights-automotive-trends-report

[13] R. D. Reitz, H. Ogawa, R. Payri, T. Fansler, S. Kokjohn, Y. Moriyoshi, A. Agarwal, D. Arcoumanis, D. Assanis, C. Bae *et al.*, "IJER editorial: The future of the internal combustion engine," *International Journal of Engine Research*, vol. 21, no. 1, pp. 3–10, 2020.

[14] M. H. Morsy, "Review and recent developments of laser ignition for internal combustion engines applications," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 7, pp. 4849–4875, 2012.

[15] R. Stone, *Introduction to Internal Combustion Engines*, 4th ed. SAE International and Macmillan Press, 2012, ISBN: 978-0-7680-2084-7.

[16] D. Dunn-Rankin, *Fundamentals of lean combustion.* Elsevier Inc., 2008, ISBN: 9780128005774.

[17] M. Raza, L. Chen, F. Leach, and S. Ding, "A review of particulate number (PN) emissions from gasoline direct injection (GDI) engines and their control techniques," *Energies*, vol. 11, no. 6, p. 1417, 2018.

[18] J. L. Lumley, "Coherent structures in turbulence," in *Transition and turbulence.* Elsevier, 1981, pp. 215–242.

[19] ——, *Engines: an introduction.* Cambridge University Press, 1999.

[20] T. Poinsot and D. Veynante, *Theoretical and Numerical Combustion*, 2nd ed. R.T. Edwards, 2005, ISBN: 1-930217-10-2.

[21] C. K. Law, *Combustion physics.* Cambridge university press, 2010.

[22] F. Leach, R. Stone, D. Richardson, A. Lewis, S. Akehurst, J. Turner, S. Remmert, S. Campbell, and R. F. Cracknell, "Particulate emissions from a highly boosted gasoline direct injection engine," *International Journal of Engine Research*, vol. 19, no. 3, pp. 347–359, 2018.

[23] J. Turner, A. Popplewell, R. Patel, T. Johnson, N. Darnton, S. Richardson, S. Bredda, R. Tudor, C. Bithell, R. Jackson *et al.*, "Ultra boost for economy: extending the limits of extreme engine downsizing," *SAE International Journal of Engines*, vol. 7, no. 1, pp. 387–417, 2014.

[24] J. Zhao, "Research and application of over-expansion cycle (Atkinson and Miller) engines – A review," *Applied Energy*, vol. 185, pp. 300–319, jan 2017.

[25] S. Tavakoli, S. A. Jazayeri, M. Fathi, and O. Jahanian, "Miller cycle application to improve lean burn gas engine performance," *Energy*, vol. 109, pp. 190–200, 2016.

[26] B. Ribeiro and J. Martins, "Direct comparison of an engine working under Otto, Miller and Diesel cycles: Thermodynamic analysis and real engine performance," *SAE Technical Paper 2007-01-0261*, 2007.

[27] G. Fontana, E. Galloni, and E. Torela, "The influence of variable valve timing on the combustion process of a small spark-ignition engine," *SAE Technical Paper 2006-01-0445*, 2006.

[28] M. E. Martins and T. D. Lanzanova, "Full-load Miller cycle with ethanol and EGR: Potential benefits and challenges," *Applied Thermal Engineering*, vol. 90, pp. 274–285, jul 2015.

[29] G. Kalghatgi and B. Johansson, "Gasoline compression ignition approach to efficient, clean and affordable future engines," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, vol. 232, no. 1, pp. 118–138, 2018.

[30] Mazda UK, "Mazda Skyactiv-X Engine," 2019, Accessed: 2021-02-03. [Online]. Available: https://www.mazda.co.uk/why-mazda/skyactiv-x/

[31] Z. Hu, J. Zhang, M. Sjöberg, and W. Zeng, "The use of partial fuel stratification to enable stable ultra-lean deflagration-based Spark-Ignition engine operation with controlled end-gas autoignition of gasoline and E85," *International Journal of Engine Research*, vol. 21, no. 9, pp. 1678–1695, 2020.

[32] C. E. C. Alvarez, G. E. Couto, V. R. Roso, A. B. Thiriet, and R. M. Valle, "A review of prechamber ignition systems as lean combustion technology for SI engines," *Applied Thermal Engineering*, vol. 128, pp. 107–120, 2018.

[33] M. Bunce, A. Cairns, and H. Blaxill, "The use of active jet ignition to overcome traditional challenges of pre-chamber combustors under low load conditions," *International Journal of Engine Research*, pp. 1–15, 2020.

[34] S. Brannys, S. Gehrke, H. Hoffmeyer, L. Hentschel, K. Blumenröder, C. Helbing, and F. Dinkelacker, "Maximum Efficiency Concept of a 1.5l TSI evo for Future Hybrid Powertrains," in *the 28th Aachen Colloquium Automobile and Engine Technology*, 2019.

[35] D. B. Gosala, C. M. Allen, A. K. Ramesh, G. M. Shaver, J. Mccarthy, D. Stretch, E. Koeberlein, and L. Farrell, "Cylinder deactivation during dynamic diesel engine operation," *International Journal of Engine Research*, vol. 18, no. 10, pp. 991–1004, 2017.

[36] E. Ortiz-Soto, R. Wang, M. Nagashima, M. Younkins, A. Müller, S. Tews, A. Balazs, and M. Thewes, "λDSF: Dynamic skip fire with homogeneous lean burn for improved fuel consumption, emissions and drivability," *SAE Technical Paper 2018-01-0891*, 2018.

[37] J. Ritzmann, N. Zsiga, C. Peterhans, and C. Onder, "A control strategy for cylinder deactivation," *Control Engineering Practice*, vol. 103, 2020.

[38] C. D. Argyropoulos and N. Markatos, "Recent advances on the numerical modelling of turbulent flows," *Applied Mathematical Modelling*, vol. 39, no. 2, pp. 693–732, 2015.

[39] F. D. Witherden and A. Jameson, "Future directions in computational fluid dynamics," in *23rd AIAA Computational Fluid Dynamics Conference*, 2017, p. 3791.

[40] S. B. Pope, *Turbulent flows*. Cambridge University Press, 2000, ISBN: 9780521591256.

[41] J. Blazek, *Computational fluid dynamics: principles and applications*. Butterworth-Heinemann, 2015, ISBN: 9780080430096.

[42] M. Schmitt, C. E. Frouzakis, Y. M. Wright, A. G. Tomboulides, and K. Boulouchos, "Direct numerical simulation of the compression stroke under engine-relevant conditions: Evolution of the velocity and thermal boundary layers," *International Journal of Heat and Mass Transfer*, vol. 91, pp. 948–960, 2015.

[43] M. Schmitt, C. E. Frouzakis, Y. M. Wright, A. Tomboulides, and K. Boulouchos, "Direct numerical simulation of the compression stroke under engine relevant conditions: Local wall heat flux distribution," *International Journal of Heat and Mass Transfer*, vol. 92, pp. 718–731, 2016.

[44] A. Barbato, C. Iacovano, and S. Fontanesi, "Cold-flow investigation of the Darmstadt engine with focus on statistical convergence: Experimental and large eddy simulation analysis," *Flow, Turbulence and Combustion*, vol. 110, no. 1, pp. 59–89, 2023.

[45] C. Iacovano, A. d'Adamo, S. Fontanesi, G. Di Ilio, and V. K. Krastev, "Application of a zonal hybrid URANS/LES turbulence model to high and low-resolution grids for engine simulation," *International Journal of Engine Research*, vol. 22, no. 8, pp. 2745–2764, 2021.

[46] H. K. Versteeg, *An introduction to computational fluid dynamics: the finite volume method*. Pearson, 2007, ISBN: 9780131274983.

[47] J. D. Anderson and J. Wendt, *Computational fluid dynamics*. Springer, 1995, ISBN: 9780070016859.

[48] R. Hamming, *Numerical methods for scientists and engineers.* Dover Publications, Inc., 1973, ISBN: 9780070258877.

[49] American Institute of Aeronautics and Astronautics, *AIAA Guide for the Verification and Validation of Computational Fluid Dynamics Simulations.* American Institute of Aeronautics and Astronautics, 1998, ISBN: 978-1-56347-285-5.

[50] W. L. Oberkampf and T. G. Trucano, "Verification and validation in computational fluid dynamics," *Progress in Aerospace Sciences*, vol. 38, no. 3, pp. 209–272, 2002.

[51] P. J. Roache, *Verification and validation in computational science and engineering.* Hermosa Pub., 1998, ISBN: 0913478083.

[52] F. Leach, R. Ismail, M. Davy, A. Weall, and B. Cooper, "The effect of a stepped lip piston design on performance and emissions from a high-speed diesel engine," *Applied Energy*, vol. 215, pp. 679–689, 2018.

[53] F. Leach, R. Ismail, and M. Davy, "Engine-out emissions from a modern high speed diesel engine–the importance of nozzle tip protrusion," *Applied Energy*, vol. 226, pp. 340–352, 2018.

[54] L. Shen, C. Willman, R. Stone, T. Lockyer, R. Magnanon, and G. Virelli, "On the use of particle image velocimetry (PIV) data for the validation of Reynolds averaged Navier-Stokes (RANS) simulations during the intake process of a spark ignition direct injection (SIDI) engine," *International Journal of Engine Research*, vol. 23, no. 6, pp. 1061–1081, 2022.

[55] R. Reitz and C. Rutland, "Development and testing of diesel engine CFD models," *Progress in Energy and Combustion Science*, vol. 21, no. 2, pp. 173–196, 1995.

[56] D. C. Haworth, "A review of turbulent combustion modeling for multidimensional in-cylinder CFD," *SAE Transactions*, vol. 114, pp. 899–928, 2005.

[57] X. Yang, S. Gupta, T.-W. Kuo, and V. Gopalakrishnan, "RANS and large eddy simulation of internal combustion engine flows—a comparative study," *Journal of Engineering for Gas Turbines and Power*, vol. 136, no. 5, 2014.

[58] C. Pera and C. Angelberger, "Large eddy simulation of a motored single-cylinder engine using system simulation to define boundary conditions: methodology and validation," *SAE International Journal of Engines*, vol. 4, no. 1, pp. 948–963, 2011.

[59] L. Shen, K.-Y. Teh, P. Ge, F. Zhao, and D. L. Hung, "Temporal evolution analysis of in-cylinder flow by means of proper orthogonal decomposition," *International Journal of Engine Research*, vol. 22, no. 5, pp. 1714–1730, 2021.

[60] S. Buhl, F. Hartmann, S. A. Kaiser, and C. Hasse, "Investigation of an IC engine intake flow based on highly resolved LES and PIV," *Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles*, vol. 72, no. 3, p. 15, 2017.

[61] K. Liu and D. C. Haworth, "Development and assessment of POD for analysis of turbulent flow in piston engines," *SAE Technical Paper 2011-01-0830*, 2011.

[62] B. Hu, S. Banerjee, K. Liu, D. Rajamohan, J. Deur, Q. Xue, S. Som, P. K. Senecal, and E. Pomraning, "Large eddy simulation of a turbulent non-reacting spray jet," in *Proceedings of the ASME 2015 Internal Combustion Engine Division Fall Technical Conference*. ASME, 2015.

[63] H. Chen, D. L. Reuss, and V. Sick, "On the use and interpretation of proper orthogonal decomposition of in-cylinder engine flows," *Measurement Science and Technology*, vol. 23, no. 8, p. 085302, 2012.

[64] H. Chen, D. L. Reuss, D. L. Hung, and V. Sick, "A practical guide for using proper orthogonal decomposition in engine research," *International Journal of Engine Research*, vol. 14, no. 4, pp. 307–319, 2013.

[65] S. Roudnitzky, P. Druault, and P. Guibert, "Proper orthogonal decomposition of in-cylinder engine flow into mean component, coherent structures and random gaussian fluctuations," *Journal of Turbulence*, no. 7, 2006.

[66] M. Dias Ribeiro, A. Mendonça Bimbato, M. Araújo Zanardi, J. A. Perrella Balestieri, and D. P. Schmidt, "Large-eddy simulation of the flow in a direct injection spark ignition engine using an open-source framework," *International Journal of Engine Research*, vol. 22, no. 4, pp. 1064–1085, 2021.

[67] M. I. Jordan and T. M. Mitchell, "Machine learning: Trends, perspectives, and prospects," *Science*, vol. 349, no. 6245, pp. 255–260, 2015.

[68] K. M. Tolle, D. S. W. Tansley, and A. J. G. Hey, "The fourth paradigm: Data-intensive scientific discovery," *Proceedings of the IEEE*, vol. 99, no. 8, pp. 1334–1337, 2011.

[69] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019, ISBN: 9781009098489.

[70] I. El Naqa and M. J. Murphy, *Machine Learning in Radiation Oncology*. Springer, 2015, ISBN: 978-3-319-18305-3.

[71] R. Vinuesa, S. L. Brunton, and B. J. McKeon, "The transformative potential of machine learning for experiments in fluid mechanics," *Nature Reviews Physics*, vol. 5, no. 9, pp. 536–545, 2023.

[72] R. Vinuesa and S. L. Brunton, "Enhancing computational fluid dynamics with machine learning," *Nature Computational Science*, vol. 2, no. 6, pp. 358–366, 2022.

[73] F. Zhao and D. L. Hung, "Applications of machine learning to the analysis of engine in-cylinder flow and thermal process: A review and outlook," *Applied Thermal Engineering*, vol. 220, p. 119633, 2023.

[74] M. Aliramezani, C. R. Koch, and M. Shahbakhti, "Modeling, diagnostics, optimization, and control of internal combustion engines via modern machine learning techniques: A review and future directions," *Progress in Energy and Combustion Science*, vol. 88, p. 100967, 2022.

[75] J. Zhang, J. Liu, and Z. Huang, "Improved deep learning method for accurate flow field reconstruction from sparse data," *Ocean Engineering*, vol. 280, p. 114902, 2023.

[76] M. E. Khan and F. Khan, "A comparative study of white box, black box and grey box testing techniques," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 6, 2012.

[77] O. Loyola-Gonzalez, "Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view," *IEEE access*, vol. 7, pp. 154 096–154 113, 2019.

[78] P. Abraham, K. Liu, D. Haworth, D. Reuss, and V. Sick, "Evaluating large-eddy simulation (LES) and high-speed particle image velocimetry (PIV) with phase-invariant proper orthogonal decomposition (POD)," *Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles*, vol. 69, no. 1, pp. 41–59, 2014.

[79] F. Hartmann, S. Buhl, F. Gleiss, P. Barth, M. Schild, S. A. Kaiser, and C. Hasse, "Spatially resolved experimental and numerical investigation of the flow through the intake port of an internal combustion engine," *Oil & Gas Science and Technology–Rev. IFP Energies nouvelles*, vol. 71, no. 1, p. 2, 2016.

[80] C. Hasse, V. Sohm, and B. Durst, "Detached eddy simulation of cyclic large scale fluctuations in a simplified engine setup," *International Journal of Heat and Fluid Flow*, vol. 30, no. 1, pp. 32–43, 2009.

[81] H. Chu, C. Welch, H. Elmestikawy, S. Cao, M. Davidovic, B. Böhm, A. Dreizler, and H. Pitsch, "A combined numerical and experimental investigation of cycle-to-cycle variations in an optically accessible spark-ignition engine," *Flow, Turbulence and Combustion*, vol. 110, no. 1, pp. 3–29, 2023.

[82] C. Welch, M. Schmidt, L. Illmann, A. Dreizler, and B. Böhm, "The influence of flow on cycle-to-cycle variations in a spark-ignition engine: a parametric investigation of increasing exhaust gas recirculation levels," *Flow, Turbulence and Combustion*, pp. 1–24, 2022.

[83] W. Zeng, S. Keum, T.-W. Kuo, and V. Sick, "Role of large scale flow features on cycle-to-cycle variations of spark-ignited flame-initiation and its transition to turbulent combustion," *Proceedings of the Combustion Institute*, vol. 37, no. 4, pp. 4945–4953, 2019.

[84] L. Engelmann, J. Laichter, P. Wollny, M. Klein, S. A. Kaiser, and A. M. Kempf, "Cyclic variations in the flame propagation in an spark-ignited engine: multi cycle large eddy simulation supported by imaging diagnostics," *Flow, Turbulence and Combustion*, vol. 110, no. 1, pp. 91–104, 2023.

[85] F. Berni, G. Cicalese, and S. Fontanesi, "A modified thermal wall function for the estimation of gas-to-wall heat fluxes in CFD in-cylinder simulations of high performance spark-ignition engines," *Applied Thermal Engineering*, vol. 115, pp. 1045–1062, 2017.

[86] N. Van Dam and C. Rutland, "Understanding in-cylinder flow variability using large-eddy simulations," *Journal of Engineering for Gas Turbines and Power*, vol. 138, no. 10, p. 102809, 2016.

[87] N. J. Beavis, S. S. Ibrahim, and W. Malalasekera, "A numerical study of intake valve jet flapping in a gasoline direct injection engine," *International Journal of Powertrains*, vol. 7, no. 1-3, pp. 38–52, 2018.

[88] F. Rulli, S. Fontanesi, A. d'Adamo, and F. Berni, "A critical review of flow field analysis methods involving proper orthogonal decomposition and quadruple proper orthogonal decomposition for internal combustion engines," *International Journal of Engine Research*, vol. 22, no. 1, pp. 222–242, 2021.

[89] P. S. Abraham, X. Yang, S. Gupta, T.-W. Kuo, D. L. Reuss, and V. Sick, "Flow-pattern switching in a motored spark ignition engine," *International Journal of Engine Research*, vol. 16, no. 3, pp. 323–339, 2015.

[90] S. Wu, S. Patel, and M. Ameen, "Investigation of cycle-to-cycle variations in internal combustion engine using proper orthogonal decomposition," *Flow, Turbulence and Combustion*, vol. 110, no. 1, pp. 125–147, 2023.

[91] B. Deng, K. Hou, X. Duan, and Z. Xu, "The correlation between intake fluctuation and combustion ccv (cycle-to-cycle variations) on a high speed gasoline engine: A wide range operating condition study," *Fuel*, vol. 304, p. 121336, 2021.

[92] Z. Ding, K. Truffin, and S. Jay, "Cause-and-effect chain analysis of combustion cyclic variability in a spark-ignition engine using large-eddy simulation, part i: From tumble compression to flame initiation," *Combustion and Flame*, vol. 267, p. 113566, 2024.

[93] K. Taira, S. L. Brunton, S. T. Dawson, C. W. Rowley, T. Colonius, B. J. McKeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley, "Modal

analysis of fluid flows: An overview," *AIAA Journal*, vol. 55, no. 12, pp. 4013–4041, 2017.

[94] F. Zhao and D. L. Hung, "Applications of machine learning to the analysis of engine in-cylinder flow and thermal process: a review and outlook," *Applied Thermal Engineering*, p. 119633, 2022.

[95] J. L. Lumley, "The structure of inhomogeneous turbulent flows," *Atmospheric Turbulence and Radio Wave Propagation*, pp. 166–178, 1967.

[96] K. Liu, D. C. Haworth, X. Yang, and V. Gopalakrishnan, "Large-eddy simulation of motored flow in a two-valve piston engine: Pod analysis and cycle-to-cycle variations," *Flow, Turbulence and Combustion*, vol. 91, pp. 373–403, 2013.

[97] W. Qin, L. Zhou, D. Liu, M. Jia, and M. Xie, "Investigation of in-cylinder engine flow quadruple decomposition dynamical behavior using proper orthogonal decomposition and dynamic mode decomposition methods," *Journal of Engineering for Gas Turbines and Power*, vol. 141, no. 8, 2019.

[98] S. Wu, S. Patel, and M. Ameen, "Investigation of cycle-to-cycle variations in internal combustion engine using proper orthogonal decomposition," *Flow, Turbulence and Combustion*, pp. 1–23, 2022.

[99] P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, pp. 5–28, 2010.

[100] Y. Fu, X. Lin, L. Li, Q. Chu, H. Liu, X. Zheng, C.-H. Liu, Z. Chen, C. Lin, T. K. Tse *et al.*, "A POD-DMD augmented procedure to isolating dominant flow field features in a street canyon," *Physics of Fluids*, vol. 35, no. 2, 2023.

[101] Y. Liu, J. Long, Q. Wu, B. Huang, and G. Wang, "Data-driven modal decomposition of transient cavitating flow," *Physics of Fluids*, vol. 33, no. 11, 2021.

[102] Y. Yuan, K. Zhou, W. Zhou, X. Wen, and Y. Liu, "Flow prediction using dynamic mode decomposition with time-delay embedding based on local measurement," *Physics of Fluids*, vol. 33, no. 9, 2021.

[103] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016, ISBN: 978-1-61197-449-2.

[104] M. Liu, F. Zhao, X. Li, M. Xu, and D. L. Hung, "Dynamic mode decomposition for extracting cycle-to-cycle variation of sidi engine in-cylinder flow under motoring condition," in *Proceedings of the ASME 2020 Internal Combustion Engine Division Fall Technical Conference*. ASME, 2020.

[105] M. Liu, F. Zhao, and D. L. Hung, "A coupled phase-invariant pod and dmd analysis for the characterization of in-cylinder cycle-to-cycle flow variations under different swirl conditions," *Flow, Turbulence and Combustion*, vol. 110, no. 1, pp. 31–57, 2023.

[106] X. Fang, L. Shen, C. Willman, R. Magnanon, G. Virelli, M. H. Davy, and R. Stone, "Manifold reduction techniques for the comparison of crank angle-resolved particle image velocimetry (PIV) data and Reynolds-averaged Navier-Stokes (RANS) simulations in a spark ignition direct injection (SIDI) engine," *International Journal of Engine Research*, vol. 23, no. 8, pp. 1275–1294, 2022.

[107] I. Scherl, B. Strom, J. K. Shang, O. Williams, B. L. Polagye, and S. L. Brunton, "Robust principal component analysis for modal decomposition of corrupt fluid flows," *Physical Review Fluids*, vol. 5, no. 5, p. 054401, 2020.

[108] C. Van Doorne and J. Westerweel, "Measurement of laminar, transitional and turbulent pipe flow using stereoscopic-PIV," *Experiments in Fluids*, vol. 42, pp. 259–279, 2007.

[109] P. Chandramouli, E. Mémin, and D. Heitz, "4D large scale variational data assimilation of a turbulent flow with a dynamics error model," *Journal of Computational Physics*, vol. 412, p. 109446, 2020.

[110] S. L. Brunton, J. Nathan Kutz, K. Manohar, A. Y. Aravkin, K. Morgansen, J. Klemisch, N. Goebel, J. Buttrick, J. Poskin, A. W. Blom-Schieber *et al.*, "Data-driven aerospace engineering: reframing the industry with machine learning," *AIAA Journal*, vol. 59, no. 8, pp. 2820–2847, 2021.

[111] G. Aversano, M. Ferrarotti, and A. Parente, "Digital twin of a combustion furnace operating in flameless conditions: reduced-order model development from CFD simulations," *Proceedings of the Combustion Institute*, vol. 38, no. 4, pp. 5373–5381, 2021.

[112] O. Elharrouss, N. Almaadeed, S. Al-Maadeed, and Y. Akbari, "Image inpainting: A review," *Neural Processing Letters*, vol. 51, pp. 2007–2028, 2020.

[113] J. Jam, C. Kendrick, K. Walker, V. Drouard, J. G.-S. Hsu, and M. H. Yap, "A comprehensive review of past and present image inpainting methods," *Computer Vision and Image Understanding*, vol. 203, p. 103147, 2021.

[114] R. Everson and L. Sirovich, "Karhunen–Loeve procedure for gappy data," *Journal of the Optical Society of America A*, vol. 12, no. 8, pp. 1657–1664, 1995.

[115] H. Gunes, S. Sirisup, and G. E. Karniadakis, "Gappy data: To krig or not to krig?" *Journal of Computational Physics*, vol. 212, no. 1, pp. 358–382, 2006.

[116] S. G. Raben, J. J. Charonko, and P. P. Vlachos, "Adaptive gappy proper orthogonal decomposition for particle image velocimetry data reconstruction," *Measurement Science and Technology*, vol. 23, no. 2, p. 025303, 2012.

[117] P. Saini, C. M. Arndt, and A. M. Steinberg, "Development and evaluation of gappy-POD as a data reconstruction technique for noisy PIV measurements in gas turbine combustors," *Experiments in Fluids*, vol. 57, pp. 1–15, 2016.

[118] A. Nekkanti and O. T. Schmidt, "Gappy spectral proper orthogonal decomposition," *Journal of Computational Physics*, vol. 478, p. 111950, 2023.

[119] T. Murata, K. Fukami, and K. Fukagata, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," *Journal of Fluid Mechanics*, vol. 882, p. A13, 2020.

[120] B. P. Epps and E. M. Krivitzky, "Singular value decomposition of noisy data: noise filtering," *Experiments in Fluids*, vol. 60, no. 8, pp. 1–23, 2019.

[121] S. L. Brunton and J. N. Kutz, *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2022, ISBN: 9781009098489.

[122] P. Dubois, T. Gomez, L. Planckaert, and L. Perret, "Machine learning for fluid flow reconstruction from limited measurements," *Journal of Computational Physics*, vol. 448, p. 110733, 2022.

[123] Y. Kumar, P. Bahl, and S. Chakraborty, "State estimation with limited sensors–a deep learning based approach," *Journal of Computational Physics*, vol. 457, p. 111081, 2022.

[124] T. Nguyen, J. Jewik, H. Bansal, P. Sharma, and A. Grover, "Climatelearn: Benchmarking machine learning for weather and climate modeling," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[125] X. Jin, P. Cheng, W.-L. Chen, and H. Li, "Prediction model of velocity field around circular cylinder over various reynolds numbers by fusion convolutional neural networks based on pressure on the cylinder," *Physics of Fluids*, vol. 30, no. 4, 2018.

[126] M. Morimoto, K. Fukami, and K. Fukagata, "Experimental velocity data estimation for imperfect particle images using machine learning," *Physics of Fluids*, vol. 33, no. 8, 2021.

[127] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015*. Springer, 2015, pp. 234–241.

[128] J. Kaltenborn, C. Lange, V. Ramesh, P. Brouillard, Y. Gurwicz, C. Nagda, J. Runge, P. Nowack, and D. Rolnick, "Climateset: A large-scale climate model dataset for machine learning," *Advances in Neural Information Processing Systems*, vol. 36, pp. 21 757–21 792, 2023.

[129] K. Bao, X. Zhang, W. Peng, and W. Yao, "Deep learning method for super-resolution reconstruction of the spatio-temporal flow field," *Advances in Aerodynamics*, vol. 5, no. 1, p. 19, 2023.

[130] Z. Deng, H. Liu, B. Shi, Z. Wang, F. Yu, Z. Liu, and G. Chen, "Temporal predictions of periodic flows using a mesh transformation and deep learning-based strategy," *Aerospace Science and Technology*, vol. 134, p. 108081, 2023.

[131] S. Ashkboos, L. Huang, N. Dryden, T. Ben-Nun, P. Dueben, L. Gianinazzi, L. Kummer, and T. Hoefler, "Ens-10: A dataset for post-processing ensemble weather forecasts," *Advances in Neural Information Processing Systems*, vol. 35, pp. 21 974–21 987, 2022.

[132] A. Güemes, C. Sanmiguel Vila, and S. Discetti, "Super-resolution generative adversarial networks of randomly-seeded fields," *Nature Machine Intelligence*, vol. 4, no. 12, pp. 1165–1173, 2022.

[133] H. Kim, J. Kim, S. Won, and C. Lee, "Unsupervised deep learning for super-resolution reconstruction of turbulence," *Journal of Fluid Mechanics*, vol. 910, 2021.

[134] T. Li, M. Buzzicotti, L. Biferale, F. Bonaccorso, S. Chen, and M. Wan, "Multi-scale reconstruction of turbulent rotating flows with proper orthogonal decomposition and generative adversarial networks," *Journal of Fluid Mechanics*, vol. 971, 2023.

[135] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.

[136] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, and M. W. Mahoney, "Characterizing possible failure modes in physics-informed neural networks," *Advances in Neural Information Processing Systems*, vol. 34, pp. 26 548–26 560, 2021.

[137] H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, "Physics-informed neural networks for solving reynolds-averaged navier–stokes equations," *Physics of Fluids*, vol. 34, no. 7, 2022.

[138] R. Wang, K. Kashinath, M. Mustafa, A. Albert, and R. Yu, "Towards physics-informed deep learning for turbulent flow prediction," in *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, 2020, pp. 1457–1466.

[139] H. Wang, Y. Liu, and S. Wang, "Dense velocity reconstruction from particle image velocimetry/particle tracking velocimetry using a physics-informed neural network," *Physics of fluids*, vol. 34, no. 1, 2022.

[140] D. Venturi and G. E. Karniadakis, "Gappy data and reconstruction procedures for flow past a cylinder," *Journal of Fluid Mechanics*, vol. 519, pp. 315–336, 2004.

[141] H. Wang, Q. Gao, L. Feng, R. Wei, and J. Wang, "Proper orthogonal decomposition based outlier correction for PIV data," *Experiments in Fluids*, vol. 56, pp. 1–15, 2015.

[142] Z. Luo, L. Wang, J. Xu, Z. Wang, M. Chen, J. Yuan, and A. C. Tan, "Reconstruction of missing flow field from imperfect turbulent flows by machine learning," *Physics of Fluids*, vol. 35, no. 8, 2023.

[143] M. Buzzicotti, F. Bonaccorso, P. C. Di Leoni, and L. Biferale, "Reconstruction of turbulent data with deep generative models for semantic inpainting from TURB-Rot database," *Physical Review Fluids*, vol. 6, no. 5, p. 050503, 2021.

[144] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[145] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. IEEE, 2009, pp. 248–255.

[146] Y. Li, E. Perlman, M. Wan, Y. Yang, C. Meneveau, R. Burns, S. Chen, A. Szalay, and G. Eyink, "A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence," *Journal of Turbulence*, no. 9, 2008.

[147] W. T. Chung, B. Akoush, P. Sharma, A. Tamkin, K. S. Jung, J. Chen, J. Guo, D. Brouzet, M. Talei, B. Savard *et al.*, "Turbulence in focus: Benchmarking scaling behavior of 3d volumetric super-resolution with blastnet 2.0 data," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[148] R. McConkey, E. Yee, and F.-S. Lien, "A curated dataset for data-driven turbulence modelling," *Scientific Data*, vol. 8, no. 1, p. 255, 2021.

[149] F. Bonnet, J. A. Mazari, P. Cinnella, and P. Gallinari, "AirfRANS: High fidelity computational fluid dynamics dataset for approximating reynolds-averaged navier–stokes solutions," in *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[150] R. Zhou, S. Balusamy, M. S. Sweeney, R. S. Barlow, and S. Hochgreb, "Flow field measurements of a series of turbulent premixed and stratified methane/air flames," *Combustion and Flame*, vol. 160, no. 10, pp. 2017–2028, 2013.

[151] M. Meijer, B. Somers, J. Johnson, J. Naber, S.-Y. Lee, L. M. C. Malbec, G. Bruneaux, L. M. Pickett, M. Bardi, R. Payri *et al.*, "Engine combustion

network (ECN): Characterization and comparison of boundary conditions for different combustion vessels," *Atomization and Sprays*, vol. 22, no. 9, 2012.

[152] P. Schiffmann, S. Gupta, D. Reuss, V. Sick, X. Yang, and T.-W. Kuo, "TCC-III engine benchmark for large-eddy simulation of ic engine flows," *Oil & Gas Science and Technology–Rev. IFP Energies nouvelle*, vol. 71, no. 1, p. 3, 2016.

[153] L. F. Richardson, *Weather prediction by numerical process*. Cambridge University Press, 1922, ISBN: 9780511618291.

[154] A. N. Kolmogorov, "Equations of turbulent motion in an incompressible fluid," *Dokl. Akad. Nauk SSSR*, vol. 30, pp. 299–303, 1941.

[155] C. A. De Moura and C. S. Kubrusly, *The Courant-Friedrichs-Lewy (CFL) condition*. Birkhauser, 2013, ISBN: 978-0-8176-8394-8.

[156] H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics*. Pearson Education Limited, 2007, ISBN: 978-0-13-127498-3.

[157] S. N. A. Yusuf, Y. Asako, N. A. C. Sidik, S. B. Mohamed, and W. M. A. A. Japar, "A short review on RANS turbulence models," *CFD Letters*, vol. 12, no. 11, pp. 83–96, 2020.

[158] S. Heinz, "A review of hybrid RANS-LES methods for turbulent flows: Concepts and applications," *Progress in Aerospace Sciences*, vol. 114, p. 100597, 2020.

[159] Z. Han and R. D. Reitz, "Turbulence modeling of internal combustion engines using RNG κ-ε models," *Combustion Science and Technology*, vol. 106, no. 4-6, pp. 267–295, 1995.

[160] F. Wang, R. D. Reitz, C. Pera, Z. Wang, and J. Wang, "Application of generalized RNG turbulence model to flow in motored single-cylinder PFI engine," *Engineering Applications of Computational Fluid Mechanics*, vol. 7, no. 4, pp. 486–495, 2013.

[161] V. K. Krastev, L. Silvestri, and G. Falcucci, "A modified version of the RNG k–ε turbulence model for the scale-resolving simulation of internal combustion engines," *Energies*, vol. 10, no. 12, p. 2116, 2017.

[162] V. Yakhot and S. A. Orszag, "Renormalization group analysis of turbulence. I. basic theory," *Journal of Scientific Computing*, vol. 1, no. 1, pp. 3–51, 1986.

[163] P. Sagaut, *Large eddy simulation for incompressible flows: an introduction*. Springer, 2005, ISBN: 978-3-540-26403-3.

[164] J. Smagorinsky, "General circulation experiments with the primitive equations: I. the basic experiment," *Monthly Weather Review*, vol. 91, no. 3, pp. 99–164, 1963.

[165] M. Lesieur and O. Metais, "New trends in large-eddy simulations of turbulence," *Annual Review of Fluid Mechanics*, vol. 28, no. 1, pp. 45–82, 1996.

[166] M. Germano, U. Piomelli, P. Moin, and W. H. Cabot, "A dynamic subgrid-scale eddy viscosity model," *Physics of Fluids A: Fluid Dynamics*, vol. 3, no. 7, pp. 1760–1765, 1991.

[167] D. Lilly, "A proposed modification of the germano sugrid-scale closure method," *Physics of Fluids A*, vol. 4, pp. 633–635, 1992.

[168] H. Glyde, "Experiments to determine velocities of flame propagation in a side valve petrol engine," *Journal of the Institute of Petroleum Technologists*, vol. 16, pp. 756–776, 1930.

[169] R. J. Adrian and J. Westerweel, *Particle image velocimetry*. Cambridge University Press, 2011, ISBN: 9780521440080.

[170] J. Westerweel, G. E. Elsinga, and R. J. Adrian, "Particle image velocimetry for complex and turbulent flows," *Annual Review of Fluid Mechanics*, vol. 45, no. 1, pp. 409–436, 2013.

[171] E. Baum, B. Peterson, B. Böhm, and A. Dreizler, "On the validation of LES applied to internal combustion engine flows part 1: comprehensive experimental database," *Flow, Turbulence and Combustion*, vol. 92, pp. 269–297, 2014.

[172] C. Welch, M. Schmidt, K. Keskinen, G. Giannakopoulos, K. Boulouchos, A. Dreizler, and B. Boehm, "The effects of intake pressure on in-cylinder gas velocities in an optically accessible single-cylinder research engine," *SAE Technical Paper 2020-01-0792*, 2020.

[173] W. Rodi, "Experience with two-layer models combining the k-epsilon model with a one-equation model near the wall," in *29th Aerospace Sciences Meeting*, 1991, p. 216.

[174] A. Barbato, S. Fontanesi, and A. D'Adamo, "Impact of grid density and turbulence model on the simulation of in-cylinder turbulent flow structures-application to the Darmstadt engine," *SAE Technical Paper 2021-01-0415*, 2021.

[175] C. Rakopoulos, G. Kosmadakis, A. Dimaratos, and E. Pariotis, "Investigating the effect of crevice flow on internal combustion engines using a new simple crevice model implemented in a cfd code," *Applied Energy*, vol. 88, no. 1, pp. 111–126, 2011.

[176] I. Stocchi, J. Liu, C. E. Dumitrescu, M. Battistoni, and C. N. Grimaldi, "Effect of piston crevices on the numerical simulation of a heavy-duty diesel engine retrofitted to natural-gas spark-ignition operation," *Journal of Energy Resources Technology*, vol. 141, no. 11, 2019.

[177] C. Iacovano, F. Berni, A. Barbato, and S. Fontanesi, "A preliminary 1D-3D analysis of the Darmstadt research engine under motored condition," in *75th National ATI Congress*, vol. 197, 2020, pp. 1–12.

[178] A. Yoshizawa, "Statistical theory for compressible turbulent shear flows, with the application to subgrid modeling," *Physics of Fluids*, vol. 29, no. 7, pp. 2152–2164, 1986.

[179] F. Di Mare, R. Knappstein, and M. Baumann, "Application of LES-quality criteria to internal combustion engine flows," *Computers & Fluids*, vol. 89, pp. 200–213, 2014.

[180] I. Ko, F. Rulli, S. Fontanesi, A. d'Adamo, and K. Min, "Methodology for the large-eddy simulation and particle image velocimetry analysis of large-scale flow structures on TCC-III engine under motored condition," *International Journal of Engine Research*, vol. 22, no. 8, pp. 2709–2731, 2021.

[181] V. K. Krastev, A. d'Adamo, F. Berni, and S. Fontanesi, "Validation of a zonal hybrid urans/les turbulence modeling method for multi-cycle engine flow simulation," *International Journal of Engine Research*, vol. 21, no. 4, pp. 632–648, 2020.

[182] D. Freudenhammer, B. Peterson, C.-P. Ding, B. Boehm, and S. Grundmann, "The influence of cylinder head geometry variations on the volumetric intake flow captured by magnetic resonance velocimetry," *SAE International Journal of Engines*, vol. 8, no. 4, pp. 1826–1836, 2015.

[183] H. Zhao, *Laser diagnostics and optical measurement techniques in internal combustion engines*. SAE International, 2012, ISBN: 9780768077667.

[184] M. Fogleman, J. Lumley, D. Rempfer, and D. Haworth, "Application of the proper orthogonal decomposition to datasets of internal combustion engine flows," *Journal of Turbulence*, vol. 5, no. 1, p. 023, 2004.

[185] L. Van Der Maaten, E. O. Postma, and H. J. Van Den Herik, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, no. 66-71, p. 13, 2009.

[186] J. L. Callaham, S. L. Brunton, and J.-C. Loiseau, "On the role of nonlinear correlations in reduced-order modelling," *Journal of Fluid Mechanics*, vol. 938, 2022.

[187] A. F. Hussain, "Coherent structures—reality and myth," *Physics of fluids*, vol. 26, no. 10, pp. 2816–2850, 1983.

[188] F. Rulli, S. Fontanesi, A. D'Adamo, and F. Berni, "A critical review of flow field analysis methods involving proper orthogonal decomposition and quadruple proper orthogonal decomposition for internal combustion engines," *International Journal of Engine Research*, vol. 22, no. 1, pp. 222–242, 2021.

[189] G. W. Stewart, "On the early history of the singular value decomposition," *SIAM Review*, vol. 35, no. 4, pp. 551–566, 1993.

[190] K. Taira, S. L. Brunton, S. T. M. Dawson, C. W. Rowley, T. Colonius, B. J. Mckeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley, "Modal analysis of fluid flows: An overview," *AIAA Journal*, vol. 55, no. 12, pp. 4013–4041, 2017.

[191] B. P. Epps and A. H. Techet, "An error threshold criterion for singular value decomposition modes extracted from PIV data," *Experiments in Fluids*, vol. 48, no. 2, pp. 355–367, 2010.

[192] A. Towne, O. T. Schmidt, and T. Colonius, "Spectral proper orthogonal decomposition and its relationship to dynamic mode decomposition and resolvent analysis," *Journal of Fluid Mechanics*, vol. 847, pp. 821–867, 2018.

[193] M. Sieber, C. O. Paschereit, and K. Oberleithner, "Spectral proper orthogonal decomposition," *Journal of Fluid Mechanics*, vol. 792, pp. 798–828, 2016.

[194] P. J. Schmid, "Dynamic mode decomposition and its variants," *Annual Review of Fluid Mechanics*, vol. 54, no. 1, pp. 225–254, 2022.

[195] K. K. Chen, J. H. Tu, and C. W. Rowley, "Variants of dynamic mode decomposition: boundary condition, Koopman, and Fourier analyses," *Journal of Nonlinear Science*, vol. 22, no. 6, pp. 887–915, 2012.

[196] M. R. Jovanović, P. J. Schmid, and J. W. Nichols, "Sparsity-promoting dynamic mode decomposition," *Physics of Fluids*, vol. 26, no. 2, 2014.

[197] R. Futrzynski and G. Efraimsson, "Dymode: A parallel dynamic mode decomposition software," KTH Royal Institute of Technology, Tech. Rep., 2015, ISBN: 978-91-7595-386-1. [Online]. Available: https://www.diva-portal.org/smash/get/diva2:786623/FULLTEXT01.pdf

[198] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, "On dynamic mode decomposition: Theory and applications," *Journal of Computational Dynamics*, vol. 1, no. 2, pp. 391–421, 2014.

[199] E. G. Birgin and J. M. Martínez, *Practical augmented Lagrangian methods for constrained optimization*. SIAM, 2014, ISBN: 9781611973365.

[200] D. G. Luenberger, *Optimization by vector space methods*. John Wiley & Sons, 1969, ISBN: 9780470349106.

[201] M. R. Hestenes, "Multiplier and gradient methods," *Journal of Optimization Theory and Applications*, vol. 4, no. 5, pp. 303–320, 1969.

[202] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[203] T.-Y. Lam, *Introduction to quadratic forms over fields*. American Mathematical Society, 2005, vol. 67, ISBN: 9781470421083.

[204] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine learning for fluid mechanics," *Annual Review of Fluid Mechanics*, vol. 52, pp. 477–508, 2020.

[205] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of Control, Signals and Systems*, vol. 2, no. 4, pp. 303–314, 1989.

[206] P. Baldi and K. Hornik, "Neural networks and principal component analysis: Learning from examples without local minima," *Neural Networks*, vol. 2, no. 1, pp. 53–58, 1989.

[207] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *International Journal of Engineering Applied Sciences and Technology*, vol. 4, no. 12, pp. 310–316, 2020.

[208] S. Narayan, "The generalized sigmoid activation function: Competitive supervised learning," *Information Sciences*, vol. 99, no. 1-2, pp. 69–82, 1997.

[209] H. Zhang, T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel, "Efficient neural network robustness certification with general activation functions," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[210] K. O'shea and R. Nash, "An introduction to convolutional neural networks," *arXiv preprint arXiv:1511.08458*, 2015.

[211] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[212] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82 031–82 057, 2021.

[213] S. Lee, M. Negishi, H. Urakubo, H. Kasai, and S. Ishii, "Mu-net: Multi-scale u-net for two-photon microscopy image denoising and restoration," *Neural Networks*, vol. 125, pp. 92–103, 2020.

[214] C.-M. Fan, T.-J. Liu, and K.-H. Liu, "Sunet: Swin transformer unet for image denoising," in *2022 IEEE International Symposium on Circuits and Systems (ISCAS)*. IEEE, 2022, pp. 2333–2337.

[215] X. Hu, M. A. Naiel, A. Wong, M. Lamm, and P. Fieguth, "Runet: A robust unet architecture for image super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[216] V. Ghodrati, J. Shao, M. Bydder, Z. Zhou, W. Yin, K.-L. Nguyen, Y. Yang, and P. Hu, "MR image reconstruction using deep learning: evaluation of network structure and loss functions," *Quantitative OIaging in Medicine and Surgery*, vol. 9, no. 9, p. 1516, 2019.

[217] M. Gavish and D. L. Donoho, "The optimal hard threshold for singular values is $4/\sqrt{3}$," *IEEE Transactions on Information Theory*, vol. 60, no. 8, pp. 5040–5053, 2014.

[218] H. Chu, C. Welch, H. Elmestikawy, S. Cao, M. Davidovic, B. Böhm, A. Dreizler, and H. Pitsch, "A combined numerical and experimental investigation of cycle-to-cycle variations in an optically accessible spark-ignition engine," *Flow, Turbulence and Combustion*, pp. 1–27, 2022.

[219] L. Engelmann, J. Laichter, P. Wollny, M. Klein, S. A. Kaiser, and A. M. Kempf, "Cyclic variations in the flame propagation in an spark-ignited engine: Multi cycle large eddy simulation supported by imaging diagnostics," *Flow, Turbulence and Combustion*, pp. 1–14, 2022.

[220] B. Enaux, V. Granet, O. Vermorel, C. Lacour, L. Thobois, V. Dugué, and T. Poinsot, "Large eddy simulation of a motored single-cylinder piston engine: numerical strategies and validation," *Flow, Turbulence and Combustion*, vol. 86, no. 2, pp. 153–177, 2011.

[221] M. M. Ameen, X. Yang, T.-W. Kuo, and S. Som, "Using LES to simulate cycle-to-cycle variability during the gas exchange process," in *Internal Combustion Engine Division Fall Technical Conference*, vol. 58325. ASME, 2017.

[222] F. Zhao, M. Liu, P. Ge, D. L. Hung, X. Li, M. Xu, X. Yang, and C. Idicheria, "Multi-plane time-resolved particle image velocimetry (PIV) flow field measurements in an optical spark-ignition direct-injection (SIDI) engine for large-eddy simulation (LES) model validations," *Oil & Gas Science and Technology–Revue d'IFP Energies nouvelles*, vol. 74, p. 52, 2019.

[223] C. Willman, B. Scott, R. Stone, and D. Richardson, "Quantitative metrics for comparison of in-cylinder velocity fields using particle image velocimetry," *Experiments in Fluids*, vol. 61, no. 2, pp. 1–16, 2020.

[224] M. El-Adawy, M. Heikal, A. R. A. Aziz, M. Siddiqui, and H. A. A. Wahhab, "Experimental study on an ic engine in-cylinder flow using different steady-state flow benches," *Alexandria Engineering Journal*, vol. 56, no. 4, pp. 727–736, 2017.

[225] G. Kohl, K. Um, and N. Thuerey, "Learning similarity metrics for numerical simulations," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5349–5360.

[226] R. Hummel, "Image enhancement by histogram transformation," *Computer Graphics and Image Processing*, vol. 6, 1977.

[227] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[228] J. J. Bramburger, J. N. Kutz, and S. L. Brunton, "Data-driven stabilization of periodic orbits," *IEEE Access*, vol. 9, pp. 43 504–43 521, 2021.

[229] L. Month and R. H. Rand, "An application of the poincaré map to the stability of nonlinear normal modes," *Journal of Applied Mechanics*, vol. 47, pp. 645–651, 1980.

[230] S. Baker, X. Fang, L. Shen, C. Willman, J. Fernandes, F. Leach, and M. Davy, "Dynamic mode decomposition for the comparison of engine in-cylinder flow fields from particle image velocimetry (PIV) and Reynolds-averaged Navier–Stokes (RANS) simulations," *Flow, Turbulence and Combustion*, vol. 111, no. 1, pp. 115–140, 2023.

[231] M. Schmidt, C.-P. Ding, B. Peterson, A. Dreizler, and B. Böhm, "Near-wall flame and flow measurements in an optically accessible si engine," *Flow, Turbulence and Combustion*, vol. 106, no. 2, pp. 597–611, 2021.

[232] T. Grenga, J. F. MacArt, and M. E. Mueller, "Dynamic mode decomposition of a direct numerical simulation of a turbulent premixed planar jet flame: convergence of the modes," *Combustion Theory and Modelling*, vol. 22, no. 4, pp. 795–811, 2018.

[233] A. d'Adamo, S. Breda, F. Berni, and S. Fontanesi, "Understanding the origin of cycle-to-cycle variation using large-eddy simulation," *SAE International Journal of Engines*, vol. 12, no. 1, pp. 79–100, 2019.

[234] K. Bizon, G. Continillo, K. Leistner, E. Mancaruso, and B. Vaglieco, "Pod-based analysis of cycle-to-cycle variations in an optically accessible diesel engine," *Proceedings of the Combustion Institute*, vol. 32, no. 2, pp. 2809–2816, 2009.

[235] J. Rabault, J. A. Vernet, B. Lindgren, and P. H. Alfredsson, "A study using piv of the intake flow in a diesel engine cylinder," *International Journal of Heat and Fluid Flow*, vol. 62, pp. 56–67, 2016.

[236] B. Petersen and P. Miles, "PIV measurements in the swirl-plane of a motored light-duty diesel engine," *SAE International Journal of Engines*, vol. 4, no. 1, pp. 1623–1641, 2011.

[237] A. Hatamizadeh, Y. Tang, V. Nath, D. Yang, A. Myronenko, B. Landman, H. R. Roth, and D. Xu, "UNETR: Transformers for 3D medical image segmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2022, pp. 574–584.

[238] M. J. Cardoso, W. Li, R. Brown, N. Ma, E. Kerfoot, Y. Wang, B. Murrey, A. Myronenko, C. Zhao, D. Yang *et al.*, "Monai: An open-source framework for deep learning in healthcare," *arXiv preprint arXiv:2211.02701*, 2022.

[239] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2536–2544.

[240] H. Liu, Z. Wan, W. Huang, Y. Song, X. Han, and J. Liao, "Pd-gan: Probabilistic diverse gan for image inpainting," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9371–9381.

[241] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[242] I. Goodfellow, "Nips 2016 tutorial: Generative adversarial networks," *arXiv preprint arXiv:1701.00160*, 2016.

[243] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," *arXiv preprint arXiv:1511.06434*, 2015.

[244] J. Yu, L. Lu, X. Meng, and G. E. Karniadakis, "Gradient-enhanced physics-informed neural networks for forward and inverse PDE problems," *Computer Methods in Applied Mechanics and Engineering*, vol. 393, p. 114823, 2022.

[245] H. Kang, Y. Kim, T.-T.-H. Le, C. Choi, Y. Hong, S. Hong, S. W. Chin, and H. Kim, "A new fluid flow approximation method using a vision transformer and a U-shaped convolutional neural network," *AIP Advances*, vol. 13, no. 2, 2023.

[246] J. Jiang, G. Li, Y. Jiang, L. Zhang, and X. Deng, "TransCFD: A transformer-based decoder for flow field prediction," *Engineering Applications of Artificial Intelligence*, vol. 123, p. 106340, 2023.

[247] Y. Xu, Y. Sha, C. Wang, and Y. Wei, "Estimation of cavitation velocity fields based on limited pressure data through improved U-shaped neural network," *Physics of Fluids*, vol. 35, no. 8, 2023.

[248] H. Hill, C.-P. Ding, E. Baum, B. Böhm, A. Dreizler, and B. Peterson, "An application of tomographic PIV to investigate the spray-induced turbulence in a direct-injection engine," *International Journal of Multiphase Flow*, vol. 121, p. 103116, 2019.

[249] L. Donato, C. Galletti, and A. Parente, "Self-updating digital twin of a hydrogen-powered furnace using data assimilation," *Applied Thermal Engineering*, vol. 236, p. 121431, 2024.