UNIVERSITY OF CALIFORNIA, MERCED

Experiments, analysis and modeling in fluid stretching

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Physics

by

Diego Tapia Silva

Committee in charge:

Professor Bin Liu, Chair
Professor Dustin Kleckner
Professor Shilpa Khatri
Professor Kevin Mitchell

2024

The dissertation of Diego Tapia Silva is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____
(Professor Dustin Kleckner)

_____
(Professor Shilpa Khatri)

_____
(Professor Kevin Mitchell)

_____
(Professor Bin Liu, Chair)

University of California, Merced

2024

DEDICATION

Dedicado a mi madre, Vianey Patricia Silva Campuzano, y para mi padre, Guillermo Reyes Nava.

EPIGRAPH

. . . y que la verdad nunca se rinda.

—René Pérez Joglar

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to first thank my family—my mother, Vianey Patricia Silva Campuzano; my father, Guillermo Reyes Nava; my brother, Amauri Tapia; and my sister, Kenya Tapia Silva—for their unwavering love and support. I am very proud of where we have gotten as a family, carrying with us our roots and never forgetting to be kind and supportive. The love I have for you all has been my greatest motivation, driving me not only to start but also to finish this journey, even through the tough times we have shared.

Secondly, I would like to express my heartfelt gratitude to my advisors, Dustin Kleckner and Shilpa Khatri, for their unwavering support during difficult life transitions. Their guidance has been truly invaluable, and I could not have asked for better mentors. I am deeply thankful for their patience and for their words of encouragement during moments when experiments—as we all have lived through as experimentalist—did not go well and were often discouraging. Lastly, I am profoundly grateful for the skills I have developed as a researcher under their mentorship

I also want to thank Bin Liu, my undergraduate advisor, who welcomed me into his lab and gave me my first research experience. Additionally, I want to express my gratitude to Kevin Mitchell, with whom I took a class. His course inspired me to build my coding expertise, motivating me to take numerical classes and establish a foundation for scientific analysis.

Lastly, I want to thank my friends for the memories we have created together during this chapter of my life. Without all of you, this experience would have not been as wonderful and as eventful as it has been. The love I have received from everyone during this time fills me with immense gratitude.

VITA

2024	Ph. D. in Physics, University of California, Merced
2018	B. S. in Physics, University of California, Merced

PUBLICATIONS

Tapia Silva, D., Cooper, C. J., Mandel, T. L., Khatri, S., & Kleckner, D. (2024). High-speed two-color scanning volumetric laser-induced fluorescence. Experiments in Fluids, 65(7), 100. Springer Berlin Heidelberg. https://doi.org/10.21203/rs.3.rs-2977609/v1.

Faaland, S., Tapia Silva, D., & Kleckner, D. (2023). Stretching behavior of knotted and unknotted flow fields. arXiv. https://arxiv.org/abs/2306.17393

Tapia Silva, D., Cooper, C. J., Khatri, S., & Kleckner, D. (in preparation). Finite time Lyapnuov exponents for Lagrangian particle tracking data.

ABSTRACT OF THE DISSERTATION

Experiments, analysis and modeling in fluid stretching

by

Diego Tapia Silva
Doctor of Philosophy in Physics
University of California Merced, 2024
Professor Bin Liu, Chair

In this thesis, we investigate mixing and transport mechanisms within complex flows, with a particular interest on stretching. Stretching, driven by localized deformation within the fluid domain, provides insight into the underlying dynamics that drive the formation of coherent structures across a wide range of spatial scales. By measuring particle separation rates, we can quantify the extent of deformation in the fluid. Regions that experience significant stretching reveal coherent structures that remain stable over finite time intervals. Understanding how stretching operates within fluid domains is essential for comprehending the flow dynamics of both experimental and simulated systems. To investigate these mechanisms, we introduce a novel high-speed imaging technique to directly capture the fluid dynamics of complex flows. Additionally, we develop robust computational methods to analyze and process the inherently noisy data, enabling accurate insights into the deformation and structure formation in the fluid domain. Finally, we demonstrate the application of our high-speed imaging system and computational tools by studying a bio-inspired system, the flows generated by a pulsing soft robotic coral.

Chapter 1

Introduction

Fluid dynamics has long captured the interest of scientists across disciplines due to its relevance in extensive and diverse range of systems, from bacterial locomotion [2, 3] to the behavior of galaxies [4, 5]. For experimentalists, imaging techniques play a central role in directly capturing the flow dynamics [6, 7, 8], while computational work often focuses on simulating flow velocity and pressure [9, 10, 11]. In this thesis, we introduce and utilize innovative experimental and computational techniques to investigate fundamental properties of complex fluid flows. We focus on two main themes: 1) developing versatile imaging and analysis methods to process noisy experimental particle tracking data sets which enables us to study 2) fundamental flow properties such as vorticity, stretching, and mixing dynamics. Our contributions lie in advancing high speed volumetric imaging for experiments, and applying robust numerical tools to analyze both experimental and simulated flows.

Progress in experimental fluid mechanics is often limited by techniques for measuring flow fields and the resulting dynamics[12]. This is especially true for chaotic flows – such as turbulence – which necessitate single shot 3D measurements [13, 14]. In recent decades, high-speed imaging has been made possible through the development of low light high-speed cameras, high-speed scanning mirrors and the development of mass data processing techniques [15]. The development of the digital camera resulted in experimental two-dimensional imaging techniques such as planar laser-induced fluorescence (PLIF), particle imaging velocimetry (PIV), and

particle tracking velocimetry (PTV). PLIF works by shining a laser sheet through a fluid sample that is embedded with a fluorescent tracer [16, 17]. The fluorescent signal that arises from the laser sheet exciting the tracer is then captured by a camera. PIV and PTV rely on imaging a fluid sample that is embedded with particle tracers [18, 19, 20, 21].

In PIV, the fluid domain is divided into grid cells. This allows for measuring the average displacement of particles between consecutive frames, resulting in a resolved Eulerian velocity field across the grid cells [7, 22]. In PTV, individual tracer particles are tracked over a finite time, resulting in a Lagrangian velocity field computed directly at each particle's position [23, 24]. In Chapter 2 we present a novel high speed two-color volumetric imaging technique. We showcase the capabilities of the imaging system by imaging deep cubic volumes for a test experiment, flow past a sphere. Direct flow visualization is provided through the use of fluorescent dye and particle tracking velocity data is collected simultaneously by tracking fluorescent labeled particles.

In addition to being able to compute velocity fields, an essential aspect of fluid dynamics involves understanding vorticity. Helmholtz theorem states that for inviscid, incompressible flows, the vorticity is transported with the fluid. This allows the flow to be visualized as a bundle of vortex lines that move with the fluid. The stability and dynamics of vortex lines have been studied in both computational and experimental work [25, 26, 27], notably showing that simple vortex structures, e.g., circular vortex rings, exhibit stability as they get advected over in time [28, 29]. In chapter 3 we explore the stability of more complex vortex configurations, knotted vortices and linked rings. These complex vortices exhibit inherent unstable behavior undergoing stretching and reconnection events [30], mirroring turbulent behavior where energy cascades to smaller scales until it dissipates. To simplify the complexity of the problem, we use 'frozen' flow fields generated by static vortex configurations and seed a material line in the fluid domain. We quantify the stretching behavior of material lines under the velocity field generated by complex vortices, providing insights into to why, where and when these vortices stretch. The areas where material lines experience the greatest stretching is analogous to

where the fluid domain itself is undergoing significant deformation.

Finite time Lyapnunov exponents (FTLEs) fields provide important information on mixing and transport processes [31, 32]. To resolve these fields, we need to have access to particle data, either in the form of Eulerian velocity fields or Lagrangian particle trajectories [33]. Particle data is used to construct a flow map which indicates the final positions of particles after a finite time, based on their initial locations. The flow map deforms as it is advected over time, and this deformation is quantified by the Cauchy strain tensor. By extracting the eigenvalues of the Cauchy deformation tensor, which correspond to the rate of separation between neighboring particles, the FTLE field can be computed. Regions characterized by maximum particle separation, often depicted as narrow ridges of high FTLE values [34], indicate areas of significant stretching within the fluid domain. In chapter 4 we present the flow map compilation (FMC) method [35, 36] for the time efficient and accurate computation of FTLE fields. The need for accurate characterization of flow properties from noisy and irregular particle distributions that are inherent to experimental particle tracking data sets, motivates the development of a polynomial interpolator that outperforms, both in speed and accuracy, standard radial basis function interpolation.

Xeniid corals display intriguing behavior characterized by the pulsing of their tentacles. Unlike most animals, these corals exhibit active motion for purposes other than locomotion [37, 1]. Inspired by these animals, we fabricate a soft-robotic coral that mimics the behavior of the corals. In chapter 5 we study the flow dynamics generated by the robot using the novel methods developed in the previous chapters. Particle tracking data sets are collected via our novel high-speed imaging technique and apply the numerical tools, developed throughout this work, to compute the velocity, vorticity fields and the associated FTLE fields at different stages of the robots pulsing period. Our study of bio-inspired flows, generated by the robotic coral, highlights the versatility of our experimental and analysis tools, enabling the exploration of fundamental fluid flow phenomena with a particular focus on fluid stretching.

Chapter 2

High speed volumetric imaging

Traditionally, imaging of fluid dynamics has been restricted to planar investigations. However, fluids flows are intrinsically three-dimensional, and hence to experimentally capture the full dynamics volumetric imaging is necessary. For example, planar imaging techniques have been widely applied to study the flow dynamics of animals [38, 39, 1]. In one study, volumetric PIV imaging was applied to study hydrodynamics of a shark tail [40], leading to qualitatively different results than previous work using planar PIV on the wake structures [41, 42]. In this particular case a dual-ring vortex structure shed from the tails could not be fully resolved from the planar measurements alone, leading to an incorrect interpretation of the overall flow structure. Similarly, in the study of turbulence intrinsically 3D mechanisms like vortex stretching can not be resolved from planar measurements alone [43].

PIV, PTV and PLIF can be extended to three-dimensional imaging. One approach is to integrate a system of multiple cameras, or stereoscopes, to capture unique viewing angles enabling the reconstruction of volumes from the various viewing perspectives. This approach is implemented in tomographic particle image velocimetry (TPIV) [44], tomographic particle tracking velocimetry (TPTV) [45] and tomographic laser-induced fluorescence (TLIF) [46]. An alternative is to image *volumes* of the fluid by sectioning slices of the sample with a laser sheet that is scanned in time [47]. This approach is used in scanning volumetric laser induced fluorescence (SVLIF); it has the benefit of simpler hardware (i.e. a single camera)

and much simpler post-processing of the resulting data. Recent technological developments – e.g. imaging hardware and mass data processing techniques – now make it possible to achieve high spatial and temporal resolutions in volumetric imaging [48, 49].

Here, we present a novel high-speed two-color scanning volumetric laser-induced fluorescence (H2C-SVLIF) technique. The core equipment of the imaging system consists of a single high-speed camera and a scanning laser sheet produced by pair of pulsed ND: YAG lasers which enable two-color imaging. The system is designed such that it can be easily adaptable to a diverse array of experimental set-ups by readily tuning its temporal and spatial resolutions. Notably, our implementation includes open source software which corrects for perspective and other distortion effects when the recorded volumes are displayed or analyzed, removing the need for telecentric lenses or other complicated optical systems. As a result, the system is incredibly versatile, able to adapt to a wide range of scales and image deep (cubic) volumes if desired. This system operates at the limit of fundamental and practical limits to volumetric imaging in terms of both speed and resolution.

Although other volumetric two-color techniques have been showcased, e.g, [50] and [51], these approaches have been constrained to small volume of interests (VOIs) with dimensions $14 \times 14 \times 24$ mm^3 and $5 \times 5 \times 50$ mm^3, respectively. The method presented here is not subject to these limitations; for example, in section 2.6 we present volumetric imaging with dimensions of $159 \times 159 \times 168$ mm^3. We conduct measurements with data throughput of 1.7×10^{10} voxels/second, which is of order 10–10^4 times more data throughput than previous two-color volumetric imaging approaches.

The apparatus described here is also the first to demonstrate high-speed two-color scans for the simultaneous acquisition of LIF and PTV data. Moreover, our technique operates at video rates or faster – greater than 60 Hz – with the use of a single high-speed camera which operates at frame rates greater than 70 kHz. Our imaging demonstrates a notable advancement over existing techniques due to its enhanced imaging capabilities, flexibility and adaptability, rendering it well suited to a wide range of problems in fluids that necessitate high speed single-shot

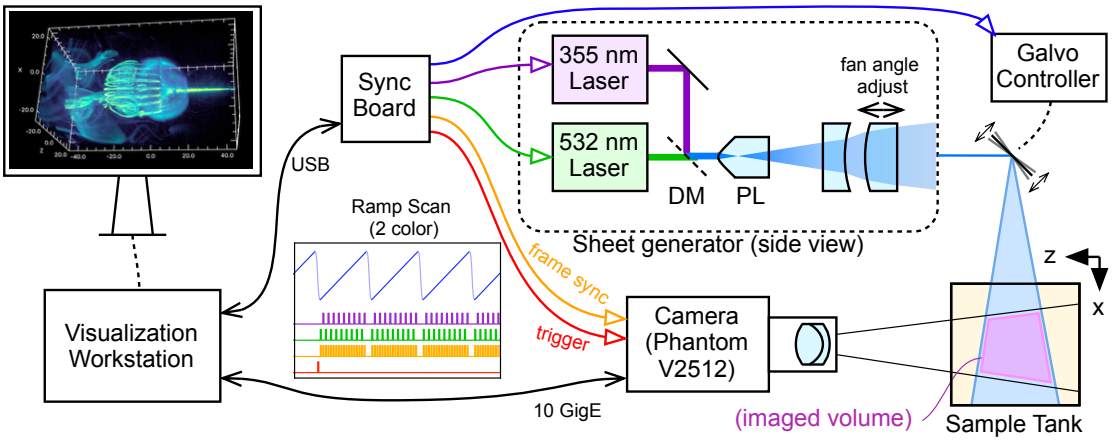


Figure 2.1: A schematic of the H2C-SVLIF control system. The entire setup is synchronized using a custom designed board controlled through a GUI in the visualization workstation. The synchronization board generates signals which are sent to the lasers triggering them on alternating frames (purple and green), the galvanometer (blue), and high-speed camera (yellow and red). Details of the laser sheet generator are given in fig. 2.2.

volumetric imaging.

As a proof of concept, we conduct H2C-SVLIF measurements for flow past a sphere. We use two separate fluorescent dyes excited by a pair of pulsed lasers to simultaneously track small tracer particles and dyed volumes of fluid. While the focus of this paper is primarily the imaging system itself, we also provide the demonstrated data processing and visualization techniques that are easily expedited using our own novel custom open source software [52]. This software has the ability to correct for spatial distortion defects in real time while simultaneously overlaying arbitrary 3D data (e.g. the location of tracked particle or dyed lines). Moreover, non-uniform illumination defects, present in the volumes acquired, are also corrected for in the pre-processing stage.

The outline of this chapter is as follows: section 2.1 presents the basic design and experimental hardware used in our imaging system, section 2.2 discusses the fundamental and practical limits to H2C-SVLIF, section 2.3 addresses challenges posed by processing large data sets and how our open source in-house viewing software is adapted to handle these data sets. In section 2.4, we discuss the post-processing for distortion correction, calibration for the captured volumes, and

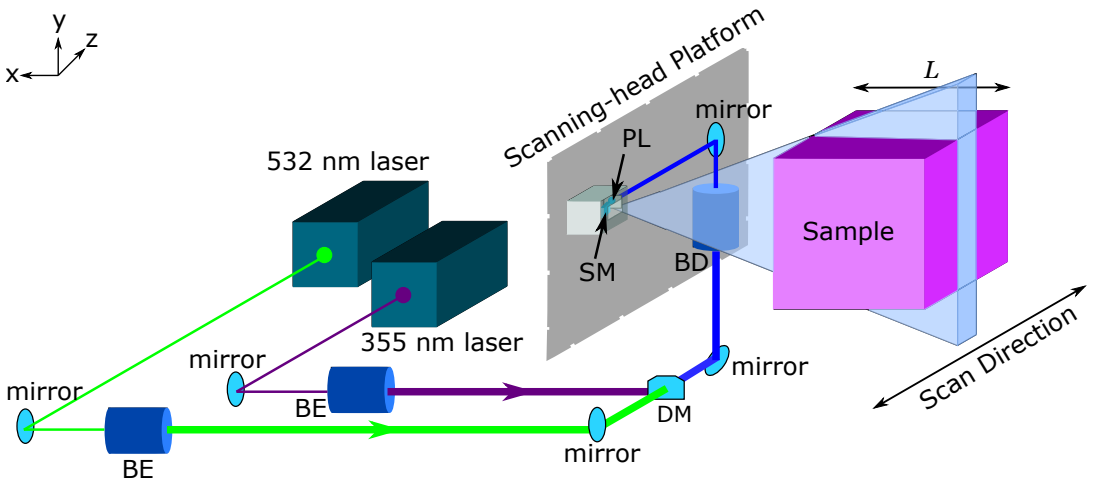


Figure 2.2: A schematic of the laser sheet generator. The laser sheet is produced via a pair of pulsed ND: YAG lasers of wavelengths 355 nm and 532 nm. The laser beams pass through a pair of beam expanders (BE). The beams then travel to a dichroic mirror (DM) which combines the beams. From there, the beams get deflected up to a scanning-head platform and pass through a beam de-expander (BD). The scanning-head platform can be manually translated to adjust to a variety of sample heights. Finally, the beams get deflected into the Powell lens (PL) and into the scanning mirror (SM). The resulting laser sheet is then swept across a sample.

correction of illumination defects. In section 2.5, we discuss the process for the tracking of the particles and the interpolation method for the reconstruction of a Eulerian velocity field. In section 2.6 we present results from our example system, followed up by the conclusion in section 2.7. The collection of particle tracking data allows us to compute FTLE fields, and by building upon this imaging technique, we develop analytical tools to study stretching and mixing mechanisms from these data sets. As demonstrated in chapter 4, these tools enable the analysis and processing of particle tracking data.

2.1 Imaging hardware

The H2C-SVLIF system includes a laser sheet generator, a high-speed camera (*Phantom* V2512), and a custom-designed synchronization board. This board – which is controlled through a visualization workstation – synchronizes the camera

and laser sheet generator (fig. 2.1). To image in three dimensions, the laser sheet is swept across the sample while the high-speed camera records the fluorescent emission slices of the sample. The system is designed to be highly adjustable: the scan width and height are adjustable to image systems at a variety of scales. The scanning laser sheet is generated by a mobile integrated platform, allowing the position and height of the laser sheet to be easily adjusted (fig. 2.2). The laser sources for the sheet are a pair of diode-pumped lasers (*Spectra Physics* Explorer Explorer One XP 355-2 and One XP 532-5). The beams are first expanded to 3 mm, and then combined on a dichroic mirror and directed upwards to the scanning head platform. The head has a manually adjustable height; the 3 mm width of the lasers in the vertical beam section ensures that the divergence of the beams is small enough that this does not significantly affect the beam size over the range of accessible heights. The beams are then de-expanded to the appropriate diameter for the Powell lens (1 mm), which spreads the laser into a uniform vertical sheet. (If needed, the collimation adjustment on the final beam expander can be used to adjust the focus of the final sheet.) Finally, the laser is reflected off a galvonometer mirror (*Thorlabs* GVS002) and directed to the sample.

The high-speed camera, galvanometer, and the lasers are synchronized using the custom designed synchronization board; schematics for this board are publicly available on GitHub [53]. This board sends an analog sawtooth signal to the galvanometer controller to define the scanned volume; this can be adjusted in software as needed. Two digital pulse signals are dispatched to the pulsed lasers, inducing fluorescence at distinct wavelengths during alternate frames. This allows a monochrome camera to be used, improving both effective resolution and signal to noise ratio as compared to cameras with color chips. A digital frame sync signal is sent to the camera to trigger synchronized acquisition. A second trigger signal is sent to the camera to designate the start of the first recorded volume. This entire setup is controlled through a visualization workstation connected to the synchronization board through a USB connection. The camera is also connected to the workstation through a 10 Gb ethernet port, allowing for fast download of large data sets.

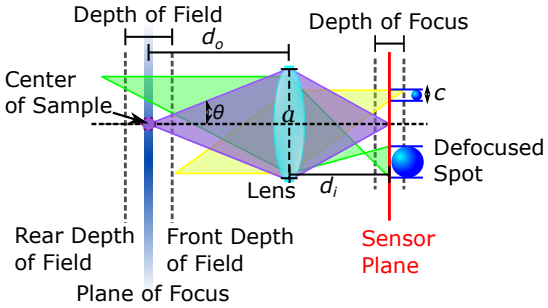


Figure 2.3: Schematic of the focusing optics of the camera. The defocused spot size occurs for distances outside the depth of field, d_{field}. The permissible circle of confusion, c, occurs at the rear and at the front depth of field. The maximum lens aperture size, a, depends on the sample angle, θ. Here, d_o and d_i are the distance to object and distance to image respectively.

2.2 Fundamental and practical limits to high-speed volumetric imaging

Volumetric imaging systems are subject to constraints in resolution, speed and signal to noise ratio. These limits may either be fundamental physical constraints (diffraction and shot noise) or practical hardware limitations (camera speed, sensitivity, and data storage). We discuss each of these below, and arrive at practical limits for high-speed volumetric imaging with the currently available hardware. This discussion is tailored to guide those involved in high-speed volumetric imaging techniques as the same constraints are applicable to their systems.

2.2.1 Image resolution

Ideally, the laser sheet would maintain the same thickness throughout the sample, but in practice this is limited by diffraction. Although we are only focusing the laser sheet along a single axis, the beam parameters can still be computed with standard Gaussian optics [54], assuming the source laser has a Gaussian profile (as is the case for the lasers used in our sheet generator). A reasonable balance between the thickness of the sheet and the distance over which it is focused can be obtained by setting the Rayleigh length, $z_R = \frac{n_s \pi w_0^2}{\lambda}$, to be half of the sample size in the direction of laser travel, L_x (fig. 2.2), where n_s is the index of refraction of

the sample, w_0 is the waist of the Gaussian beam, and λ is the vacuum wavelength of the laser.

To ensure uniform sampling, we set the depth of a single voxel to be equal to the the full width half maximum (FWHM) of the laser sheet. This results in a voxel size $h_{z,dl} = w_0\sqrt{2\ln 2} = \sqrt{\frac{\lambda L_x \ln 2}{\pi n_s}}$. This should be regarded as the maximum useful resolution; higher resolutions would result in a negligible increase in information in the measured volume. Although the resolution limits above were based on the laser sheet, the same fundamental limits apply to the camera system and corresponding lens, i.e. $h_{x,dl} = h_{y,dl} = \sqrt{\frac{\lambda L_z \ln 2}{\pi n_s}}$. Table 2.1 shows the diffraction limited resolutions as a function of sample size.

Table 2.1: The diffraction limited resolutions for cubic volumes, for the 355 nm wavelength laser and the 532 nm wavelength laser, and an index of refraction of the sample, $n_s = 1.33$, for water.

λ (nm)	$L_x = L_y = L_z$ (mm)	$h_{z,dl}$ (mm)	$N_{z,dl}$
355	500	0.172	2910
	250	0.120	2060
	100	0.077	1300
	50	0.054	920
532	500	0.148	2380
	250	0.115	1680
	100	0.094	1060
	50	0.067	750

For large volume sizes, $L_z \gtrapprox 100$ mm, the diffraction limited resolution, $N_{x,dl}$, exceeds the image resolutions of commonly available high-speed cameras. As a result, it is sometimes desirable to under-resolve the image. This brings additional benefits as it allows for an increase in the physical size of the aperture in the camera, which corresponds to a brighter image for the same amount of laser power. Provided our desired image resolution, N_x, is lower than the diffracted limited resolution $N_{x,dl}$, we can approximate the defocusing with a ray optics approach

[55]. Here we define an F-number for the lens aperture: $F = f/a$, where a is the aperture diameter and f is the focal length. (Note that per our definition, $F = 8$ corresponds to 'f/8', as it is usually specified on camera lenses.) To compute the acceptable lens aperture, we will set the 'circle of confusion' (the size of a defocused spot at the edge of the volume) to $c = 2p$, where p is the pixel pitch on the camera sensor ($p = 28$ μm for the Phantom V2512). If we then require that the depth of field be equal to L_z, we can obtain:

$$F = \frac{M^2 L_z}{2 n_s c} = \frac{p L_z}{4 n_s h^2} \tag{2.1}$$

where $M = p/h$ is the image magnification, and h is the size of a voxel along the x or y axis at the center of the imaged volume. The correction due to the sample index of refraction, n_s, can be computed using a ray transfer matrix [56].

2.2.2 Image intensity

In order to obtain an image with good signal to noise ratio, we need to ensure a sufficient amount of light reaches the image sensor. Electronic camera sensors work by converting incoming photons to electrons, which are then amplified and digitized. The number of electrons per pixel read by the sensor is given by $n_e = \eta n_\gamma$, where η is the quantum efficiency ($\eta \approx 0.4 - 0.95$ for modern sensors [57] and n_γ is the number of photons which reach a single pixel in each image exposure. In practice the two relevant sources of noise are technical readout noise and shot noise which comes from the fact that electrons are quantized. The technical readout noise is typically referred to as the dark noise and measured in terms of number of electrons, n_d, and the quantum mechanical shot noise is due to relative fluctations of photons hitting the sensor and has an amplitude of $\sqrt{n_e}$ [57]. Assuming these are the only sources of noise, the signal to noise ratio is given by $\mathrm{SNR} = n_e/\sqrt{n_d^2 + n_e}$. Although n_d can be as low as 1 electron for high sensitivity SCMOS or CCD cameras, it is approximately of order $n_d \approx 10 - 20$ electrons for high-speed cameras [58]. If we assume $\eta = 0.5$ and $n_\gamma = 1000$ photons/pixel, SNR ≈ 20. Note that this is primarily limited by quantum efficiency and shot noise, and so a lower noise camera would produce only marginally better results (a *perfect* camera with $\eta = 1$

and no excess noise would produce SNR ≈ 30). In the following discussion we will use $n_\gamma = 1000$ photons/pixel as the reference level of intensity; in practice this produces satisfactory images for later analysis.

We image samples by seeding them with a fluorescent dye which is excited by the laser sheet. If we wish to produce n_γ photons per pixel, the required energy in a single laser pulse, assuming the small angle approximation, is given by,

$$E_{pulse} = \frac{N_y n_\gamma E_\gamma}{\gamma \eta_{opt}} \left(\frac{4\pi}{\pi \theta^2} \right) = \frac{N_y n_\gamma E_\gamma}{\gamma \eta_{opt}} \left(\frac{4 n_s F}{M} \right)^2, \tag{2.2}$$

where E_{pulse} is the laser pulse energy, $E_\gamma = h_p c/\lambda \approx 4 \times 10^{-19}$ J is the energy of a single photon (h_p is the Planck constant, c is the speed of light, and we have computed the energy for $\lambda = 500$ nm), $\eta_{opt} \approx 0.5$ is the overall optical efficiency of the system (including the quantum yield of the dye and any other optical sources of loss), $\theta = \frac{M}{2 n_s F}$ is the sample half-angle (see fig. 2.3), γ is the extinction coefficient of a single voxel which is fully dyed, and N_y is the image resolution in the vertical direction. The term in parentheses is the fractional solid angle of light collected by the camera lens. For situations where the majority of the sample is dyed, we require $\gamma \ll 1/N_x$ to prevent significant shadowing effects.

Let us consider a reference case, where we are imaging a cubic volume of size $L_x = L_y = L_z = 100$ mm, filled with water ($n_s = 1.33$), and pumped with a 532 nm laser. If we use the diffraction limited resolution ($N_{dl} = 1060$) for the image resolution and have a pixel size of $p = 28$ μm, we obtain $M = 0.297$. The corresponding aperture size per (2.1) is $F \approx 59$ (assuming $n = 1.33$). If we assume an extinction coefficient of $\gamma = 0.1/N_x \approx 10^{-4}$, and a peak imaging brightness of $n_\gamma \approx 1000$, the required pulse energy is ≈ 10 mJ, which is around 100 times more power than is produced by common pulsed lasers with a sufficiently high repetition rate of $\approx 10^4 - 10^5$ Hz [59].

This reference case suggests a practical limitation: when imaging macro-scale objects, the resulting large F-number and small magnification will necessitate a large energy per pulse. In this case, a significant improvement can be made by under-resolving the image, and increasing the camera aperture so that the resolution is limited by the depth-of-field of the lens, as described in section 2.2.1. A

decrease in the resolution will result in bigger spot sizes of the permissible circle of confusion, and will in turn correspond to a larger half-angle of the imaging sample, scaling as $\theta \propto 1/N_z$.

As an example, suppose we reduce the resolution to $N_x = N_y = N_z = 500$. In this case we can decrease the magnification to $M = 0.140$, and increase the aperture size so that $F \approx 15$. Furthermore, let us assume that only a small fraction of the sample is dyed, then we can increase the dye density so that $\gamma = 1/N_x$. Taking into account all of these modifications, the resulting energy required per pulse is $E_{pulse} \approx 65$ μJ. If we are recording volumes at a rate of 60 Hz, this is an average power of 2 W, which is achievable with commonly available lasers.

2.2.3 Volumetric recording rate

The volumetric acquisition rate of our H2C-SVLIF imaging system is limited by the sweeping rate of the galvanometer and the resolution of the high-speed camera. At low resolutions, the volumetric acquisition rate will approach the maximum sweeping rate of the galvanometer motor. For a sawtooth signal the maximum sweeping rate at full scale bandwidth for our galvanometer is 175 Hz. Our volumetric acquisition rates, $\approx$ 60 Hz (see Table 2.2), are the fastest for deep (cubic) volumetric imaging, faster than reported two-color volumetric imaging techniques, e.g. [50, 51].

At high resolutions, the volumetric acquisition rate is limited by the throughput of the high-speed camera. Due to the finite speed of the galvanometer, it will produce a smooth linear ramp during a fraction of the total scan time, known as the duty cycle, η_d. The duty cycle depends on the volume scan rate and the speed at which the galvanometer can reset and stabilize. The camera recording signal is synchronized with the effective scan period to only record during this period. The following equation is used to compute the volume rate of the system when limited by the throughput of the camera,

$$r_v = \frac{r_f \eta_d}{N_c N_z} \tag{2.3}$$

where r_v is the volumetric acquisition rate, r_f is the frame rate of the camera,

and N_z is the resolution in the scanning depth direction, and N_c is the number of channels.

Table 2.2: The maximum volumetric acquisition rate, r_v for H2C-SVLIF assuming an ν_d =85% duty cycle and using the maximum frame rates for a *Phantom* V2512 camera at the specified resolutions.

volumetric spatial resolution	r_f (Hz)	r_v (Hz)
384 × 384 × 384	117,117	129
512 × 512 × 512	75,669	62
640 × 640 × 640	52,974	35
768 × 768 × 768	39,180	21
1024 × 768 × 768	31,815	17
1280 × 800 × 800	25,722	13

2.3 File formats and viewing software

One of the challenges presented by volumetric imaging is the large data sets produced by the method. For example, 10 seconds of 512^3 volumes recorded at 100 vol/s results in a 134 GB of data (assuming 8-bit data storage). This data can be compressed, though for scientific analysis it should be done using a lossless method. Also, ideally the data would be capable of being decompressed fast enough for real-time viewing. This is a difficult constraint: playback at 30 volumes per second would require a decompression speed of 4 GB/s (as measured in the decompressed data).

In a typical experiment a large fraction of the image volume is empty (i.e. containing no dye). In practice, this can range from 50 – 95% of the image voxels. In order to facilitate compression, we clip all pixel values below a certain threshold, chosen to eliminate the dark noise but have minimal impact on the collected data. After this clipping, the image volumes contain large contiguous regions of zeros which can be easily compressed using an LZ4 algorithm, supported by the widely

used VTK data format [60]. Using this data format, we are also able to include perspective correction metadata in these files. This allows for acquired videos, corrected for volume distortions, to be played back at high rates by our open source multi-scale ultra-fast volumetric imaging (MUVI) software [52] on standard desktop or laptop computers. This software is also capable of dynamically displaying other 3D data, including derived flow fields (see for example fig. 2.9).

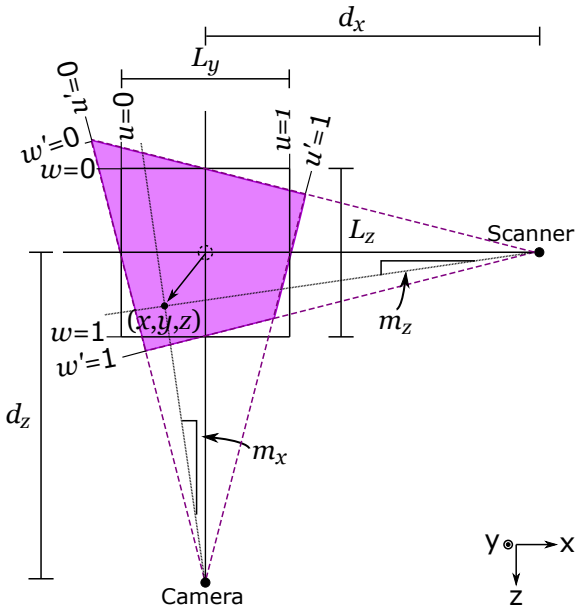


Figure 2.4: Schematic of imaged volume. The purple area indicates the raw distorted volume, the square black line shows the corrected volume. Note that the signs of d_z and d_x depend on the direction of the scan, and on the location of the scanner relative to the camera, respectively.

2.4 Post-processing

The volume imaged by the high-speed camera and scanning system is subject to distortion due to the viewing angle of the camera and the scan as seen in fig. 2.1. Thus, the raw data obtained by our 3D imaging system includes distortion and non-uniform illumination due to the varying angle of the laser sheet and perspective foreshortening from the imaging system. Since we employ the use of prime lenses (*Mikro-Nikkor AF* 105 mm f/2.8 D), distortion effects are negligible

as these prime lenses have extremely low distortion [61]. Although in principle the perspective effects could be eliminated optically (i.e. using a telecentric lens), this significantly complicates the experimental design and, in practice, limits the volume size. Moreover, non-uniformity in the illumination provide additional variations of order 10 – 20%. We have developed methods for correcting the resulting distortion in real-time display of the volumes, and for correcting the non-uniform illumination in the pre-processing. This is enabled through our customized viewing software (MUVI) presented in section 2.3.

By employing the use of a single high-speed camera and imaging directly in 3D, the post-processing task for the reconstruction of volumes is streamlined when compared to tomographic imaging techniques such as TPIV and TPTV. Tomographic imaging techniques necessitate multiple distinct viewing angles typically achieved through camera systems or reflecting mirrors [44, 50]. These techniques result in intricate imaging system designs, complex volume reconstruction algorithms and distortion correction procedures particularly when imaging deeper volumes. As a result, these techniques are often constrained to VOIs which are shallow in at least one dimension, e.g, [13, 62, 63]. In contrast, our technique utilizes a straightforward geometric approach that directly maps raw space coordinates (the position of each voxel in the recorded data) to physical space coordinates. The implementation of single-camera H2C-SVLIF also enables the imaging of complete 3D structures and deep VOIs (see Table 2.4), extending beyond particle imaging. This kind of imaging is not readily achievable with systems reliant on only a few distinct viewing angles or cameras.

2.4.1 Distortion correction

To correct the perspective, we use the following coordinate systems and transformations between them (fig. 2.4):

- **Raw space coordinates:** Each point is assigned a coordinate (u', v', and w') in the range of 0 – 1, based on the relative position of each voxel along each axis in the *raw* recorded data. These voxels will be subject to distortion

due to the scan angle of the laser sheet and the viewing angle of the high-speed camera.

- **Slope space coordinates:** Due to the nature of the imaging system, u', v', and w' correspond to *slopes* (m_x, m_y and m_z) in physical space, rather than displacements. Note that, in the case of the laser scanner this 'slope' actually refers to an angle, but the difference is negligible (less than 1%) if the maximum angle is less than $\approx 10°$. At the maximum scan angle of our scanner (12.5°), voxels will appear $\approx 20\%$ larger at the edges but in practice we maintain the scan angle relatively low ($\approx 8°$) to mitigate this effect. The relationship between raw coordinates and the slopes is given by:

$$u' = \left(\frac{d_z}{L_x} m_x + \frac{1}{2}\right) \tag{2.4}$$

$$v' = \left(\frac{d_z}{L_y} m_y + \frac{1}{2}\right) \tag{2.5}$$

$$w' = \left(\frac{d_x}{L_z} m_z + \frac{1}{2}\right). \tag{2.6}$$

- **Physical space coordinates:** We can compute the slope space coordinates in relation to physical space coordinates (x, y, and z) using the equations:

$$m_x = \frac{x}{d_z - z} \tag{2.7}$$

$$m_y = \frac{y}{d_z - z} \tag{2.8}$$

$$\tan(m_z) \approx m_z = \frac{z}{d_x - x}. \tag{2.9}$$

- **Idealized image coordinates:** It is also useful to define idealized image coordinates (u, v, and w) in an *undistorted* space:

$$x = \left(u - \frac{1}{2}\right) L_x \tag{2.10}$$

$$y = \left(v - \frac{1}{2}\right) L_y \tag{2.11}$$

$$z = \left(w - \frac{1}{2}\right) L_z. \tag{2.12}$$

The limits of these coordinates extend slightly beyond the range 0 to 1 because of the mismatch between the non-rectilinear imaged space and the rectilinear idealized space (i.e. the extent to which the purple box in fig. 2.4 extends beyond the black outline).

From these definitions, we can relate the raw image coordinates to the idealized image coordinates:

$$u' = \frac{u + \epsilon_z}{1 + 2\epsilon_z} \tag{2.13}$$

$$v' = \frac{v + \epsilon_z}{1 + 2\epsilon_z} \tag{2.14}$$

$$w' = \frac{w + \epsilon_x}{1 + 2\epsilon_x} \tag{2.15}$$

where:

$$\epsilon_z \equiv \frac{L_z}{4d_z}(1 - 2w) \tag{2.16}$$

$$\epsilon_x \equiv \frac{L_x}{4d_x}(1 - 2u)\,. \tag{2.17}$$

It is also possible to reverse this transformation:

$$u = \frac{u' + \epsilon'_z(2u' - 1 - 2\epsilon'_x)}{1 - 4\epsilon'_x\epsilon'_z} \tag{2.18}$$

$$v = \frac{v' + \epsilon'_z(2v' - 1 - 2\epsilon'_x)}{1 - 4\epsilon'_x\epsilon'_z} \tag{2.19}$$

$$w = \frac{w' + \epsilon'_x(2w' - 1 - 2\epsilon'_z)}{1 - 4\epsilon'_x\epsilon'_z} \tag{2.20}$$

where:

$$\epsilon'_x \equiv \frac{L_x}{4d_x}(1 - 2u') \tag{2.21}$$

$$\epsilon'_z \equiv \frac{L_z}{4d_z}(1 - 2w') \tag{2.22}$$

Although in principle the volumes could be perspective corrected before viewing and analysis, this would either result in image degradation or require up-scaling of the data. Both are highly undesirable. As a result, we have developed viewing software which corrects the perspective as it is being displayed. This works by converting the raw image coordinates into the idealized image coordinates using

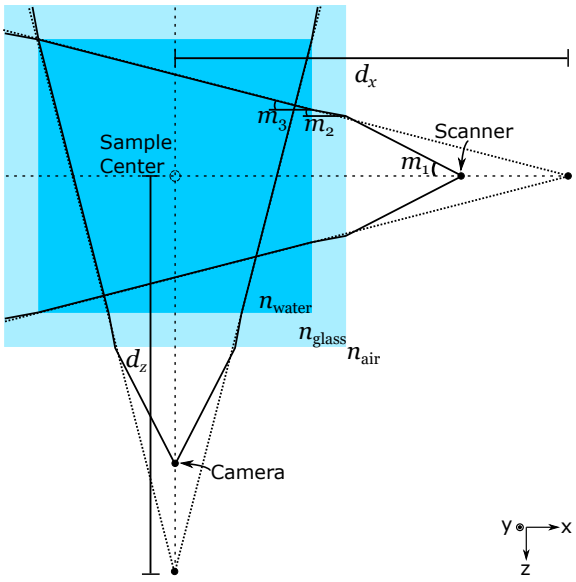


Figure 2.5: Schematic depicting the effective distances, d_x and d_z. The effective distance is given by the distance from the center of sample to where the rays in the sample would converge in the absence of refraction.

the equations above. As the transformation is relatively simple, this can be done at a low level in the display pipeline with minimal impact on performance.

The distortion also needs to be corrected for any features extracted from the raw volume data. For example, consider the problem of particle tracking: if the raw volumes are sent to particle identification software, it will return their positions in the raw image space. These positions can then be converted to the idealized image space after the particle identification. The software library also provides built-in functions for this coordinate transformation.

2.4.2 Effective distance

The distortion correction discussed in the previous section assumes that the imaged volume has the same index of refraction as medium surrounding the scanning head and camera. For typical experiments, however, the imaged volume will be in a fluid, and so the deflection of the light rays by the dielectric interfaces need to be accounted for. In practice, this means that the distances, d_x and d_z, from the sample center to the scanner head and from the sample center to the camera

needs to be replaced with an effective distance, d_{eff}, accounting for the changing index of refraction, see fig. 2.5.

The relationship between the maximum ray slopes, m_i, in different layers (e.g., air, glass, and water) is given by Snell's law, and applying the small angle approximation results in,

$$m_i = \frac{n_j}{n_i} m_j, \tag{2.23}$$

where n_i is the index of each layer. This relationship can be derived from Snell's law assuming the light rays are nearly perpendicular to all surfaces.

Thus, for a total of K interfaces,

$$\frac{L}{2} = \sum_{i=1}^{K} m_i d_i = n_K m_K \sum_{i=1}^{K} \frac{d_i}{n_i}\,, \tag{2.24}$$

and therefore,

$$d_{\text{eff}} = \frac{L}{2m_K} = n_K \sum_{1}^{K} \frac{d_i}{n_i}\,, \tag{2.25}$$

where n_K is the index of the final layer (i.e. $n_K = n_s$ is the index of the imaged volume). This effective distance can be used if layers are measured directly. Note, this correction is applied to both the distance from the scanner to the sample center, d_x, and the distance from the camera to the sample center, d_z, for both the spatial calibration and intensity correction.

2.4.3 Spatial calibration

We have also developed a process for automating the determination of parameters, L_x, L_y, L_z and d_x, d_z, using a calibration target. The determination of these parameters allow us to map the raw distorted space coordinates to physical space coordinates. The target is composed of a regular grid of fluorescent spots, spaced by 5 mm, on a flat sheet of acrylic. The target is created from a sheet of 6 mm thick orange fluorescent acrylic, painted black on one side. A laser engraving system is used to engrave a rectangular grid of 1 mm spots, removing the black paint in these areas and exposing the fluorescent plastic underneath.

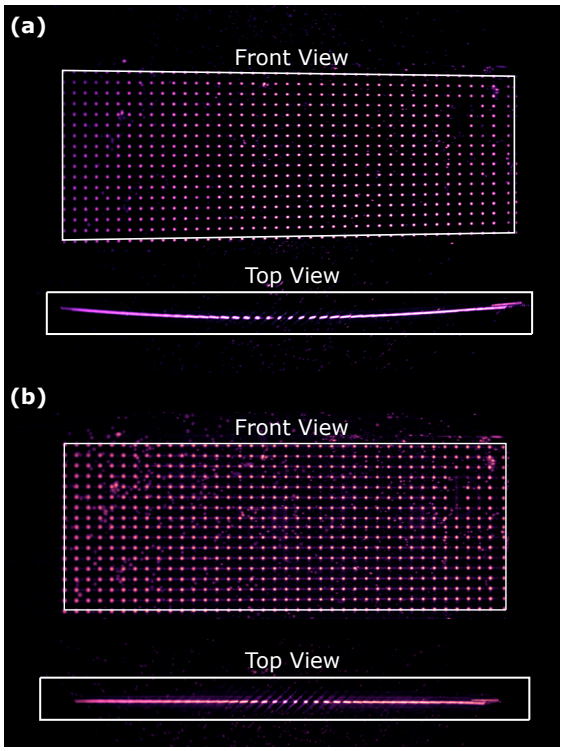


Figure 2.6: Calibration target displayed in MUVI software. (**a**) Raw image data capture by the camera, without correction for perspective. (**b**) The same data after perspective correction.

The distortion correction parameters, L_x, L_y, L_z and d_x, d_z, can be obtained directly from the raw distorted volumes of the grid captured by the camera. The process of obtaining the distortion correction parameters is as follows:

1. Image the target in the sample fluid, placed at an approximately 45° angle with respect to the camera so that the laser scanner can illuminate it entirely while also being fully visible to the camera.

2. Locate the centers of each spot in raw space coordinates using a particle tracking algorithm (see section 2.5).

3. Construct a model which transforms from the positions on the target to the raw space coordinates, including distortion corrections. This model incorporates L_x, L_y, L_z and d_x, d_z as well as arbitrary displacements and rotations of the target (modeled as Euler angles). Eqns. 2.18 - 2.22 return the idealized image coordinates (u, v and w) from the raw distorted space coordinates (u', v', and w'). Then, we transform the idealized image coordinates to physical space coordinates (x, y and z) by applying Eqns. 2.10 - 2.12. Finally, the calibration target is rotated and displaced to lie flat and centered on the Z = 0 plane. The model first takes a 3×3 grid near the center of the volume and fits the grid onto the plane by optimizing the distortion parameters discussed in the next step.

4. Optimize the parameters of this model using the Broyden–Fletcher–Goldfarb–Shanno algorithm [64]. We minimize the following egg-crate fitness function:

$$U = -\cos^2\left(\frac{\pi x'}{s}\right)\cos^2\left(\frac{\pi y'}{s}\right) + \left(\frac{\pi z'}{s}\right)^2, \tag{2.26}$$

where $s = 5$ mm is the known distance between target points, and x', y', and z' are physical space coordinates of the tracked points after applying the transformation model in the previous step. The optimizer returns the distortion parameters, displacements and rotations.

5. The entire image target is calibrated by repeating Step 3 and Step 4 and using initial guesses, returned by Step 4, the approximate distortion parameters, displacements and rotations.

Fig. 2.6 shows an actual captured image of a calibration target before and after the perspective correction. We compute the RMS between the tracked points and the reconstructed points of our model, and find it to be within subvoxel accuracy, RMS ≈ 0.23 voxels.

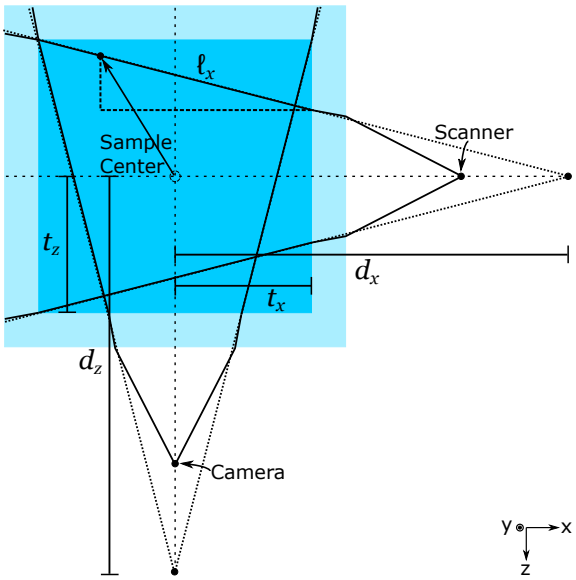


Figure 2.7: Schematic showing the intra-tank distances, ℓ_x, t_x and t_z used for the LIF reference model, eqn. 2.27.

2.4.4 LIF calibration

For many LIF applications it is necessary to produce a signal which is proportional to the local dye density. In the weak excitation limit, the fluorescence intensity of the dye is proportional to the local dye concentration [47], and in turn the digital signal recorded on the camera is also proportional to the fluorescence intensity [65]. However, due to the optical configuration of our laser scanning system, the intensity of the laser sheet systematically varies by 10% or more across the volume. To correct this, we obtain a correction field which compensates for several experimental imperfections using a solution of uniform dye intensity. We image the uniform dye with the 3D scanning system to obtain an intensity field, c_u, measured in units of digital counts on the camera sensor. We then fit this to a reference model:

$$c_r(x,y,z) = b + f_y(s_y) f_z(\ell_z) \frac{d_x}{d_x - x} \exp(-\eta \ell_x) \tag{2.27}$$

$$s_y = \frac{y}{d_x - x} \tag{2.28}$$

$$\ell_x = (t_x - x)\sqrt{1 + \frac{y^2 + z^2}{(d_x + x)^2}} \tag{2.29}$$

$$\ell_z = (t_z - z)\sqrt{1 + \frac{x^2 + y^2}{(d_z + z)^2}}, \tag{2.30}$$

where b is the black level of the camera. f_y is a piece-wise cubic spline correction function in s_y fitted with 20 points. f_z is also a piece-wise cubic spline correction function in l_z fitted with 5 points. ℓ_x and ℓ_z are the lengths of light ray from the inside of the tank to a point in the imaged volume (see fig. 2.7), and s_y corresponds to the slope along the laser sheet (used to correct the non-uniform sheet profile). η is a decay constant with units of 1/m, used to model the absorption of the excitation by the uniform dye.

This complete model is fit using a least squares minimization routine, which obtains optimal values for η and the parameters defining the cubic splines for f_y and f_z. All other values are have been obtained previously using the volume calibration procedure (see section 2.4.3). A corrected digital signal value, c', proportional to the dye intensity is then given by:

$$c'(x,y,z) = [c(x,y,z) - b] \left[f_y(s_y) f_z(\ell_z) \frac{d_x}{d_x - x} \right]^{-1}, \tag{2.31}$$

where c is an experimentally recorded intensity field. Note that the exponential decay of the illumination is not included, as this is only relevant for the uniformly dyed reference sample, c_u. In other words, this neglects dye shadowing effects in experimentally relevant samples [66]. The performance of this correction is limited by camera readout noise, shot noise, and intensity noise of the laser sheet (stated by the manufacturer to be 4% for 355 nm excitation and 3% for 532 nm), each of which produces random fluctuations that cannot be corrected. To reduce the impact of these sources of noise on the reference intensity, c_r, we optimize our correction parameters using a temporal average of 19 volumes, $\bar{c}_u = \sum_{t_i} \frac{c_u}{T}$.

The magnitude of uncorrectable noise can be obtained directly from fluctuations in c_u for different volumes: $\sigma_u^2 = \langle\overline{(\bar{c}_u - c_u)^2}\rangle$, where $\langle\cdot\rangle$ represents a spatial average, e.g, $\langle c\rangle = \sum_{x_i,y_i,z_i} \frac{c}{N_x N_y N_z}$. In practice the fit of c_r will always have some noise; this can be characterized by $\sigma_r^2 = \langle(\bar{c}_u - c_r)^2\rangle$. Experimentally, we obtain $\sigma_r \approx \frac{1}{2}\sigma_u$. Thus, variations in the corrected intensity for experimental data sets are dominated by random fluctuations rather than imperfections in the reference model. In practice, we obtain signal-to-noise ratios of ≈ 17 dB for the UV channel and ≈ 30 dB for the Green channel. Note, for the Green channel, the noise is obtained after initially cropping 10% of the volume from the edges before analysis.

Table 2.3: The reported mean signal, computed noise levels, and signal-to-noise ratio for the two different excitation sources. Reported values are for the experiments described in section 2.6. The SNR for the 355 nm laser was calculated for the regions on the x-axis where $\langle\bar{c_u}\rangle > \frac{1}{4}\max(\bar{c_u})$.

λ (nm)	$\langle\bar{c_u}\rangle$ (counts)	σ_u (counts)	σ_r (counts)	$\mathrm{SNR} = 20\log_{10}\left[\frac{\langle\bar{c_u}\rangle}{\sigma_u}\right]$ (dB)
355	40.4	5.7	2.5	17.0
532	279.2	9.3	4.7	29.5

2.5 Velocity measurements

In order to measure the velocity of the fluid, we use a particle tracking velocimetry (PTV) approach which locates individual tracer particles in the 3D volumes and links them into tracks over time. We then use a resampling algorithm to interpolate these tracks into smooth particle velocities, and a second algorithm to interpolate these discrete velocities onto an arbitrary regular grid, giving an effective velocity resolution on the order of particle spacing. We note that if desired, it would also be possible to perform 3D PIV to obtain a velocity field (for example using OpenPIV; [67]). However, for the example cases presented below, it is possible to resolve individual tracer particles, so a particle tracking method is preferred.

We use an open-source library, *Trackpy* [68] for particle identification and tracking from volume to volume. *Trackpy* is an implementation of the Crocker-Grier algorithm [69], which works by finding local peak intensities in the volumes. After localizing the peak intensities, the algorithm finds the positions of the particles by taking the average position of the pixels for which the particles spans and weights them by the brightness to locate their center of mass. After recovering the positions of the particles, we update the image coordinates of the particles to physical space coordinates using the distortion parameters, as discussed in section 2.4.3. More advanced Lagrangian particle tracking (LPT) methods are discussed in [70] and [71] for high number of particle identification and the corresponding tracking of particles over extensive periods of time.

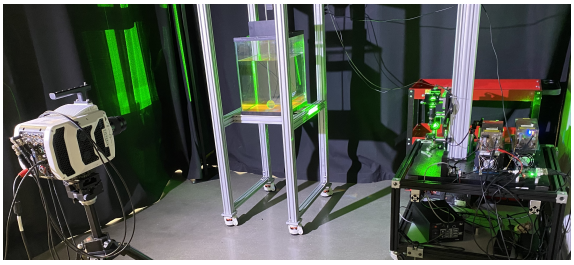


Figure 2.8: A photograph of the experimental setup. In view are the imaging apparatus consisting of a high-speed camera (left), the experiment (center), and the laser sheet generator (right).

Under optimal scenarios the Crocker-Grier algorithm can return particle locations with sub-voxel accuracy, although in practice this is not always the case. Sub-voxel tracking requires that the tracer particles are well resolved, in other words that they span at least several pixels in the recorded image. Tracking a large number of particles would require a much high resolution data set, with a correspondingly lower recording rate. Given the inherent data rate limitations of volumetric imaging with current technology (as opposed to 2D imaging), in most cases it will be preferable to use sub-voxel tracer particles at high density.

To smooth out the resulting trajectories, we implement a modified Savitzky-Golay (SG) filter [72]. An average of $\approx$ 22,000 - 70,000 particles were present in each frame for the experiments outlined in section 2.6. The filter requires a

minimum track length and fits a 15 point window around each measurement to a second order polynomial. After the fit, $\approx$ 20,000 - 55,000 trajectories are yielded, which allows us to resolve ≈ 4 times as many trajectories than in other studies [73, 47], depending on the experimental configuration. The "modified" part of the filter includes the ability to smooth trajectories of particle tracks with a small number of missing frames. This filter returns a new list of particle locations and velocities, giving us a randomly-spaced velocity field measurement of the fluid. The complete implementation of this method is included in the MUVI software [52].

Given the inherent resolution limits of H2C-SVLIF – and the desire to minimize the Stokes number of the tracer particles – it is typically the case that the tracer particles are smaller than an individual voxel. As a result, sub-pixel tracking is typically not feasible, and the resulting discretization of the particle position provides the largest source of noise in the velocity reconstruction. To estimate the noise in the worst case scenario, we implement our SG filter on 5,000 virtual particles moving at assigned random velocities, with positions discretized onto a grid. The error between assigned velocities and the ones computed using the filter was found to have an uncertainty (standard deviation) of $\sigma_t = 0.016\ hr_v$, where h is the voxel size along the specified axis and r_v is the volumetric acquisition rate. The exact Stokes number in our example data in section 2.6 depend on the water temperature and particle size used, but is of order 10^{-3}.

To resample the velocity obtained at our randomly seeded particle locations to a regular grid, we implement a second order windowed polynomial fit in three spatial coordinates [74]. One advantage of this method is one can also compute the strain tensor or vorticity from the derivatives of the polynomial fit [75]. In practice, we select a radial cosine window as the weight function, $w(r) = \frac{1}{2}(1 + \cos\frac{\pi r}{a_0})$, and apply a least squares fit for smoothing [76]. The size of the interrogation window is set to $a_0 = 3\overline{r}$, where $\overline{r}$ is the median particle spacing; $\bar{r} = 3$ – 5 mm. Higher particle concentrations are achievable in order to increase the resolution, however, attenuation effects and shadowing effects due to high particle concentration must be considered. To resolve features in our flow field we seed particles resulting in

concentrations of $\approx 12 - 20$ particles per cm^3 which is ≈ 28 times higher than in the study by [24], for the case of our highest seeded particle concentration. The window size is selected such that, on average, there are 3.6 times more measured particle velocities than fit parameters. If desired, this size can be reduced to improve the spatial resolution of the results at the expense of an increase in noise sensitivity. The noise associated with the resampler, σ_p, will be less than the noise for individual particle trajectories due to the inherent smoothing associated with the interpolation method. The resampler noise depends on the size of the interrogation window and the uncertainty of the trajectories. For an interrogation window of size $a_0 = 3\bar{r}$ the noise was found to be $\sigma_p = 0.005\ hr_v$. This was determined by calculating the standard deviation from a resampled volume containing randomly seeded trajectories of standard deviation σ_t.

To eliminate the effect of spurious trajectories that sometimes result from errors in the linking stage, we first implement the windowed polynomial interpolation scheme on the sparse grid and interpolate the data back onto itself. This allows us to calculate the relative error between the interpolated velocities and the data; we then discard the worst 5% of particles. From the remaining trajectories, we reapply the windowed polynomial interpolation scheme to compute the velocity and vorticity on a regular grid.

Without modification, the effectiveness of this interpolation scheme is diminished near physical boundaries which reduces the number of measured particles in the fit window. In situations where these boundaries are at known locations and speeds – as in the experimental examples described below – this can be mitigated by inserting artificial tracer particles on the boundary which move at the boundary speed. In order to avoid biasing the resampling algorithm, we set the spacing of particles on the surface equal to the mean particle spacing. This scheme significantly improves the quality of the velocity measurements near the boundaries.

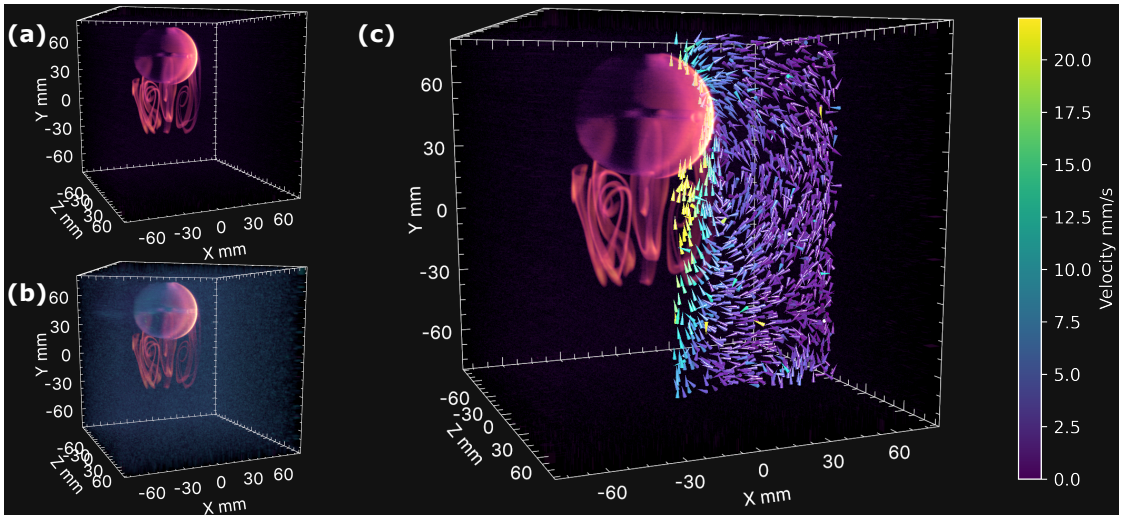


Figure 2.9: Raw and processed data for flow past a sphere, taken at $Re = 400$, visualized in the MUVI software. **(a)** A volume of the UV dye channel, showing the sphere and the fluid shed from the dye injection points which lie along the equator of the sphere. **(b)** An overlay of the 3D tracer particle data (teal, obtained from the green excited dye channel). **(c)** An overlay of some of the velocities measured from tracer particles (purple to yellow arrows).

2.6 Results

To validate our novel H2C-SVLIF technique, we designed a validation experiment consisting of a 50 mm sphere pulled through a tank of water using a linear actuator (see fig. 2.8 and table 2.4) for two different derivatives of Coumarin dye (*Thermo Scientific Chemicals* Coumarin 2 and *Thermo Scientific Chemicals* Coumarin 120). The solute of the dyes is water and the resulting molarity of the solutions are 25 μmol/L – 140 μmol/L. The fluid is randomly seeded with neutrally buoyant tracer particles (*Cospheric* Orange Polyethylene Micro-spheres with 106 – 125 μm diameter); the number density is chosen so that they have an average spacing of $\bar{r} = 3 - 5$ mm. Their fluorescent absorption is well matched to the 532 nm laser (peak absorption: 525 nm, peak emission: 548 nm). A 532 nm dielectric laser notch filter is placed in front of the camera lens to block any light directly reflected from the sphere or other parts of the apparatus (i.e. so that we only image the fluorescent light).

The sphere is accelerated from rest and is suspended vertically by a 9 mm

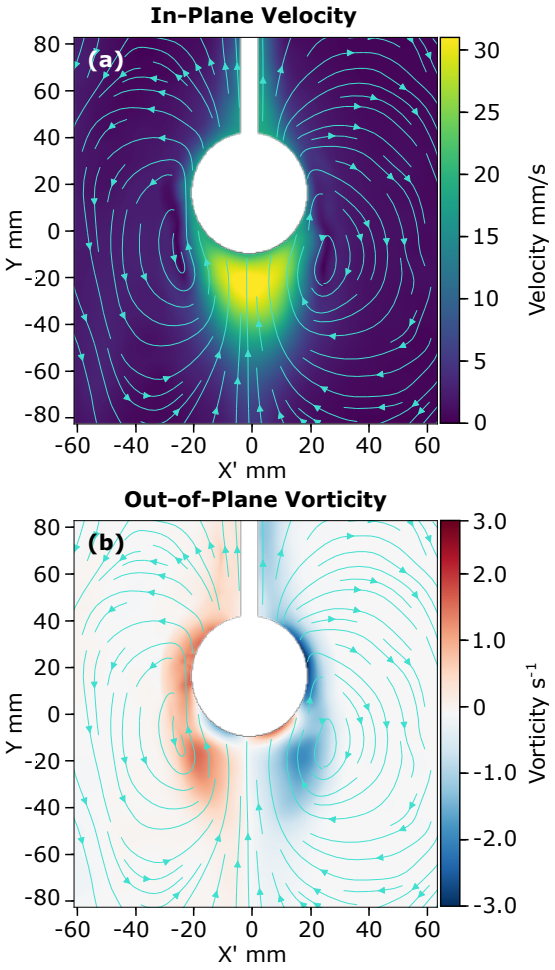


Figure 2.10: **(a)** The in-plane velocity field and **(b)** out-of-plane vorticity field are plotted for a vertical slice of the volume at $L_y = -1.8$ mm, and rotated about the y-axis by -40°. Measurements are taken at $Re = 950$. Streamlines are plotted to reveal flow patterns near the sphere.

diameter stainless steel tube which also acts as a feeding channel for the dye. The dye is then ejected from a system of escape channels that lie along the equator of the sphere. We conduct measurements at $Re = 400$ and 950 for the respective UV dyes. The absorption maxima of the dyes are well matched to the 355 nm laser. Coumarin 2, while commonly available, exhibits lower water solubility compared to Coumarin 120. Additionally, its absorption peak is slightly further from the UV laser pulse, making Coumarin 120 a preferable choice for achieving a better signal-to-noise ratio. We implement the intensity correction method outlined in

section 2.4.4 for datasets acquired using Coumarin 120 as the dye for ejection. Note that to further improve the SNR of the UV excited dyes we use a larger aperture of f/2.8. Although this sacrifices some resolution near the edges of the sample, these regions primarily contain only tracer particles. Fortunately, spreading the intensity of a single tracer particle over several pixels should actually *improve* the particle tracking accuracy [69].

The reconstructed 3D images illustrate dye shedding from the sphere alongside the PTV velocities are shown in fig. 2.9. The intensity shown in fig. 2.9 is corrected as described in section 2.4.4 so one can retrieve quantitative concentration data from the images shown here, if needed. Our study aligns with previous research [11, 77], which predict the emergence of vortex rings within the range of $400 \leq Re \leq 950$. The presence of these rings is visualized through the circulation of dye around them (see to fig. 2.9). Moreover, our analysis identifies a boundary layer near the sphere where vorticity flips sign, and the core of the vortex rings, as evidenced in fig. 2.10 and fig. 2.11. In a similar experimental investigation conducted by [62], these features were absent. They reported an error estimate of 1% to 2% for their reconstructed velocities after applying a 3×3 smoothing Gaussian filter, and report a velocity field of size $32 \times 32 \times 5$. Our investigation yields a comparable noise estimate, utilizing the Savitzky-Golay filter outlined in section 2.5, of approximately 1.4% and a velocity field of size $\approx 40 \times 40 \times 40$ for deep VOIs.

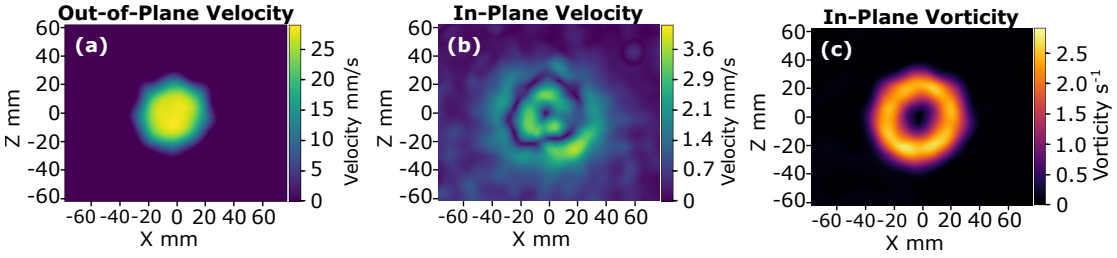


Figure 2.11: **(a)** The out-of-plane velocity field, **(b)** the in-plane-velocity field, and **(c)** the in-plane vorticity field are plotted for a horizontal slice near the bottom of the sphere at $L_y = -18$ mm. Measurements are taken at $Re = 950$. The in-plane velocity field reveals the core of a vortex ring, and the in-plane vorticity field reveals the corresponding vortex ring that is shed from the surface of the sphere.

Values	Definitions
25 – 140 μmol/L	Coumarin 2 and Coumarin 120 dye concentrations
$\bar{r} = 3$ – 5 mm	Mean tracer particle spacing
$\rho = 0.0003$ – 0.0005 particles/voxel	Mean tracer density
14.5 mL/min	Dye injection rate
$u = 7.1$ – 18.8 mm/s	Sphere final velocity
$D = 50$ mm	Sphere diameter
$Re = \frac{uD}{\nu} = 400$ – 950	Reynolds number
$F = 2.8$ (f/2.8)	Lens aperture
$L_x = L_y = 159.3$ – 175.6 mm	Imaged volume size in x and y
$L_z = 143.3$ – 168.5 mm	Imaged volume size in z
$d_x = 1278$ – 1291.5 mm	Effective distance from camera nodal point to center of volume
$d_z = 1445.5$ – 1822.1 mm	Effective distance from scanning-head-platform to center of volume
$r_v = 62.3$ – 65.1 Hz	Volumetric acquisition rate
$N_x \times N_y \times N_z = 512 \times 512 \times 512$	Volumetric spatial resolution

Table 2.4: The parameters used for flow past a sphere and conducting measurements using H2C-SVLIF.

2.7 Conclusion

We have presented a novel H2C-SVLIF imaging technique which permits flexible high-speed volumetric imaging of multiple data channels in a fluid flow. This system is considerably faster than existing multi-channel volumetric imaging techniques; is capable of two channel speeds of up to 65 volumes per second at a resolution of 512 × 512 × 512. We provide a comprehensive discussion of the practical limits of the technique, including laser intensity, sample size/resolution, and camera and scanning speed. We have also described fundamental limits to volumetric imaging, including both diffraction limits to resolution and the effects of shot noise to the SNR of the resulting images. All of these limits should apply not only to our experiments, but to any high-speed volumetric imaging using similar techniques.

Our system also includes open source software and hardware which simplifies and automates much of the data gathering, imaging, and processing. This includes

automated correction of perspective distortion effects using a calibration target, correction of non-uniform illumination, and real time playback and manipulation of the resulting data which incorporates these corrections. By integrating the ability to correct for distortion effects we considerably simplify the experiment setup, which would otherwise require large and complicated optics to produce rectilinear imaging.

The test experiment presented in this chapter demonstrates simultaneous visualization of reconstructed velocity tracks and dyed fluid regions, showcasing the capability to obtain multiple high-speed 3D data streams from a single experiment using two-channel imaging. Building on this imaging technique, we develop a custom built interpolator suitable for noisy experimental data sets. This interpolator is designed to effectively approximate derivatives and smooth out noise, enabling the computation of accurate fields from the experimental data. The detail of this interpolator are outlined in chapter 4, and we implement this interpolation method to study FTLE fields, demonstrating its suitability for processing particle tracking data. In chapter 5, we demonstrate the versatility of the imaging and interpolation method by applying them to analyze bio-inspired flows, specifically visualizing and quantifying the active flows generated by a robotic coral.

Chapter 3

Stretching behavior of knotted vortex generated flow fields

In this chapter, we investigate the stretching mechanics of simulated flow fields through the lens of vorticity. Our goal is to develop a foundational understanding of why and how stretching occurs in flow fields, particularly those that exhibit exponential stretching—a key feature when visualizing FTLE fields [33]. By examining flow fields that generate exponential stretching, we gain insights into the mechanisms underlying deformation and stretching of the fluid domain. We extend this analysis, later on in this thesis, to experimental data collected by our novel imaging system. In chapter 4 and chapter 5, we develop and apply numerical tools to handle noisy data sets and to study stretching mechanics in bio-inspired flows.

There are many examples of flows with concentrated vortex lines, including tornadoes [78], smoke rings [79], flows inside the heart, and turbulence [80, 25]. Indeed any incompressible flow can be regarded as a collection of vortex lines by the Helmholtz Theorem, and so understanding their behavior provides a complete description of such flows. As a result, there has been considerable theoretical, numerical and experimental research focused on understanding the behavior of concentrated vortex filaments (e.g. [81, 82, 27, 26, 83, 84]).

While a simple circular vortex ring is known to be quite stable [28, 29]–even if distorted–knotted and tangled vortices have been observed to be highly unstable in simulations [30, 85, 86] and experiments [87, 88]. In particular, tangled vortex

lines are observed to rapidly stretch, leading to vortex reconnection events which ultimately untie and/or dissipate the flow. This behavior is quite reminiscent of the features of turbulent flows [89, 90], and has both enstrophy production (vortex stretching) and transport of energy to small scales, where is is dissipated (vortex reconnection events).

The connection between vortex stretching and vortex reconnection can be understood in terms of energy conservation: in order to stretch the vortex lines without increasing energy, it is necessary to create regions of closely space counter-rotating vortices [30, 91]. As the vortices continue to stretch, these counter-rotating vortices get closer together, ultimately resulting in reconnections which continue until a non-stretching vortex state is reached [30, 87]. Moreover, vortex stretching (i.e., enstrophy production) is a key feature of turbulence; it can be regarded as the key feature which separates 2D from 3D turbulence [43]. As a result, vortex stretching is intimately connected to the stability of flows, motivating research in to how and why collections of vortex lines self-stretch.

Previous work has modeled this stretching using either vortex filament models [92, 93] or direct simulations of Newtonian super-fluids using the Navier-Stokes [27] or Gross-Pitaevskii equations [94]. Additionally, related work has modeled the stretching of material lines in various flows, including simulated turbulence [95, 96, 97]. In all cases, the non-linearity of fluid flows makes these models difficult to simulate accurately, and complicates the interpretation of the results.

Here we take a different approach: 'freezing' vortex generated flow fields, and investigate how material lines stretch when advected in these flows. Given that vortex lines themselves are transported by the flow [98], it follows that a flow field which stretches material lines will also stretch vortices. In this manuscript we investigate if this stretching can be understood in terms of the properties of the field at a single instant in time, rather than as a consequence of the non-linear evolution of the flow. Using this approach, we find that simple unlinked and unknotted vortices have a flow field which produces linear stretching, while the flow field of linked and knotted vortices produces regions of exponential stretching. Moreover, this stretching can be attributed to the generation of 'bights' in the material lines

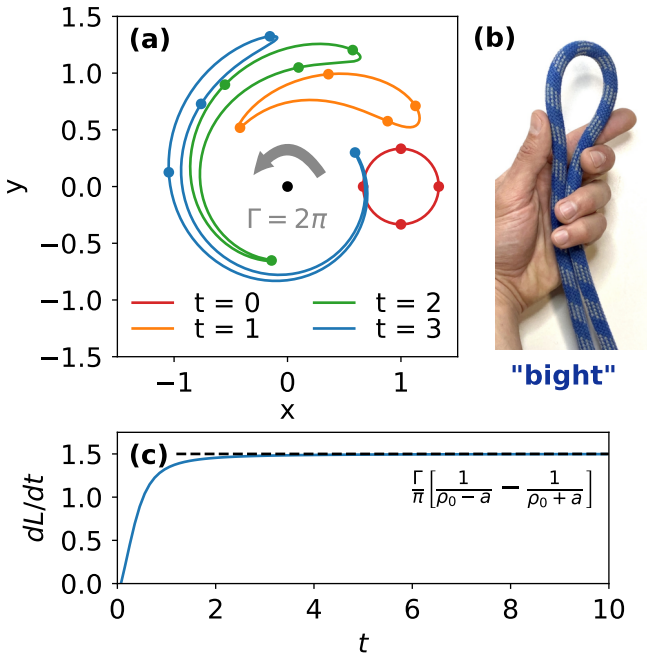


Figure 3.1: **(a)** The time evolution of a circular material loop (initial radius $a = 1/3$ and centered at $\rho_0 = 1$) advected in the flow field of an infinite line vortex with circulation $\Gamma = 2\pi$, oriented along the z-axis. As the initially circular loop evolves, it develops compact 'bights' at the leading and trailing edges. **(b)** A bight in a physical piece of rope, as used in knot tying. **(c)** The stretching rate of the material line, dL/dt. This rate quickly settles to a constant value, which can be predicted using only the velocity field calculated at the locations of the two 'bights' which form as the loop is stretched.

(see fig. 3.1), and that exponential stretching is only possible if these bights are continually produced. This result offers a potential connection to existing results in two-dimensional topological mixing [99], and appears to explain why knotted vortices themselves are unstable. Moreover, qualitatively, and quantitatively similar results have previously been observed in material line stretching in simulated turbulent flows [96, 100]. The concept of bights provides a simple mechanism for understanding how lines in vortex dominated flow fields should stretch over time, with potential applications to a variety of fields.

3.1 Stretching from a line vortex

Let us consider a simple case, the stretching of a material line subject to an infinite vortex which lies along the z-axis:

$$\boldsymbol{u} = \frac{\Gamma}{2\pi\rho}\hat{\boldsymbol{\phi}}, \tag{3.1}$$

where Γ is the circulation, ρ and ϕ are the cylindrical coordinates. The time evolution of small circular line displaced from the vortex core is shown in fig. 3.1. We use the dimensionless spatial and time coordinates, and a dimensionless circulation of $\Gamma = 2\pi$. A material line will only stretch along the direction of the flow field when the tangential component of the material line, $\hat{\boldsymbol{T}}$, is aligned with the flow field.

Consider a small perturbation, $\boldsymbol{\epsilon}$, in a generalized flow field:

$$\boldsymbol{u}(\boldsymbol{r}+\boldsymbol{\epsilon}) = \boldsymbol{u}(\boldsymbol{r}) + \hat{\boldsymbol{T}}\cdot\nabla\otimes\boldsymbol{u}(\boldsymbol{r})\cdot\boldsymbol{\epsilon} + \mathcal{O}(\epsilon^2), \tag{3.2}$$

the stretching is attributed to the tangential component of the strain tensor, $\hat{\boldsymbol{T}}\cdot\boldsymbol{\nabla}\otimes\boldsymbol{u}(\boldsymbol{r})$ (see appendix A for standard tensor notation). Hence the stretching rate of a closed material line, and integrating along the material line, is given by:

$$\dot{L} = \oint \underbrace{\hat{\boldsymbol{T}}\cdot(\boldsymbol{\nabla}\otimes\boldsymbol{u})}_{\frac{\partial\boldsymbol{u}[\boldsymbol{r}(\boldsymbol{s})]}{\partial s}}\cdot\hat{\boldsymbol{T}}\, ds \tag{3.3}$$

$$= -\oint \boldsymbol{u}\cdot\kappa\hat{\boldsymbol{N}}\, ds, \tag{3.4}$$

where $\boldsymbol{r}(s)$ is the material line displacement as a function of arc length coordinate, s. $\hat{\boldsymbol{T}}$, $\hat{\boldsymbol{N}}$, and κ are the Frenet-Serret tangent vector, normal vector, and curvature, respectively, which obey the relationships $\hat{\boldsymbol{T}} = \frac{\partial\boldsymbol{r}}{\partial s}$ and $\kappa\hat{\boldsymbol{N}} = \frac{\partial\hat{\boldsymbol{T}}}{\partial s}$. Eqn. 3.4 is obtained using integration by parts.

Over time, for an infinite line vortex, any section of the material line will tend to align or anti-align itself with $\boldsymbol{u}$ as $t\to\infty$, such that $\boldsymbol{u}\cdot\hat{\boldsymbol{N}} = 0$. To see this, consider a short material line embedded in the flow field given of the infinite line vortex from eqn. 3.1. The transport of a single point subject to the flow field in

eqn. 3.1 is governed by:

$$\frac{\mathrm{d}\phi}{\mathrm{d}t} = \frac{\boldsymbol{v}\cdot\hat{\boldsymbol{\phi}}}{\rho} \tag{3.5}$$

leading to:

$$\phi(t) = \phi(0) + \frac{t}{\rho^2} \tag{3.6}$$

while the other components remain constant ($u_{\hat{\rho}}$ and $u_{\hat{z}}$). If our short material line segment is described by the tangential vector, $\boldsymbol{T}$ (where $|\boldsymbol{T}|$ is identified with the infinitesimal length), we can derive its time evolution:

$$\boldsymbol{T}(t) = T_\rho(0)\hat{\boldsymbol{\rho}} + \left[T_\phi(0) - \frac{2tT_\rho(0)}{\rho(0)^2}\right]\hat{\boldsymbol{\phi}} + T_z(0)\hat{\boldsymbol{z}}. \tag{3.7}$$

These unit vectors are specified at the location of the moving point. From this, we can see that the material line tangent vector will simply increase or decrease in the ϕ direction, depending on the sign of $T_\rho(0)$. Thus, segments pointing out from the z-axis will asymptotically align with $-\boldsymbol{u}$, and segments pointing in will align with $+\boldsymbol{u}$ as $t \to \infty$.

For a closed material line, the material line must change direction with respect to $\boldsymbol{u}$. The material line bends in on itself, resulting in compact regions of 180° bends that we denote as 'bights'. We define a 'bight' at the point at which $\boldsymbol{u}\cdot\hat{\boldsymbol{T}}$ changes sign along a material line. Near the bight, $\boldsymbol{u}\cdot\hat{\boldsymbol{N}} \neq 0$, and so from eqn. 3.4 we would predict that these bights will result in stretching or contraction of the material line, depending on whether they are 'leading' bights ($\hat{\boldsymbol{N}}\cdot\hat{\boldsymbol{u}} < 0$) or 'trailing' bights ($\hat{\boldsymbol{N}}\cdot\hat{\boldsymbol{u}} > 0$).

If we assume that *all* the stretching can be attributed to bights – each of which is assumed to be a compact 180° bend – we obtain a simple expression for the stretching rate:

$$\dot{L}(t\to\infty) \approx -2\sum_{\text{bights}} \hat{\boldsymbol{N}}\cdot\boldsymbol{u}, \tag{3.8}$$

where $\boldsymbol{u}$ and $\hat{\boldsymbol{N}}$ are computed at the bights and an overall factor of 2 is obtained by integrating $\boldsymbol{u}\cdot\kappa\hat{\boldsymbol{N}}$ around the 180° bend.

As an example, consider a material line which is a circle displaced from a single vortex line fig. 3.1. If this circle has center displacement, ρ_0, and radius, a, it will form a leading bight near the location closest to the vortex, at $\rho = \rho_0 - a$, and a trailing bight near the spot furthest from the vortex, at $\rho = \rho_0 + a$. Under the assumption that each bight is a 180° bend in the material line, we predict a stretching rate of:

$$\dot{L}(t \to \infty) = \frac{\Gamma}{\pi}\left[\frac{1}{\rho_0 - a} - \frac{1}{\rho_0 + a}\right]. \tag{3.9}$$

As seen in fig. 3.1**c**, the stretching rate approaches this result after a relatively short period of time.

3.2 Numerical model

It is unclear whether such a formulation should apply to more complicated geometries (e.g., vortex rings and vortex knots), for which we cannot assume that the material lines will always align with $\boldsymbol{u}$. The flow field of more complex vortex shapes is given by the Biot-Savart law:

$$\boldsymbol{u}(\boldsymbol{r}) = \frac{\Gamma}{4\pi}\oint \frac{\hat{\boldsymbol{T}}' \times (\boldsymbol{r} - \boldsymbol{r}')}{|\boldsymbol{r} - \boldsymbol{r}'|^3} ds', \tag{3.10}$$

where $\boldsymbol{r}'(s')$, $\boldsymbol{T}'$, and s' refer to the vortex path (rather than a material line), and we set $\Gamma = 2\pi$. In practice, the vortex path(s) are represented using a piecewise linear approximation to the the vortex path with a total of 100 points, in which case an exact expression can be obtained for the flow field [101]:

$$\boldsymbol{u}(\boldsymbol{r}) = \sum_j \frac{\Gamma}{4\pi}\,\frac{2\epsilon_j}{1-\epsilon_j^2}\,\frac{\hat{\boldsymbol{\Delta}}_j \times \boldsymbol{R}_j}{R_j R_{j+1}} \tag{3.11}$$

$$\boldsymbol{R}_j = \boldsymbol{r'}_{-}\boldsymbol{r}; \tag{3.12}$$

$$R_j \equiv |\boldsymbol{R}_j|; \tag{3.13}$$

$$\boldsymbol{\Delta}_j = \boldsymbol{r'}_j - \boldsymbol{r'}_{j+1}; \tag{3.14}$$

$$\hat{\boldsymbol{\Delta}}_j \equiv \frac{\boldsymbol{\Delta}_j}{|\boldsymbol{\Delta}_j|}; \tag{3.15}$$

$$\epsilon_j = \frac{|\boldsymbol{r'}_j - \boldsymbol{r'}_{j+1}|}{R_j + R_{j+1}} \tag{3.16}$$

where the sum is over all line segments. Because we are considering frozen flow fields, we do not advect the vortex paths in time – as would happen in flows described by the Navier-Stokes equation – but rather treat them as fixed.

The strain rate tensor of this field ($\boldsymbol{\nabla} \otimes \boldsymbol{u}$) can be explicitly computed:

$$\boldsymbol{\nabla}_{\boldsymbol{v}} \otimes \boldsymbol{u}(\boldsymbol{r}) = \sum_j \frac{\Gamma}{4\pi} \frac{2\epsilon_j}{1-\epsilon_j^2} \frac{\hat{\boldsymbol{\Delta}}_j}{R_j R_{j+1}} \times \left[\boldsymbol{v} - \boldsymbol{R}_j \left(\boldsymbol{v} \cdot \left[\frac{\hat{\boldsymbol{R}}_j}{R_j} + \frac{\hat{\boldsymbol{R}}_{j+1}}{R_{j+1}} + \frac{\hat{\boldsymbol{R}}_j + \hat{\boldsymbol{R}}_{j+1}}{R_j R_{j+1}} + \frac{1+\epsilon_j^2}{1-\epsilon_j^2} \right] \right) \right], \tag{3.17}$$

which allows for the advection of infinitesimal vectors attached to each point on the material line. Points are advected in the flow field:

$$\frac{d\boldsymbol{r}_i}{dt} = \boldsymbol{u}(\boldsymbol{r}_i(t)) \tag{3.18}$$

$$\frac{d\boldsymbol{V}_{i,n}}{dt} = \boldsymbol{V}_{i,n} \cdot \boldsymbol{\nabla} \otimes \boldsymbol{u}(\boldsymbol{r}_i(t)). \tag{3.19}$$

The material lines are represented as a series of cubic Bézier curves. The material line is represented as a series of points along the curve, $\boldsymbol{x}_i$, with a single attached tangent vector, $\boldsymbol{V}_i$, which evolve in accordance to eqn. 3.18 and eqn. 3.19. Each point has an attached segment, length, s_i, which is constant in time expecting period resampling of the curve. The cubic Bézier curve is given by:

$$\boldsymbol{r}_i(z) = (1-z)^3 \vec{\mathbf{P}}_0 + 3(1-z)^2 z \vec{\mathbf{P}}_1 + 3(1-z) z^2 \vec{\mathbf{P}}_2 + z^3 \vec{\mathbf{P}}_3. \tag{3.20}$$

where $z \in [0,1]$ is the position along each curved segment. The control points are defined as:

$$\vec{\mathbf{P}}_0 = \boldsymbol{r}_i \tag{3.21}$$

$$\vec{\mathbf{P}}_1 = \boldsymbol{r}_i + \frac{s_i}{3} \boldsymbol{V}_i \tag{3.22}$$

$$\vec{\mathbf{P}}_2 = \boldsymbol{r}_{i+1} - \frac{s_i}{3} \boldsymbol{V}_{i+1} \tag{3.23}$$

$$\vec{\mathbf{P}}_3 = \boldsymbol{r}_{i+1}, \tag{3.24}$$

initially, s_i corresponds to the arc length of the segment, and the tangent vectors, $\boldsymbol{V}_i$, are all unit length. The end points and vectors are numerically integrated

in time using the Dormand-Prince method [102, 103] with an absolute velocity tolerance of 10^{-8}.

As the curve is advected, $\boldsymbol{V}_i$ will change length to compensate for stretching (rather than including it in s_i). This is done to ensure that $\frac{\partial r_i}{\partial s}(z=0) = \boldsymbol{V}_i$ and $\frac{\partial r_i}{\partial s}(z=1) = \boldsymbol{V}_{i+1}$. At regular intervals, the path is resampled: s_i are replaced with the estimated length of each Bézier segment and the tangent vectors are re-normalized. The actual length of each curve is estimated by sampling each curve at 100 points and computing $|\partial r_i/\partial z|$. To predict the stretching rate from bights, we identify the location on the material line where $\hat{\boldsymbol{T}} \cdot \boldsymbol{u}$ changes sign to compute the normal vector from the implicit Bézier curve. This allows us to compare the actual stretching rate from the derivative of the length of the material line to the predicted one (see eqn. 3.8).

Because some sections of the material line will stretch more rapidly than others, new points will be added dynamically to interpolated positions as to ensure a more accurate sampling of the path. An error is computed using:

$$N_i = \max\left(\frac{|\hat{\boldsymbol{V}}_i \times \hat{\boldsymbol{\Delta}}_i| + |\hat{\boldsymbol{V}}_i \times \hat{\boldsymbol{\Delta}}_{i+1}|}{\epsilon_a}, \frac{|\boldsymbol{\Delta}_i|}{\epsilon_\ell}, \frac{|\boldsymbol{\Delta}_i|}{\epsilon_n |\boldsymbol{\Delta}_{sn}|}\right) \tag{3.25}$$

where $\boldsymbol{\Delta}_i = \boldsymbol{x}_{i+1} - \boldsymbol{x}_i$ is the segment displacement vector ($\hat{\boldsymbol{\Delta}}_i$ is the normalized equivalent), $|\hat{\boldsymbol{\Delta}}_{sn}|$ is the length of the shortest neighboring segment, $\epsilon_a = 0.1$ is the angular tolerance parameter, $\epsilon_\ell = 0.1$ is the length tolerance parameter, and $\epsilon_n = 2.5$ is the neighbor length tolerance parameter. The first term ensures that curves are adequately sampled with a minimum of $2\pi/\epsilon_a \approx 63$ points for a circle. The second term ensures that no segments get longer than 0.1 spatial units. The last term forces no segment to be longer than 2.5 times the length of the neighboring segments.

The addition or removal of points happens at the same interval as the resampling described above. If $N_i > 1$ for any segment, new points are inserted between the ends of the segment using the Bezier approximation; the number of inserted points is ceil$[N_i] - 1$. To remove redundant points, we consider pairs of segments with even and odd indices; if the sum of N_i for these neighboring segments is < 0.9, the midpoint is removed. Finally, to prevent numerical precision problems we do

not insert any new points into segments whose length is less than 10^{-8} spatial units. (If this is not enforced, the very sharp tips of distorted bights will have points so close together that double floating point precision is not enough to resolve them.)

3.3 Simulation details

To understand how shape and topology affect the stretching behavior of 3D vortices, we consider several vortex shapes: pairs of circular or distorted rings, and trefoil knots of varying aspect ratio. The vortex path used in the simulations are defined on the surface of a torus; the conversion between torroidal ($R,\ a,\ \phi$ and α, cylindrical ($\rho,\ \phi,$ and z), and cartesian ($x,\ y,$ and z) coordinates is as follows:

$$\rho = R + a\cos\alpha \tag{3.26}$$

$$x = \rho\cos\phi \tag{3.27}$$

$$y = \rho\sin\phi \tag{3.28}$$

$$z = \rho\cos\alpha \tag{3.29}$$

where R/a corresponds to the major/minor radius of the torus, ϕ is the azimuthal angle and α is the polloidal angle. The major radius is set to $R = 1$ so that the aspect ratio is given by a.

The parameters for the vortex knots are given by:

$$\text{Trefoil knot: } \phi \in [0 - 4\pi] \tag{3.30}$$

$$\alpha = \frac{3}{2}\phi \tag{3.31}$$

$$R = 1, \tag{3.32}$$

where $a = r/R$ is the aspect ratio. For the simulations $a = 0.1, 0.2, 0.3, 0.4$. Note that this knotted solution wraps around the torus twice in the azimuthal direction; we desire for our unknotted cases to do the same for a more direct comparison. The three unknotted cases are each composed of two separated vortices which each wrap once a torus.

The parameters for 'stacked', 'twisted' and 'linked' rings are given by:

$$R_{1/2} = 1, \tag{3.33}$$
$$a_{1/2} = 0.3 \tag{3.34}$$
$$\phi_{1/2} = (0 - 2\pi) \tag{3.35}$$
$$\text{Stacked Rings: } \alpha_1 = -\pi/2 \tag{3.36}$$
$$\alpha_2 = \pi/2 \tag{3.37}$$
$$\text{Twisted Rings: } \alpha_1 = \pi \sin \phi_1 \tag{3.38}$$
$$\alpha_2 = \pi \sin \phi_2 + \pi \tag{3.39}$$
$$\text{Linked rings: } \alpha_1 = \phi_1 - \pi/2 \tag{3.40}$$
$$\alpha_2 = \phi_2 + \pi/2. \tag{3.41}$$

The tilted rings are formed by taking a pair of rings in the $z = 0$ plane. The first has a radius of $R_1 = 0.7$ and is rotated by 0.3 radians about the y-axis. The second has a radius of $R_2 = 1.3$ and is rotated by -0.3 radians about the y-axis. Note that in each case we chosen the aspect ratios and separation between vortices to match the $a = 0.3$ torus knot case.

3.4 Results

We investigate the stretching rates of material lines for varying vortex geometries and topologies. As mentioned previously, knotted vortices have been observed to be highly unstable and to rapidly stretch in dynamical models (vortices evolve subject to the flow). This rapid stretching leads to reconnection events and the life of the vortices are short lived. To mitigate this, we freeze the flow field and seed a material line around the vortex. It follows that the stretching rate of the material line (subject to the flow produced by the frozen vortex) is analogous to the stretching experienced by the vortex itself in a dynamical model. In cases where exponential stretching is observed, we use FTLEs to quantify the stretching rate of material lines and compare it to the rate predicted by eqn. 3.8.

3.4.1 Role of geometry and topology in stretching of material lines

To quantify the stretching rate, a material line is initially seeded around the torus. The material line is a perfect cicle oriented either toroidally (wrapping around the major axis and threaded through the center of the torus), or poloidally (wrapping around the minor axis of the torus with radius $r_p = 1.5a$). As the material line is advected by the flow field of the vortex, the material will stretch. The evolving length of the material lines, for each vortex, is shown in fig. 3.2.

The simplest case we consider is a pair of stacked circular vortices which produce linear stretching (see fig. 3.2**(a-b)**). More dramatic cases of stretching are observed for trefoil knots; in each case, the material line stretches at an exponential rate which depends primarily on the aspect ratio. We fit the last 1.5 time units of each simulation to the equation:

$$L(t) = Ae^{kt}, \tag{3.42}$$

where k is the stretching rate. The values of k obtained for each simulation are shown in Table 3.1.

Intermediate stretching results are obtained for pairs of vortex rings which are tilted or twisted around one another, but which are not topologically linked. In these cases, the stretching often appears exponentially for short periods of time, only to eventually settle down into a linear stretching trend. Interestingly, however, sustained exponential growth of the material line is observed when when the rings are linked with one another. This suggests that the long term stretching behavior of the material line is primarily determined by vortex topology.

In all cases, it can be seen that the evolving material line rapidly forms new bights when regions of it pass between vortices. These bights can be used to estimate the stretching rate (see eqn. 3.8 and fig. 3.2**(c-d)**). This provides a good estimate of the stretching rate, although it is not as accurate as the case of the infinite straight line vortex. The slight loss of accuracy can be explained by noting –unlike in the case of an infinite straight line – new bights are continuously produced for the linked and knotted vortices. These newly formed bights are yet

to reach the long time limit of sharp tips.

We find that the key distinction between the unknotted and knotted (or linked) vortices is the ability to produce new bights even in the long time limit. Indeed, for trefoil knots and linked rings, the number of bights is observed to grow exponentially, along with the length of the material line. As for the case of the knotted vortices, the aspect ratio affects the speed of stretching, but not the qualitative appearance of the material line. The increase in rate can be explained by the fact that the time it takes for a point to be transported around the vortices will be much reduced for smaller aspect ratio.

One might expect that the stretching rate could be modified by changing the radius of the initially seeded poloidal material line, r_p. We conducted an additional 5 simulations for a torus knot with aspect ratio $a = 0.3$ and material line radius ranging from $r_p = 0.35 - 0.75$. The stretching behavior is show in fig. 3.3. We observe that the exponential stretching rate for $r_p = 0.35 - 0.65$ is nearly the same, ranging from $k = 2.17 - 2.37$. As for $r_p = 0.75$, we observe no bight production and linear growth of the material line. Evidently, if we are sufficiently far from the vortex the bight production will be suppressed. This can be explained by noting that the material line no longer crosses an unstable manifold, as discussed in the next section.

3.4.2 Finite time Lyapunov exponents

An alternative method to probe for exponential stretching is to compute FTLE fields [104]. FTLE fields characterize the exponential separation rates of neighboring parcels of fluid in the direction of maximum strain, and can be used to identify stable and unstable manifolds of a flow.

We compute the FTLE using the Cauchy deformation tensor; for an initial point, $\boldsymbol{x}(t = 0) = \boldsymbol{X}$, this is given by:

$$C_{ij} = \sum_{\ell=1}^{3} \frac{\partial x_\ell}{\partial X_i} \frac{\partial x_\ell}{\partial X_j}, \tag{3.43}$$

where $\boldsymbol{x}(T)$ is the point $\boldsymbol{X}$ advected to final time T. In practice, the time derivatives are computed by attaching three infinitesimal vectors, $\boldsymbol{V}_n(t)$ to the advected

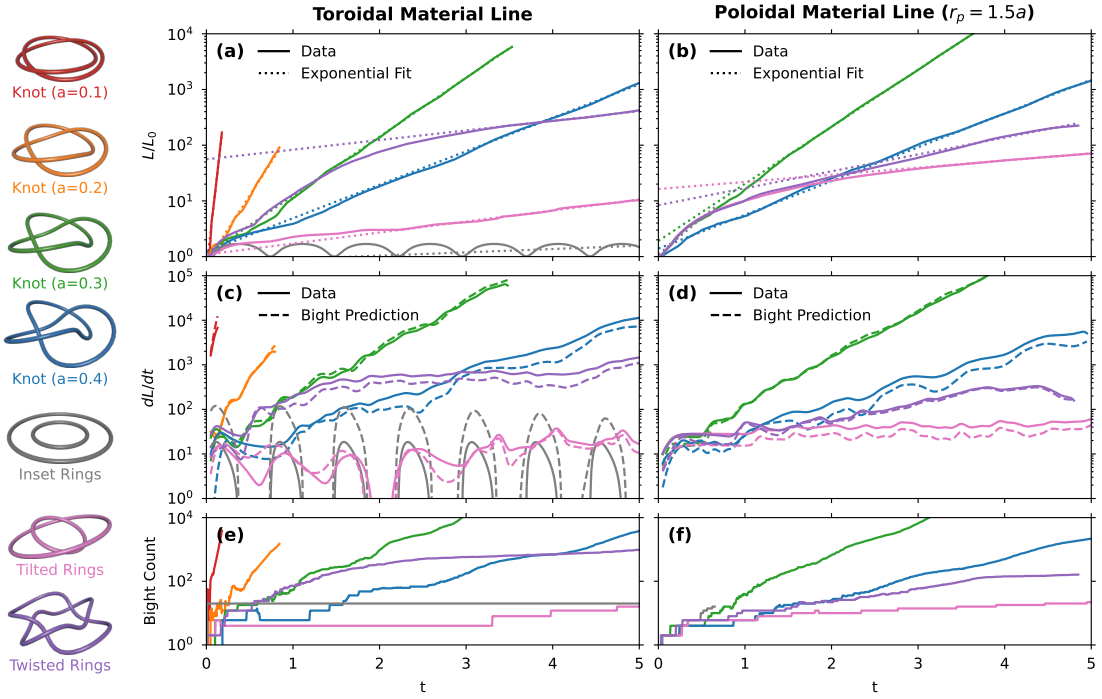


Figure 3.2: The stretching of material lines by vortices of varying shape and topology. For all plots the left plots show initial toroidal material lines, and the right side shows poloidal material lines where $r_p = 1.5a$. **(a-b)** The total length as a function of time. Dotted lines show exponential fits; note that in the cases of unknotted vortices it can be seen that the trend is not exponential, while it is for the trefoil knots. **(c-d)** The stretching rate as a function of time. The solid lines are computed from the numerical derivative of the length, with a Gaussian smoothing of width $\sigma_t = 0.05$ applied to remove high frequency noise. The dashed lines shows an estimate of the stretch rate, obtained from eqn. 3.8. **(e-f)** The bight count, obtained by locating the number of points on the material line where $\boldsymbol{u} \cdot \hat{\boldsymbol{T}}$ changes sign.

Vortex Shape	Aspect Ratio	Tor. Rate	Pol. ML Radius	Pol. Rate
Trefoil Knots	$a = 0.1$	$k_t = 31.5$	$r_p = 0.15$	$k_p = 27.9$
	$a = 0.2$	$k_t = 5.30$	$r_p = 0.3$	$k_p = 5.08$
	$a = 0.3$	$k_t = 2.48$	$r_p = 0.35$	$k_p = 2.16$
			$r_p = 0.45$	$k_p = 2.36$
			$r_p = 0.55$	$k_p = 2.37$
			$r_p = 0.65$	$k_p = 2.25$
			$r_p = 0.75$	$(k_p = 0.237)^\dagger$
	$a = 0.4$	$k_t = 1.41$	$r_p = 0.6$	$k_p = 1.41$

Table 3.1: Measured exponential stretching rates for torus knots. In each case, an exponential stretching rate, k, is determined by a least squares fit of $\ln L(t) = a + kt$. To eliminate initial transients, the fit is over only the last 1.5 time units of each simulation. Fits are shown for initial material lines that are both toroidally (tor.) and poloidally oriented (pol.). In the latter case the radius of the initial material line, r_p is indicated. † indicates data sets which do not fit well to exponential growth curves; see fig. 3.2 and 3.3.

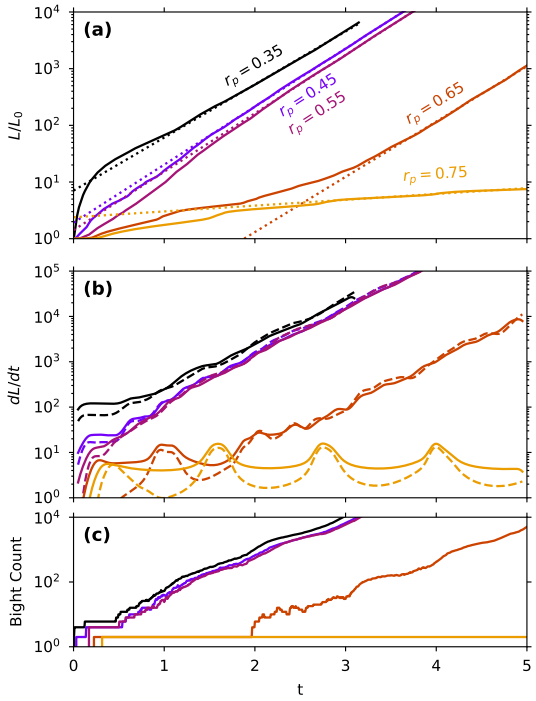


Figure 3.3: Stretching of poloidal material lines of varying radius, r_p for a trefoil knot of aspect ratio $a = 0.3$. For $r_p = 0.75$ new bights are never formed and the material line has a limited stretching rate; in all other cases new bights are formed and exponential stretching is reached with approximately the same growth rate. See 3.4**c** for an overlay of the poloidal material lines with the FTLE values.

point, $\boldsymbol{x}(t)$. Each vector is initialized such that $\boldsymbol{V}_n(t=0) = \hat{\boldsymbol{e}}_n$, where $\hat{\boldsymbol{e}}_n$ are the Cartesian unit vectors. In this representation the strain tensor and Cauchy strain tensor are given by:

$$G_{ij} = \hat{\boldsymbol{e}}_{\boldsymbol{i}} \cdot \boldsymbol{V}_j \tag{3.44}$$

$$C_{ij} = \sum_{\ell} (\boldsymbol{V}_i \cdot \hat{\boldsymbol{e}}_\ell)(\boldsymbol{V}_j \cdot \hat{\boldsymbol{e}}_\ell). \tag{3.45}$$

The FTLE field is then given by:

$$\Lambda_{\text{field}} = \frac{1}{|T|} \ln \sqrt{\lambda_{\max}}, \tag{3.46}$$

where $\lambda_{\max}$ is the largest eigenvalue extracted from the deformation tensor, C_{ij}. The theory for FTLE fields, particularly for particle tracking data, is discussed in more detail in chapter 4.

FTLE fields for knotted vortices of varying aspect ratio are shown in fig. 3.4. In each case, the regions of highest exponential growth appear in regions between the vortices. This is despite the fact that the regions of highest instantaneous shear are immediately adjacent to the vortices; these regions do not produce new bights, and so produce only linear stretching in the long time limit. In practice, the stretching can be so rapid that numerical precision can limit the maximum observed FTLE value. We show that the FTLE values computed for a finite time when the relative stretching of all vectors is less than 10^{10}.

Most notably, the FTLE fields reveal ridges of high stretching – unstable manifolds – which connect sections of the vortex. These can be seen clearly in the center of fig. 3.4**(e-f)**. The narrow regions where the ridges appear, seem to be the regions responsible for producing new bights. We note that if the material line does not cross one of these ridges we do not observe exponential stretching. For example, for $a = 0.3$, $r_p = 0.75$, we observe only linear growth (see fig. 3.3). This material line lies just outside the FTLE ridge. For a slightly smaller value of $r_p = 0.65$ (which *does* cross the ridge) exponential stretching is indeed observed (the location of the material lines are shown in fig. 3.3**(c)**). Finally, we note that the quantitative rate of exponential stretching observed for both toroidal and poloidal material lines has roughly the same time constant as the peak value on the

ridges of high stretching (observed in the FTLE plots); this indicates that these regions are responsible for the exponential stretching.

3.5 Conclusion

We have compared stretching of arbitrary material lines subject to vortical flow fields of varying topology and geometry. In particular we have shown that the flow fields of linked and knotted vortices produce exponential stretching of material lines (in certain regions), which is not seen in the stretching of flow fields of unknotted vortices. This change in the qualitative character is confirmed by computing the corresponding FTLE fields. The FTLE results demonstrate that narrow regions that appear as ridges *between* the vortices are the regions responsible for the exponential stretching. Intuitively, this can be explained by the production of 'bights' in these regions. Conversely, when region of flow is dominated by a single vortex the rapid strain prevents the formation of new bights, ultimately producing less stretching over the long term even though the local strain rate may be much higher.

These results suggest previously unknown connections between previous results in the physics of mixing and vortex dominated fluid flows. In 2D flows, it is well known that the topology of a time-dependent flow field is connected to long term exponential stretching [99, 105]. In this case, exponential stretching is produced by a 'stretching' and 'folding' action reminiscent of a taffy puller. Our work suggests that topology in space alone can play a similar role in 3D flows; in our case the 'folding' of the material lines is indicated by the production of bights. (Note that in the 2D case the folding action will produce new bights.)

Our results may also offer an explanation of the apparent instability of knotted vortices. In both experiments and simulations, linked and knotted vortices have been observed to be highly unstable to self-stretching [30, 85, 86]. Although our results do not model the self-stretching of vortex lines, they do demonstrate that vortex topology has a dramatic effect on the stretching behavior of the flow field.

As previously noted, one can model the stretching produced by vortices whose

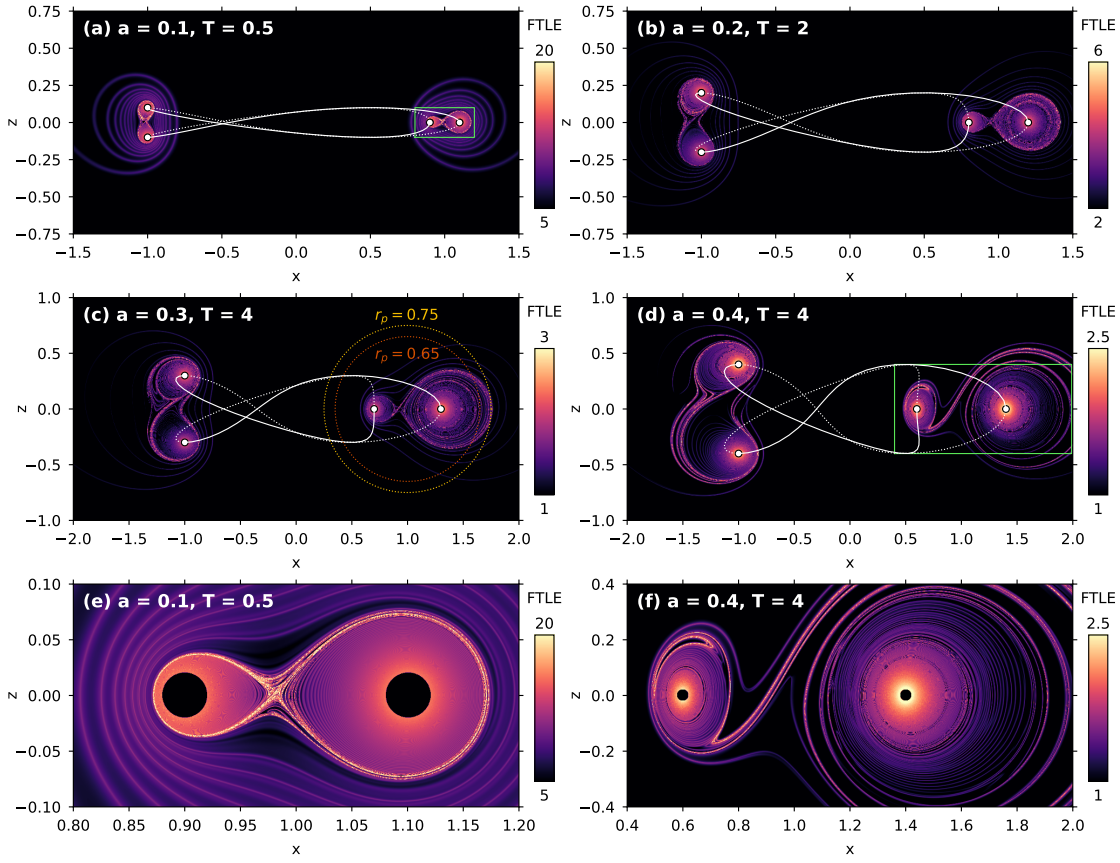


Figure 3.4: A map of finite time Lyapunov exponents, computed for trefoil torus knots of varying aspect ratio. The exponents are computed for a time interval of $T = 30$, and values are shown for a slice in the $y = 0$ plane. The white (dashed) lines show the location of the vortex lines in front of (or behind) the plotted plane. The first four panels **(a-d)** show both sides of the vortex for aspect ratios from a =0.1–0.4, while **(e-f)** shows a zoomed in portion of two cases. (The zoomed in regions are indicated with green boxes in **(a)** and **(d)**) The FTLE isn't computed for points within a radius of $r = 0.02$ of the vortex cores (black circles) to prevent numerical precision issues. Poloidal material line locations for $r_p = 0.65, 0.75$ are indicated in **(c)**. Note that the $r_p = 0.75$ line lies just outside the unstable manifold; as shown in fig. 3.3 this material line has only linear stretching.

shape evolves in time, however, knotted vortices will rapidly approach reconnection events. Thus, a dynamical model is limited in the total amount of time for which the vortices remain knotted, making it difficult or impossible to separate exponential from non-exponential stretching. Interestingly, our results suggest that a more complex model is not required: the stretching dynamics of 'frozen' flow fields may be sufficient to indicate their long term stability. This also suggests that the non-linearities present in the full Navier-Stokes equation are not required to explain the difference in stability between knotted and unknotted vortices. Although future work is needed to show that these results hold for a wider variety of flow configurations –e.g. multiple vortices of different circulation, those with finite core size, etc.

Our work offers clues as to why, when, and where vortex stretching occurs. Stretching arises from the local deformation of the fluid domain, driven by velocity gradients produced by vortex-generated flow fields. The continual formation of bights indicates when significant (in this case, exponential) stretching is expected to occur. We observe that ridges in the FTLE fields correspond to narrow regions where the fluid experiences substantial stretching, particularly near the vortices. In the next chapter, we build on this analysis of FTLE fields and introduce accurate, time-efficient methods for computing these fields for both simulated and experimental datasets.

Chapter 4

FTLE fields for Lagrangian particle tracking data

Finite time Lyapunov exponents (FTLES) measure the separation rate between neighboring particles over a finite time, providing a measure of local deformation and stretching of the fluid domain [33]. Regions of high deformation appear as ridges in FTLE fields which are candidates for Lagrangian coherent structures (LCSs). LCSs are material structures in the flow which act as barriers to fluid transport and are considered stable or unstable manifolds over a finite time. In two-dimensional flows, LCS appear as material lines [106], and for three-dimensional flows they appear as material surfaces [107]. Although ridges may appear in the FTLE field, not all ridges constitute LCSs. This is because LCSs require the rate of strain to be zero along the ridge [31]. Extracting LCSs from the FTLE field poses a further challenge. Nonetheless, by identifying these ridges in the FTLE field we can get insights into mixing and transport mechanisms that govern the fluid domain.

Despite their wide application [32, 108, 109], the visualization of FTLE fields is limited by the high computational cost required to compute these fields. Particularly, the standard method, velocity field integration (VFI), involves advecting either a dense uniform grid or randomly distributed particles to construct a flow map. The flow map indicates the final positions of the grid points or particles based on their initial conditions. VFI is computationally expensive, as it relies on

extensive numerical integration, with accuracy being highly dependent on the time step. The method becomes increasingly costly at high particle densities or for long integration times.

In this chapter, we validate an existing method for FTLE calculation using flow map compilation (FMC) [35], specifically designed for Lagrangian particle tracking data. Unlike the infinitesimal vector approach discussed in section 3.4.2, where vectors are attached to points on a uniform grid to track their evolution and construct the Cauchy strain tensor needed for FTLE field computation, FMC directly utilizes the particle trajectories to construct the flow map and, subsequently, the Cauchy strain tensor. This approach is particularly well-suited for experimental data, as it leverages the particle tracks—directly obtained from experiments—without requiring any integration steps (in contrast to the infinitesimal vector approach and in VFI).

We pair FMC with a custom windowed polynomial (WP) interpolator tailored for Lagrangian particle data sets. We demonstrate that WP interpolation yields accurate interpolated fields even when working with noisy experimental data. Moreover, we highlight, that our custom interpolator outperforms standard radial basis function (RBF) interpolation particularly when computing derivatives of the interpolated fields for both noisy and non-noisy data sets. This work presents a framework for efficient and accurate field interpolation and derivative computation, enabling novel applications in experimental fluid dynamics where precise field measurements are needed. We specifically use this approach for the computation and visualization of FTLE fields.

4.1 Interpolation schemes

The aim of this section is to demonstrate the benefit of using windowed polynomial fitting over standard RBF interpolation when computing smooth derivative fields from Lagrangian particle data. This is particularly of interest because FTLE field computation relies on using gradients of the flow map. Another practical example of where it is necessary to compute smooth accurate derivatives from the

interpolated field is the vorticity field.

As a simple test case in this chapter, we consider the Lamb-Oseen vortex [110], a quasi-3D viscous vortex that has a single nonzero velocity component and a single nonzero vorticity component, for which both depend on the radial distance r from the vortex core. The nonzero components, in cylindrical coordinates, are given by:

$$v_\phi(r) = \frac{\Gamma}{2\pi r}\left[1 - \exp\left(-\frac{r^2}{\sigma^2}\right)\right] \tag{4.1}$$

$$w_z(r) = \frac{\Gamma}{\pi\sigma^2}\exp\left(-\frac{r^2}{\sigma^2}\right) \tag{4.2}$$

where v_ϕ is the velocity component in the azimuthal direction, w_z is the vorticity in the axial direction, Γ is the circulation strength of the vortex and $\sigma = \sqrt{4\nu t}$ is the core size. Cylindrical coordinates are related to Cartesian coordinates via:

$$x = r\cos\phi \tag{4.3}$$

$$y = r\sin\phi \tag{4.4}$$

$$z = z. \tag{4.5}$$

We set $\Gamma = 2\pi$ and fix the core size to $\sigma = 0.5$ and initialize 2000 randomly spaced particles within the domain $x, y, z \in [-3, 3]$ and compute their analytical velocities and analytical vorticities. The velocity field is then interpolated onto a dense grid using RBF interpolation, and the vorticity field is computed from the interpolated field using discrete differentiation as $w_z = \partial_x v_y - \partial_y v_x$.

4.1.1 Radial basis function theory

Radial basis functions (RBFs) are a standard tool for approximations and interpolation in high-dimensional spaces, particularly effective for mesh-free datasets [111, 112, 113]. These functions are radially symmetric, which depend solely on the Euclidean distance, $|\cdot|$, and are rotationally invariant. The construction of an RBF-based approximant starts by defining the source points $\boldsymbol{x}_i$ where $i = 1, \ldots, N$ with corresponding function values $\underline{y} = [y_1, y_2, \ldots y_N]^\top$ and $\boldsymbol{x} \in \mathbb{R}^d$, with d repre-

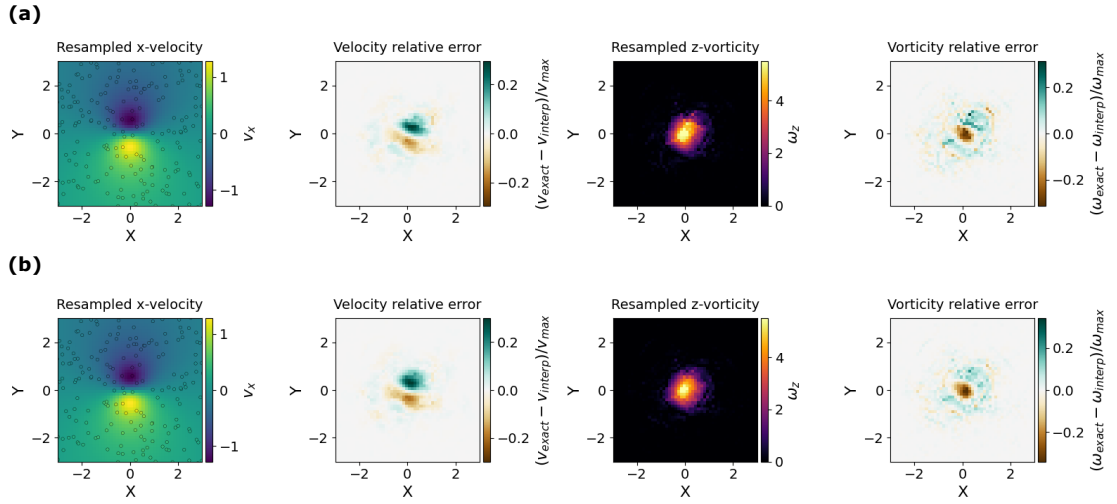


Figure 4.1: RBF interpolated fields using a **(a)** cubic and a **(b)** thin-plate-spline kernel, , with the vorticity computed discretely from the interpolated velocity field. From left to right: the interpolated x-component of the velocity field component, the relative error between the exact and interpolated velocity fields, the interpolated z-component of the vorticity, and the relative error between the exact and interpolated z-component of the vorticity.

senting the number of dimensions. The approximant, $\tilde{\underline{y}}$, is constructed as follows:

$$\tilde{y}(\boldsymbol{x}) = \sum_{i=1}^{N} \beta_i \phi(|\boldsymbol{x} - \boldsymbol{x}_i|) \tag{4.6}$$

where β_i are the unknown coefficients.

The approximant is enforced to exactly match the function values at the source points (e.g., $\tilde{y}(\boldsymbol{x}_j) = y_j$):

$$\sum_{i=1}^{N} \beta_i \phi(|\boldsymbol{x}_j - \boldsymbol{x}_i|) = y_j \quad j = 1, \ldots, N \tag{4.7}$$

which can be written in matrix form:

$$\begin{bmatrix} X_{11} & X_{12} & \ldots & X_{1N} \\ X_{21} & X_{22} & \ldots & X_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \ldots & X_{NN} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_N \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{4.8}$$

where

$$X_{ij} = \phi(|\boldsymbol{x}_j - \boldsymbol{x}_i|) \quad i = 1, \ldots, N \quad \text{and} \quad j = 1, \ldots, N, \tag{4.9}$$

or in compact form:

$$\mathbf{X}\underline{\beta} = \underline{y}. \tag{4.10}$$

A unique solution is guaranteed for the symmetric $N \times N$ design matrix $\mathbf{X}$ if the column vectors of matrix are linearly independent. In practice, it is generally expensive to compute the invertible matrix $\mathbf{X}^{-1}$, to be able to solve for the coefficients via:

$$\underline{\beta} = \mathbf{X}^{-1}\underline{y}. \tag{4.11}$$

From here, we can use the recovered coefficients $\underline{\beta} = [\beta_1, \beta_2, \ldots, \beta_N]^\top$ to interpolate to the arbitrary target point $\boldsymbol{r}$:

$$\tilde{y}(\boldsymbol{r}) = \underline{\beta} \cdot \underline{\phi}(|\boldsymbol{r} - \boldsymbol{x}_i|). \tag{4.12}$$

Common kernels for the RBFs (see fig. 4.2) include:

$$\text{Gaussian: } \phi(x) = e^{-\epsilon x^2} \tag{4.13}$$

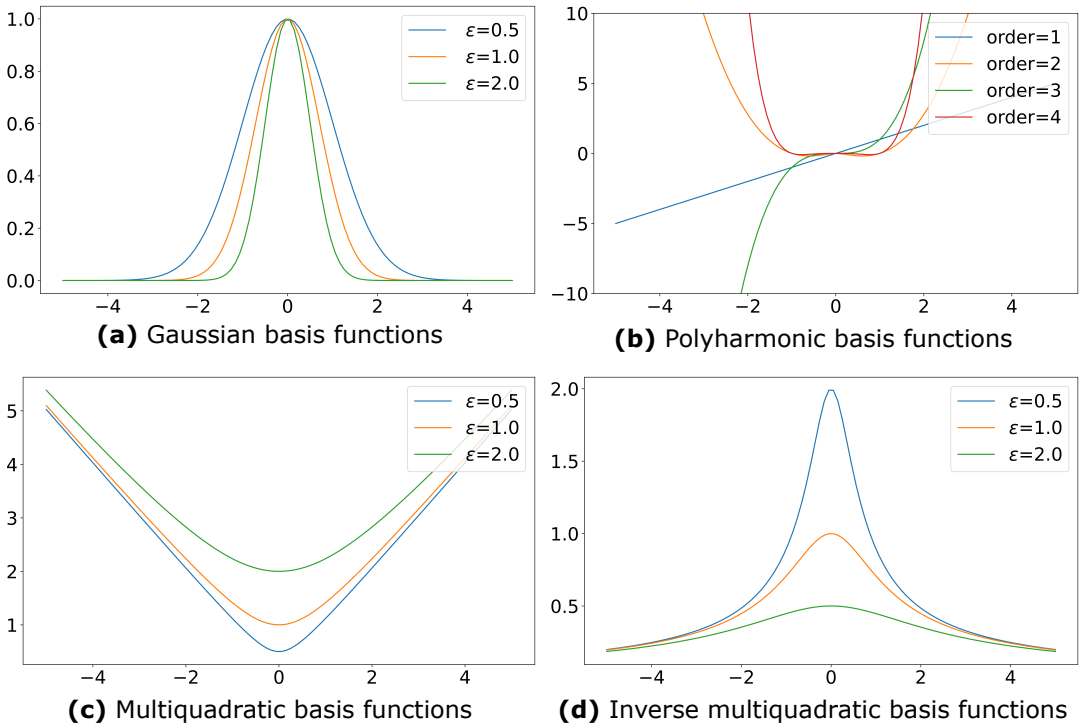
$$\text{Polyharmonic spline: } \phi(x) = \begin{cases} x^p, & p = 1, 3, 5, \ldots \\ x^p \ln(x), & p = 2, 4, 6, \ldots \end{cases} \tag{4.14}$$

$$\text{Multiquadratic: } \phi(x) = \sqrt{x^2 + \epsilon^2} \tag{4.15}$$

$$\text{Inverse multiquadratic: } \phi(x) = \frac{1}{\sqrt{1 + (\epsilon x^2)}} \tag{4.16}$$

where ϵ controls the spread of the RBF.

Fig. 4.1 visualizes the interpolated results using RBF interpolation with cubic (polyharmonic with $p = 3$) and a thin-plate-spline (polyharmonic with $p = 2$) kernels. Although RBF interpolation produces smooth interpolated velocity fields, the resulting vorticity fields–computed using the derivatives of the interpolated velocity–are non-smooth, exhibiting significant noise. This motivates the development of our windowed polynomial (WP) interpolator, which we demonstrate to outperform RBF interpolation in both accuracy, particularly when computing derivative fields from noisy data, and computational speed.



(a) Gaussian basis functions

(b) Polyharmonic basis functions

(c) Multiquadratic basis functions

(d) Inverse multiquadratic basis functions

Figure 4.2: Radial basis functions

4.1.2 Polynomial fitting with Hann window weights and QR theory

The construction of our weighted linear regression model begins by modeling the function using polynomials as our basis functions. For a 2^{nd}-order 2-dimensional function we have:

$$y(\boldsymbol{x}) = \beta_1 + \beta_2 x_1 + \beta_3 x_2 + \beta_4 x_1 x_2 + \beta_5 x_1^2 + \beta_6 x_2^2 + \epsilon \tag{4.17}$$

where x_1 and x_2 represent vector components.

For N-number of source points:

$$y_i = \underline{\beta} \cdot \underline{\phi}(\boldsymbol{x}_i) \quad i = 1, \ldots, N \tag{4.18}$$

$$\tag{4.19}$$

where $\underline{\beta} = [\beta_1, \beta_2, ..., \beta_M]^\top$ containing the unknown polynomial coefficients, $\underline{\phi}(\boldsymbol{x}_i)$ is the vector of polynomial basis functions evaluated at the i^{th} source point $\boldsymbol{x}_i$. Our goal is to find an approximant, $\tilde{y}$, using weighted least squares such that:

$$\tilde{y}_i = \sum_{j=1}^{M} \beta_j \phi(\boldsymbol{x}_i) \tag{4.20}$$

and minimizing the over determined system with $N > M$ (generally there will always be more source points than unknown coefficients) so that we are minimizing the error:

$$\arg\min_{\beta} \sum_{i=1}^{N} w_i |y_i - \tilde{y}_i| \tag{4.21}$$

leading to the normal equations:

$$\mathbf{X}^\top \mathbf{W} \mathbf{X} \underline{\beta} = \mathbf{X}^\top \mathbf{W} \underline{y} \tag{4.22}$$

where:

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1M} \\ X_{21} & X_{22} & \dots & X_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \dots & X_{NM} \end{bmatrix} \qquad \underline{\beta} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix}$$

$$\underline{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} \tag{4.23}$$

where each element X_{ij} is defined as follows:

$$X_{ij} = \phi_{j-1}(\boldsymbol{x}_i) \tag{4.24}$$

and X_{ij} corresponds to the evaluation of the polynomial basis function ϕ_{j-1} at the source point $\boldsymbol{x}_i$. In one dimension, the polynomial basis functions are given by:

$$\phi_0(x) = x^{M-1}, \tag{4.25}$$

$$\vdots \tag{4.26}$$

$$\phi_{M-2}(x) = x \tag{4.27}$$

$$\phi_{M-1}(x) = 1. \tag{4.28}$$

The weight matrix $\mathbf{W}$ is a diagonal matrix where the diagonal elements represent the weights assigned to the source points. Scaled weights, w_i, can be computed as:

$$w_i = \sqrt{W_{ii}}, \tag{4.29}$$

where W_{ii} are the entries on the diagonal of $\mathbf{W}$. This allows for the normal equations to be written in the same form as ordinary least squares:

$$\left(\mathbf{X}'^{\top}\mathbf{X}'\right)\underline{\beta} = \mathbf{X}'^{\top}\underline{y}', \tag{4.30}$$

where we define the scaled matrix and vector as:

$$\mathbf{X}' = \mathbf{W}'\mathbf{X}, \quad \underline{y}' = \mathbf{W}'\underline{y}, \tag{4.31}$$

where $\mathbf{W}'$ is the diagonal matrix:

$$\mathbf{W}' = \begin{bmatrix} w_1 & 0 & \dots & 0 \\ 0 & w_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_N \end{bmatrix}. \tag{4.32}$$

For rectangular matrices with $N > M$ we are guaranteed a unique solution for $\underline{\beta}$ if the columns of the design matrix $\mathbf{X}'$ are orthogonal to each other.

To find a unique solution, we use QR-decomposition and let $\mathbf{X}' = \mathbf{QR}$, making this substitution into eqn. 4.30 leading to:

$$\mathbf{R}\underline{\beta} = \mathbf{Q}^\top \underline{y}' \tag{4.33}$$

where $\mathbf{Q}$ is the matrix $(N \times M)$:

$$\mathbf{Q} = \begin{bmatrix} \hat{q}_1 & \hat{q}_2 & \dots & \hat{q}_M \end{bmatrix} \tag{4.34}$$

where $\hat{q}$ are the orthonormal column vectors of the matrix $\mathbf{X}'$, obtained via Gram-Schmidt process (see appendix B) and given by:

$$\hat{q}_j = \begin{bmatrix} Q_{1j} \\ Q_{2j} \\ \vdots \\ Q_{Nj} \end{bmatrix} \tag{4.35}$$

and $\mathbf{R}$ is the diagonal matrix:

$$\mathbf{R} = \begin{bmatrix} \hat{q}_1 \cdot \underline{X'}_1 & \hat{q}_1 \cdot \underline{X'}_2 & \cdots & \hat{q}_1 \cdot \underline{w} \\ 0 & \hat{q}_2 \cdot \underline{X'}_2 & \cdots & \hat{q}_2 \cdot \underline{w} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{q}_M \cdot \underline{w} \end{bmatrix} \tag{4.36}$$

where $\underline{w}$ is the weight vector and $\underline{X}'_j$ are the column vectors of $\mathbf{X}'$ (the rescaled column vectors of $\mathbf{X}$):

$$\underline{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}, \quad \underline{X}'_j = \begin{bmatrix} w_1 X_{1j} \\ w_2 X_{2j} \\ \vdots \\ w_N X_{2N} \end{bmatrix}. \tag{4.37}$$

We can write eqn. 4.33 as:

$$\begin{bmatrix} \hat{q}_1 \cdot \underline{X'}_1 & \hat{q}_w \cdot \underline{X'}_2 & \cdots & \hat{q}_1 \cdot \underline{w} \\ 0 & \hat{q}_2 \cdot \underline{X'}_2 & \cdots & \hat{q}_2 \cdot \underline{w} \\ 0 & 0 & \ddots & \vdots \\ 0 & 0 & \cdots & \hat{q}_M \cdot \underline{w} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_M \end{bmatrix} = \begin{bmatrix} \hat{q}_1 \cdot \underline{y'} \\ \hat{q}_2 \cdot \underline{y'} \\ \vdots \\ \hat{q}_M \cdot \underline{y'} \end{bmatrix} \tag{4.38}$$

which gives:

$$\beta_M = \frac{\hat{q}_M \cdot \underline{y'}}{(\hat{q}_M \cdot \underline{w})} = \frac{\sum_{i=1}^{N} w_i Q_{iM} y_i}{\sum_{i=1}^{N} w_i Q_{iM}}. \tag{4.39}$$

$$\tag{4.40}$$

It is useful to shift our coordinates and center the polynomial fit locally around our target points, such that $\boldsymbol{x}' = \boldsymbol{x} - \boldsymbol{r}$. From here, we can readily recover the coefficients by evaluating the polynomial basis functions at the origin $\boldsymbol{x}' = 0$ (all column vectors of $\mathbf{R}$ will yield 0 except the last column vector). Our interpolating approximant is then given by:

$$\tilde{y}(\boldsymbol{r}) = \underline{\beta} \cdot \underline{\phi}(\boldsymbol{x}'). \tag{4.41}$$

Similarily, as we did for RBF interpolation, we compute the analytical velocities and vorticies for the Lamb-Oseen vortex. Using a square window and a Hann window for WP interpolation, we resolve the interpolated velocity field, and the resulting vorticity fields using the derivative of the interpolated velocity (see fig. 4.3).

The Hann window is given by:

$$w(\boldsymbol{x}_i) = \frac{1}{2}\left(1 + \cos\left(\frac{\pi|\boldsymbol{x}_i - \boldsymbol{r}|}{a}\right)\right) \tag{4.42}$$

where a is the cutoff radius. The square window can be thought as a uniform weight function within the same cutoff radius (see fig. 4.4). Every point $\boldsymbol{x}_i$ is assigned a weight based off its relative distance from the target point $\boldsymbol{r}$. If the square window is used the influence of distance-based weighting is turned off. In practice, in WP interpolation, only points lying within the cutoff radius are considered.

4.1.3 RBF vs WP Interpolation

WP interpolation with a Hann window generally produces smoother derivative fields compared to using a square window or standard RBF interpolation with different kernels. This smoothness, as seen in fig. 4.3, is due to the Hann window assigning weights based on the proximity of the source points to the target point (assumed to be the center of the window). Source points that are close to the target point are assigned larger weights. This dynamic weighting enables the interpolator to capture local features more accurately. Moreover, points entering or exiting the window have their weights smoothly adjusted, leading to gradual transitions that reduce abrupt changes in the interpolated field and its derivatives.

For datasets with relatively low noise, RBF interpolation outperforms WP interpolation when computing interpolated fields (see fig. 4.5**(a)**). This is because RBF interpolation performs an exact fit, which is advantageous for non-noisy data where smoothing is unnecessary and often results in introducing error. When noise is present, WP interpolation is shown to outperform RBF because of the adaptive weighting, which effectively smooths out noise leading to more accurate interpolated fields and derivative fields. Interestingly, WP interpolation with a square window—which is effectively larger than the Hann window, as the Hann window assigns minimal weight to edge points—appears to be more accurate than the Hann window at relatively low noise levels. In such case, distributing the influence equally across all source points, as seen with the square window and with RBF interpolation, seems to mitigate this effect. Nonetheless, fig. 4.5**(b)** suggests that when computing derivatives of the interpolated fields, WP interpolation with a Hann window outperforms in accuracy both the square window and RBF interpolation, effectively reducing inherent noise from the interpolation process.

4.2 Computing FTLE fields

Typical process of computing the FTLE field starts off by first generating a uniform grid of points and advecting these points. A flow map $\boldsymbol{F}_0^T(\boldsymbol{X})$ is defined on a grid and constructed from initial time $t = 0$ to final time $t = T$. However,

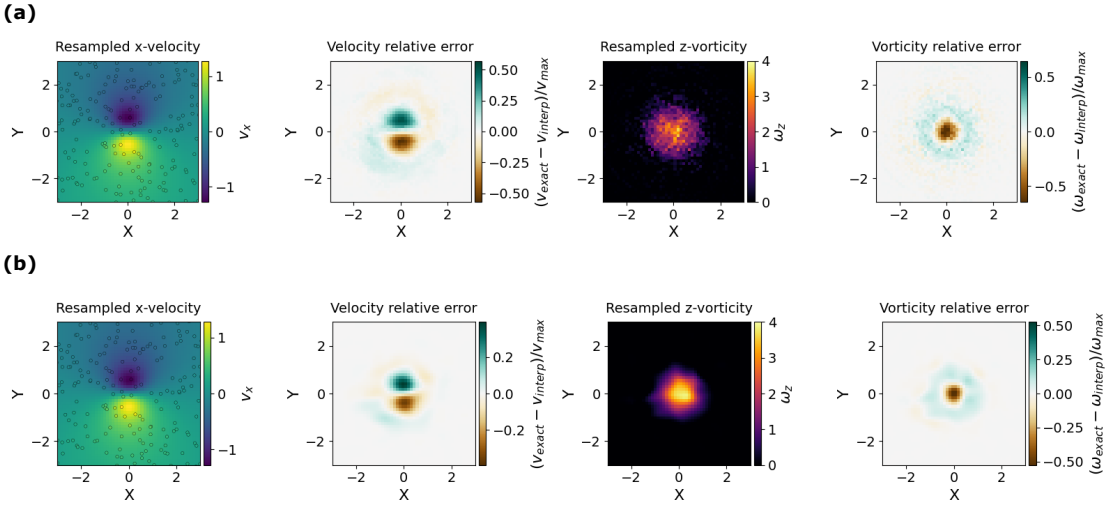


Figure 4.3: WP interpolated fields using a **(a)** square and a **(b)** Hann window, with the vorticity computed discretely from the interpolated velocity field. From left to right: the interpolated x-component of the velocity field component, the relative error between the exact and interpolated velocity fields, the interpolated z-component of the vorticity, and the relative error between the exact and interpolated z-component of the vorticity.

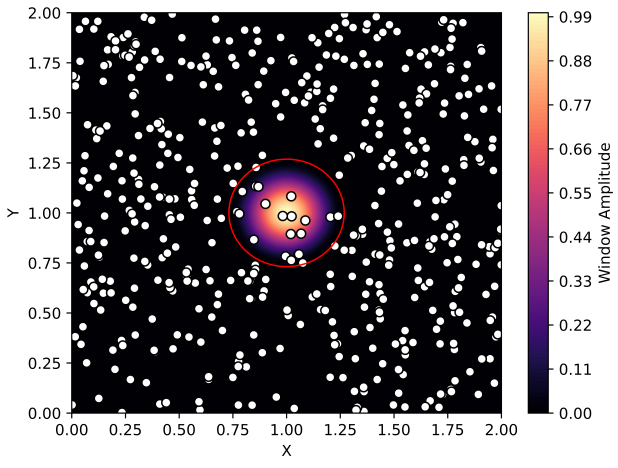


Figure 4.4: Hann window with a hard cutoff radius visualized by the red circle. Source points lying inside the cutoff window are used for WP interpolation. Note, visually, the interpolation target point is assumed to be in the center of the window.

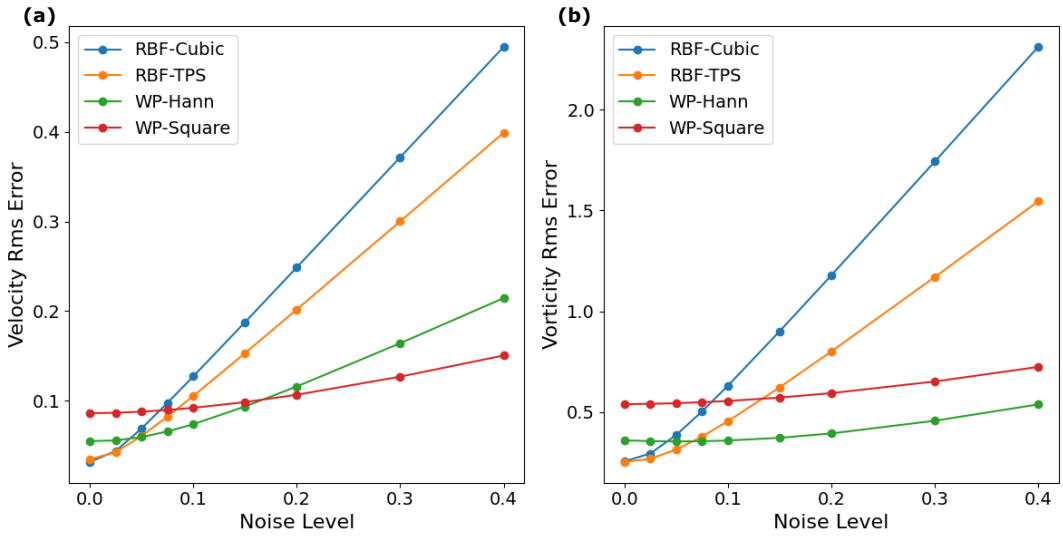


Figure 4.5: RMS error for the Lamb-Oseen vortex across a range of noise levels–(a) for the velocity field and (b) for the vorticity field–computed using RBF interpolation with cubic and thin-plate spline (TPS) kernels, as well as WP interpolation with square and Hann windows.

for noisy Lagrangian particle tracking data, this process is more complicated since particle trajectories are randomly distributed at each frame. Here we will discuss the process of computing FTLE fields via the standard velocity integration (VFI) method and via flow map compilation (FMC). FMC was first proposed by [35] and it eliminates the need for velocity tracks, and the integration step present in VFI (see fig. 4.6).

4.2.1 Velocity field integration

To compute an FTLE field via VFI for randomly distributed Lagrangian particle tracks, consider a velocity field $\boldsymbol{u}$ and a particle trajectory $\boldsymbol{x}(t)$ which satisfies:

$$\dot{\boldsymbol{x}} = \boldsymbol{u}(\boldsymbol{x}, t) \tag{4.43}$$

where $\dot{\square} = \frac{\partial \square}{\partial t}$. The flow map $\boldsymbol{F}_0^T(\boldsymbol{X})$, with $\boldsymbol{X} = \boldsymbol{x}(t = 0)$, is computed by advecting particles in time:

$$\boldsymbol{F}_0^T(\boldsymbol{X}) = \boldsymbol{X} + \int_0^T \boldsymbol{u}(\boldsymbol{x}(t'), t')dt'. \tag{4.44}$$

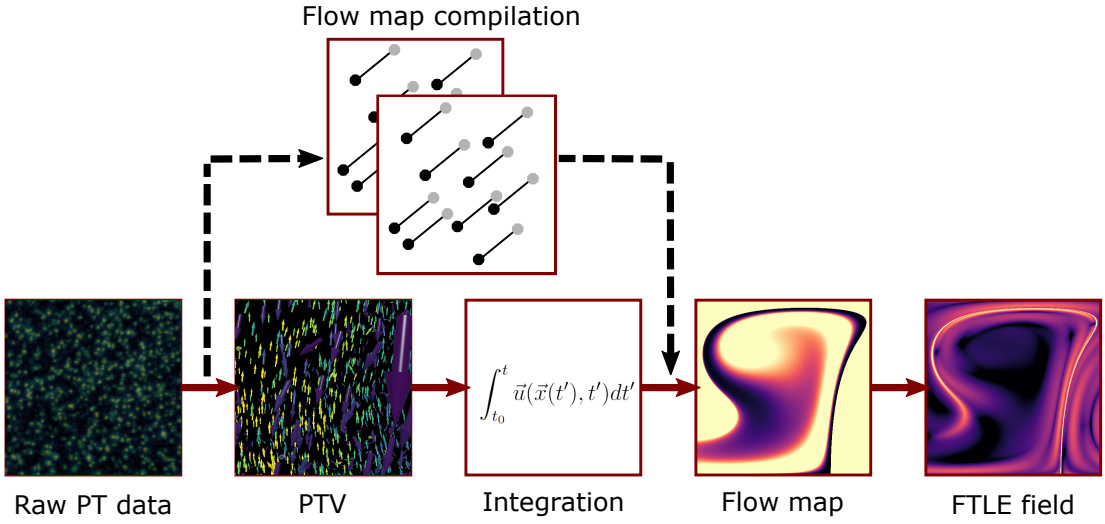


Figure 4.6: Work flow to compute FTLE fields using FMC method. FMC bypasses the need to compute PTV trajectories that would be used for standard VFI method to compile the flow map. Adjacent frames are used to filter particle trajectories to construct the flow map in FMC.

The flow map, however, is defined on a discrete grid, thus $\boldsymbol{F}_0^T$ must be interpolated on an initial dense grid to capture its deformation over time. At each time step, the flow map $\boldsymbol{F}(\boldsymbol{x}(t))$ is interpolated, and the interpolator operator ξ acts on $\boldsymbol{F}$ such that:

$$\xi \boldsymbol{F}(\boldsymbol{x}(t)) : \mathbb{R}^d \to \mathbb{R}^d. \tag{4.45}$$

lies on the uniform grid. As particles are advected forward in time, the interpolated flow map $\xi\boldsymbol{F}$ serves as the updated grid for which the flow map for the next time step. Thus, this results in the effectively smoothly deformed flow map $\boldsymbol{F}_0^T(\boldsymbol{X})$.

We compute the gradient of of the flow map (see appendix A for notation):

$$\mathbf{G}(\boldsymbol{X}, t) = \boldsymbol{\nabla} \otimes \boldsymbol{F} \tag{4.46}$$

$$= \begin{pmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_2}{\partial X_1} & \frac{\partial x_3}{\partial X_1} \\ \frac{\partial x_1}{\partial X_2} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_3}{\partial X_2} \\ \frac{\partial x_1}{\partial X_3} & \frac{\partial x_2}{\partial X_3} & \frac{\partial x_3}{\partial X_3} \end{pmatrix} \tag{4.47}$$

$$\tag{4.48}$$

where X_i denotes the initial coordinates on the uniform grid, and x_i the final coordinates on the deformed grid. This allows us to compute the Cauchy-Green deformation tensor, $\mathbf{C}$, which provides the local stretching rates given:

$$\mathbf{C} = \mathbf{G}^{\top} \cdot \mathbf{G}, \tag{4.49}$$

or equivalently using summation notation as expressed in eqn. 3.43. The largest eigenvalues, $\lambda_{\max}$, which correspond to the maximum separation rate of neighboring trajectories, are extracted from the Cauchy deformation tensor to compute the forward FTLE field, which is given by:

$$\Lambda_{\text{FTLE}} = \frac{1}{|T|} \ln\left(\sqrt{\lambda_{\max}}\right), \tag{4.50}$$

similarly, the backward FTLE field is yielded by using the smallest eigenvalues [34], $\lambda_{\min}$, which measure the convergence of neighboring trajectories.

4.2.2 Flow map compilation

For experimental particle tracking data sets, the particles and their positions are tracked over a finite time, capturing the flow dynamics directly. Thus, we can approximate the flow map by composing intermediate flow maps between adjacent time frames without needing to advect the particles through numerical integration at each time step. The redundant integration step is eliminated and thus provides an efficient way to compute the flow map with only the raw particle trajectories. Similarly to VFI, because our particle trajectories are randomly distributed in the fluid domain, we still have to interpolate onto a continuous grid at each time step as we did in eqn. 4.45.

In practice, because particles can appear and disappear between successive frames, only particles that appear in two adjacent frames are used for each interpolated flow map step. Hence, an approximate flow map is composed with the intermediate flow maps:

$$\boldsymbol{F}_0^T(\boldsymbol{X}) = \xi\boldsymbol{F}_{(k-1)h}^{kh} \circ \cdots \circ \xi\boldsymbol{F}_h^{2h} \circ \boldsymbol{F}_0^h(\boldsymbol{X}) \tag{4.51}$$

where $T = kh$, and the flow map is broken into k smaller time-h flow maps. From here, the FTLE field is computed exactly the same as for VFI using eqns. 4.46 - 4.50.

4.3 The double gyre flow and ABC flow

To test VFI and FMC methods, and the RBF and WP interpolator, we generate the periodic double gyre flow [34]. This field can be thought as two counter rotating vortices confined within the fluid domain. The double gyre is time periodic and lies within the closed bounded domain, $[0, 2] \times [0, 1]$. The stream function:

$$\psi = A \sin(\pi f(x,t)) \sin(\pi y) \tag{4.52}$$

$$f(x,t) = \epsilon \sin(\omega t) x^2 + x - 2\epsilon \sin(\omega t) x \tag{4.53}$$

yields the vector field:

$$\dot{\boldsymbol{x}} = \begin{pmatrix} -\pi A \sin(\pi f(x,t)) \cos(\pi y) \\ \pi A \cos(\pi f(x,t)) \sin(\pi y) \frac{df}{dx} \end{pmatrix} \tag{4.54}$$

we set $A = 0.1$, $\epsilon = 0.25$, and $\omega = \pi/5$.

We initialize a uniform grid of size 512×256 and advect it forward in time using RK4 with fixed time step $\Delta t = 10^{-4}$ to resolve an 'exact' FTLE field (see fig. 4.7). For the interpolated field, we seed 1000 particles in the fluid domain and advect them forward in time with time steps $\Delta t \in [10^{-3}, 1]$, and interpolate the flow map onto a continuous grid of the same size. Depending on whether the VFI or FMC method is used, the interpolation step follows eqn. 4.45 or eqn. 4.51, respectively.

We extend the FTLE field computation for 3D flows using the ABC flow which is aperiodic in time, and has spatially periodic boundary conditions. The vector field of the ABC flow is given by [35]:

$$\dot{\boldsymbol{x}} = \begin{pmatrix} \left(A + \frac{1}{2} t \sin(\pi t)\right) \sin(z) + C \cos(y) \\ B \sin(x) + \left(A + \frac{1}{2} t \sin(\pi t)\right) \cos(z) \\ C \sin(y) + B \cos(x). \end{pmatrix} \tag{4.55}$$

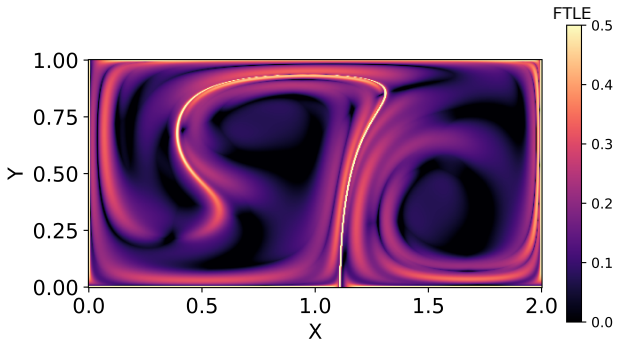


Figure 4.7: The exact FTLE field computed for the double gyre flow reveals a narrow ridge in the center. The ridge represents a coherent structure that acts as a boundary, separating distinct regions in the fluid domain which inhibits mixing.

The FTLE fields are computed on the periodic cube $X, Y, Z \in [0, 1)$, where $x = 2\pi X$, $y = 2\pi Y$, $z = 2\pi Z$. Fig. 4.8 visualize both the corresponding backward and forward FTLE fields.

4.4 Results

Table 4.1: RMS error and computation time for FTLE in the double gyre using the VFI and FMC methods with RBF interpolation and a cubic kernel.

Method	Δt	RMS error	computation time (s)
VFI	1	0.190	21
	0.1	0.118	147
	0.01	0.049	1445
	0.001	0.038	13714
FMC	1	0.037	11

We begin by quantifying the accuracy and computation time for both VFI and FMC methods. Since FMC uses particle trajectories as direct measurements of the flow map, its accuracy is independent of the chosen time step. For VFI, the method relies on numerical integration, making its accuracy highly sensitive to the size of the time step. We test the accuracy of VFI and FMC using RBF interpolation with a cubic kernel with zero noise. The results are illustrated in fig. 4.9 and in fig. 4.10. Qualitatively, VFI is shown to converge to the exact field as the time

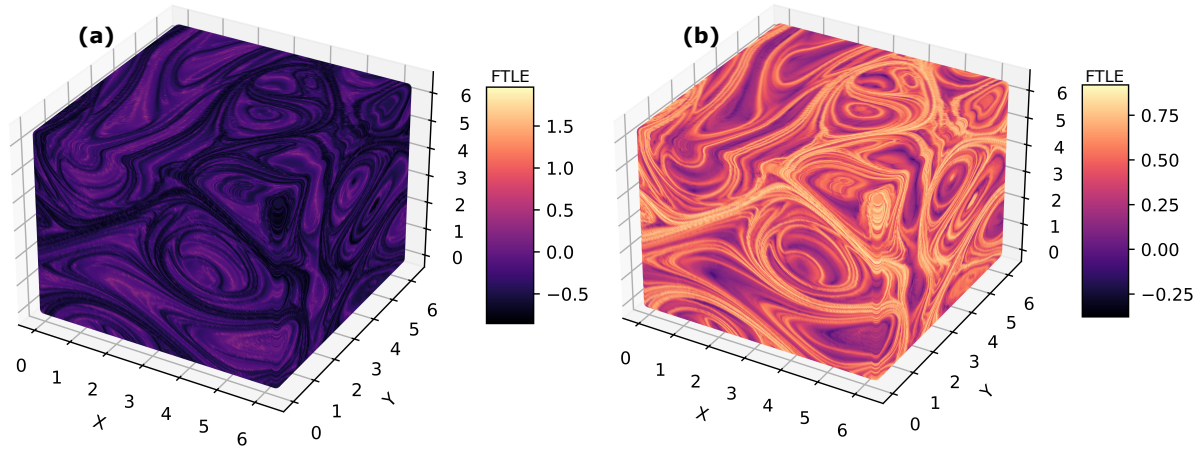


Figure 4.8: Exact **(a)** backward and **(b)** forward FTLE fields computed for the ABC flow.

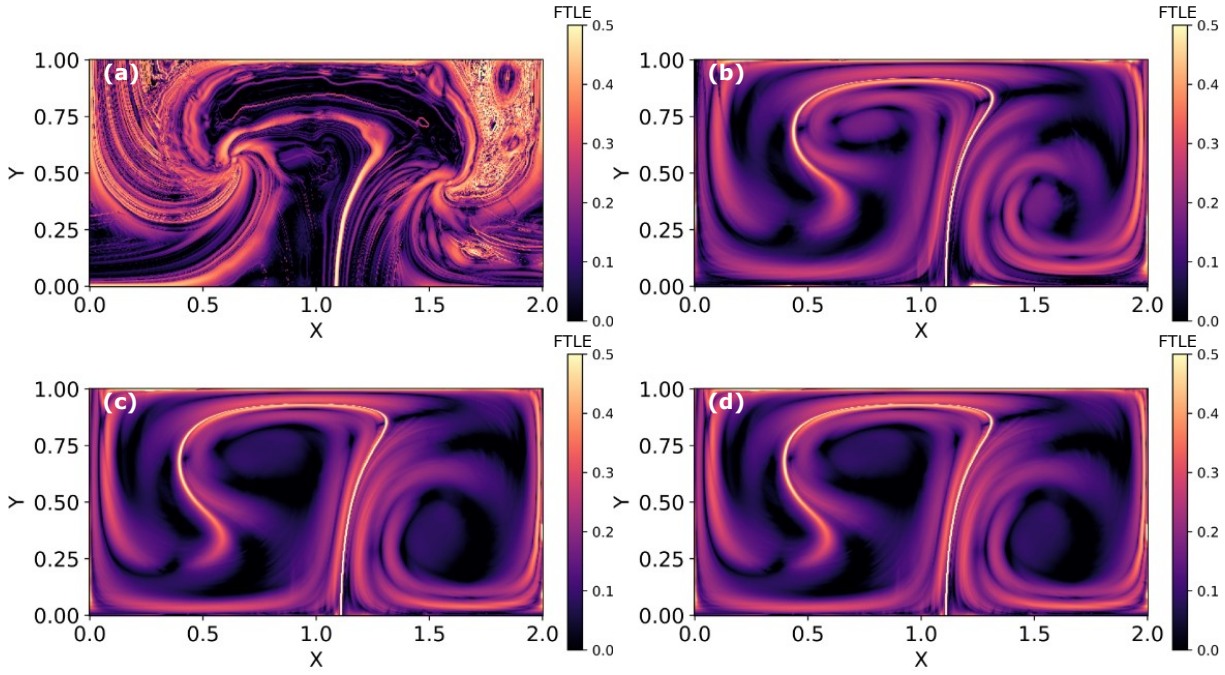


Figure 4.9: The FTLE field computed for the double-gyre flow using RBF interpolation with a cubic kernel for the VFI method at fixed time steps of $\Delta t = 1, 10^{-1}, 10^{-2}, 10^{-3}$, shown in **(a)** – **(d)**, respectively.

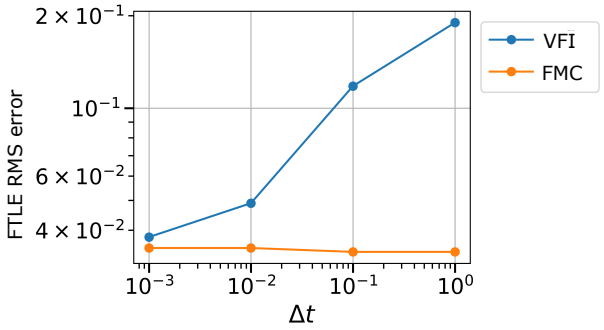


Figure 4.10: FTLE RMS error computed between the exact field and the interpolated field using RBF interpolation with a cubic kernel.

step decreases, while the accuracy of FMC remains unaffected by the time step size. Not shown is the resolved interpolated FTLE field computed using FMC, but it resembles the VFI result in fig. 4.9**(d)**. FMC outperforms standard VFI, achieving a speedup factor of over 1000 for equivalent accuracy (as demonstrated in table 4.1).

To get a qualitative and quantitative understanding on how WP and RBF interpolation behave under noise conditions when computing the interpolated flow map, and derivatives, we add moderately fixed noise to the particle trajectories of 0.51 times of the grid spacing. As seen in fig. 4.11, the FTLE field computed using WP interpolation with a Hann window has less error to the exact field than the one produced by RBF with a cubic kernel. Moreover, WP interpolation is also demonstrated to be 1.6 times faster, when computing the interpolated flow map, compared to the RBF with an execution time of 10 s and 16 s respectively.

This analysis can also be extended to 3D flows, such as the ABC flow illustrated in Fig. 4.8. To compute the 'exact' field, we use VFI with a fixed time step of $\Delta t = 10^{-4}$ and a uniform volumetric grid of size $256 \times 256 \times 256$. The same procedure applied to the double gyre can be used to resolve the interpolated FTLE fields for the ABC flow.

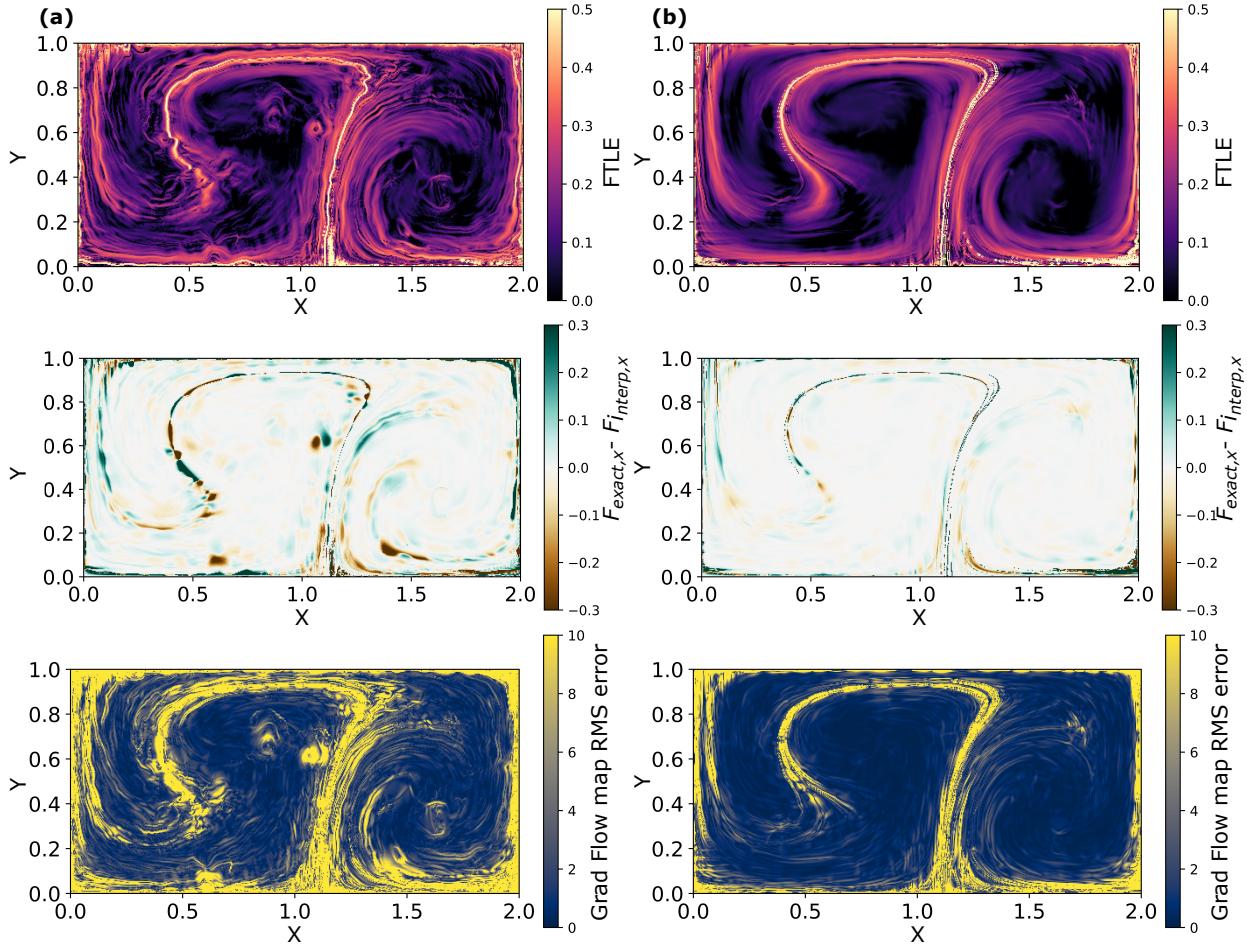


Figure 4.11: **(a)** RBF interpolation with a cubic kernel and **(b)** WP interpolation with a Hann window. From top to bottom: the FTLE field for a noise level of 0.51 of the grid spacing, the relative error between the exact and interpolated x-component of the flow map, and the RMS error between the exact gradient of the flow map and the gradient of the flow map (computed from the interpolated flow map).

4.5 Conclusion

In this chapter we have demonstrated the need for accurate interpolation schemes to compute derivatives of interpolated fields, focusing on two test cases: (1) the Lamb-Oseen vortex and (2) the double gyre. Our custom built windowed polynomial (WP) interpolation is shown to outperform standard radial basis function (RBF) interpolation in both accuracy and speed when computing derivatives of interpolated fields for both noisy and non-noisy data sets. Although, for non-noisy data sets RBFs yield better accuracy when computing solely the interpolated field.

Our results suggest that local features are more accurately captured through a Hann window which dynamically assigns weights based off the proximity of the source points to the interpolation target. However, at low noise levels, this might lead to over-fitting, in which case a square window is preferred. This effectively distributes the noise across all source points equally.

We validate that flow map compilation (FMC) method is the preferred method to compile the flow map used to compute FTLE fields. FMC is demonstrated to not be sensitive to the size of the time step, and is demonstrated to be significantly faster than the velocity field integration (VFI) method because of this. Pairing FMC with WP interpolation using a Hann window yields smooth and accurate FTLE fields making it the most effective approach to compile the flow map and the FTLE field from noisy data sets. This allows us to have an efficient and accurate framework to analyze real-world data in 3D. In the following chapter we will demonstrate the versatility of the tools that we have developed throughout this work to capture and analyze inherently noisy experimental flows.

Chapter 5

Flows generated by a soft robotic pulsing coral

Rising ocean temperatures and increased acidification, driven by the absorption of atmospheric carbon dioxide, have lead to widespread coral bleaching and mortality [114, 115, 116]. Xeniidae, soft tissued corals, are unique among other animals in exhibiting pulsing motion of their tentacles for purposes unrelated to locomotion. Their soft tissued morphology, and perhaps their active motion, may offer ecological advantages in this changing environment, as they have been observed to out compete their stony corals in certain regions [117]. Given their small size, one might think that the energy expenditure of pulsing would outweigh its benefits. It has been hypothesized that these animals pulse their tentacles to generate feeding current, but what has been found is that when analyzing their stomach cavities they are rarely found with food [118]. Their pulsing mechanism, as suggested in a study [119], is now thought to increase the rates of photosynthesis of their symbiotic algae, which provides the animals with most of their energy. Studying the pulsing behavior of Xeniidae corals offers insights in the active flows that they generate and the role that these flows play in their survival and ecological success.

Although prior experimental and computational work have studied the behavior of live Xeniidae corals and simulated models, experimental work has largely been limited to two-dimensional flow measurements [37, 1]. The flows generated by pulsing corals are intrinsically three dimensional, and hence, investigating the

flows generated by these animals require three dimensional volumetric imaging. Since biological fluid dynamics of these organisms pose a challenging problem, a combination of mathematical models, numerical simulations and experiments must be combined to understand the dynamics that govern these animals.

Inspired by Xeniidae corals, we design and fabricate a robot that mimics the pulsing behavior of the corals. To capture the flow dynamics generated by the pulsing, we use our imaging technique H2C-SVLIF and pair it with computational tools developed throughout this thesis to analyze the acquired experimental particle tracking data. Particularly, we use our custom built WP interpolator, presented in section 4.1, to resolve smooth and accurate interpolated fields and derivatives of such fields. We resolve the velocity, vorticity, and FTLE fields at different stages of the robot's pulsing period.

By examining these fields, we can quantify key flow properties resulting from the robot's pulsing motion. The velocity field provides insight into the overall flow structures. Vorticity fields, in turn, reveal regions of concentrated rotational motion, often appearing near specific features such as the tips of the tentacles. Additionally, the FTLE fields quantify the local stretching rate of the fluid domain over a finite time interval. These flow characteristics are critical for understanding how transport and mixing processes operate during the pulsing cycle. Our work provides a framework for future studies by constructing a simplified physical model that replicates key features observed in actual corals, laying the foundation for further exploration.

5.1 Robot fabrication and actuation

The process of collecting 3D flow measurements begins with designing and fabricating a soft robot that mimics the pulsing behavior of Xeniidae corals. This is achieved using an established soft robotic fabrication technique [120] known as fast pneumatic networks. This technique is readily implemented by designing and printing 3D molds. Silicone (*Zhermack* Elite Double 8 and *Platsil* 73-15) is then poured into the molds to cast the robot. The tentacles of the robot are designed

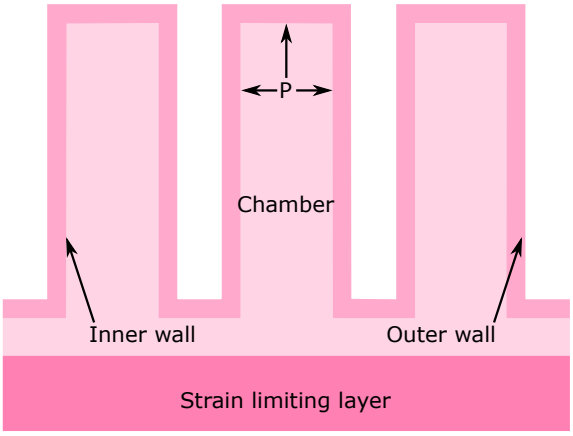


Figure 5.1: Cross-section of the soft robot tentacle using the fast pneumatic fabrication technique. Empty chambers are filled with water through a syringe system. The water pressurizes the inner walls, and the gaps between the outer walls of the chambers causes the robot to flex along the transverse direction. A strain limiting layer is added at the bottom of the tentacle, causing the tentacle to curl.

Figure 5.2: **(a)** Photograph of Xeniidae corals (credit: Anna Dias de Almeida taken at Monterey Bay Aquarium 2024, used with permission) **(b)** downward and **(c)** upward view of robot coral.

with internal chambers that are hollow. Water is injected using a syringe system, causing internal pressurization of the inner walls (see fig. 5.1). Gaps between the outer chamber walls and a strain-limiting layer results in the tentacles elongating and curling. Unlike the corals, the robot is designed with paddle-like shaped tentacles (see fig. 5.2) instead of bristles. This design is chosen to simplify the fabrication process while maintaining functionality. To mimic the pulsing motion, the water injection rate is driven by a sinusoidal signal with an angular frequency of $\omega = 0.051$–0.513 Hz. The tentacle motion is modeled as an oscillation described by:

$$\boldsymbol{X}(r,t) = A\sin(\omega t)\hat{\boldsymbol{e}} \tag{5.1}$$

where A is the amplitude of the oscillation, ω is the angular frequency ($\omega = 2\pi f$) and $\hat{\boldsymbol{e}}$ is the direction of oscillation (along the axis of the tentacle).

5.2 Reynolds number

Xeniidae corals operate within an intermediate Reynolds number regime, approximately $Re \approx 14$. The Reynolds number, Re, is a dimensionless parameter representing the ratio of inertial to viscous forces in a fluid, defined as:

$$Re = \frac{\rho L^2 f}{\mu} \tag{5.2}$$

where L is the characteristic length (tentacle length in this case), f is the pulsing frequency, ρ is the fluid density, and μ is the fluid's dynamic viscosity.

While the tentacles of corals are approximately 0.4 cm in length, our robot's tentacles measure 3.7 cm due to fabrication constraints. To maintain operation within a similar Reynolds number regime, the pulsing frequency of the robot was adjusted accordingly. This ensures that the robot operates within a similar flow regime. Table 5.1 provides a summary of the relevant parameters for the biological corals and the robot prototype. This approach enables us to explore flows in an intermediate Reynolds regime without requiring strict biological accuracy.

Table 5.1: Physical parameters of Xeniidae corals [1] and the robotic coral. Note that ρ and μ are not directly measured but are assumed to be their standard values at room temperature.

Model	L (cm)	$T = 1/f$ (s)	ρ (kg m^{-3})	μ (kg ms^{-1})
Xeniidae	0.41	1.89	1.02×10^3	1.08×10^{-3}
Robot	3.7	12.2	1.00×10^3	1.00×10^{-3}

5.3 Results

Values	Definitions
$\bar{r} = 3.4$ mm	Mean tracer particle spacing
$\rho = 0.0009$ particles/voxel	Mean tracer density
$Re = 100$	Reynolds number
f/5.6	Lens aperture
$L_x = L_y = 178$ mm	Imaged volume size in x and y
$L_z = 156$ mm	Imaged volume size in z
$d_x = 1170$ mm	Effective distance from camera nodal point to center of volume
$d_z = 1300$ mm	Effective distance from scanning-head-platform to center of volume
$r_v = 19.5$ Hz	Volumetric acquisition rate
$N_x \times N_y \times N_z = 512 \times 512 \times 512$	Volumetric spatial resolution

Table 5.2: The parameters used to image flows of the robotic coral using H2C-SVLIF.

We use the H2C-SVLIF imaging technique described in chapter 2 to image the robot and capture the flows generated by its pulsing. To visualize the flow, we seed fluorescent labeled particles (*Cospheric* Fluorescent Polyethylene Micro spheres 106 microns – 125 microns) into the fluid domain. A calibration target is employed and perspective correction is automated as described in section 2.4.3. Although our imaging system is able to capture two different data streams, we use a single channel to capture particle tracking data. The fluid injection is driven by a sinusoidal signal with an angular frequency of $\omega = 0.051$–0.513. Table 5.2 summarizes the experimental parameters for this investigation. Figure 5.3 visualizes the raw data of the robot at various stages of its pulsing cycle.

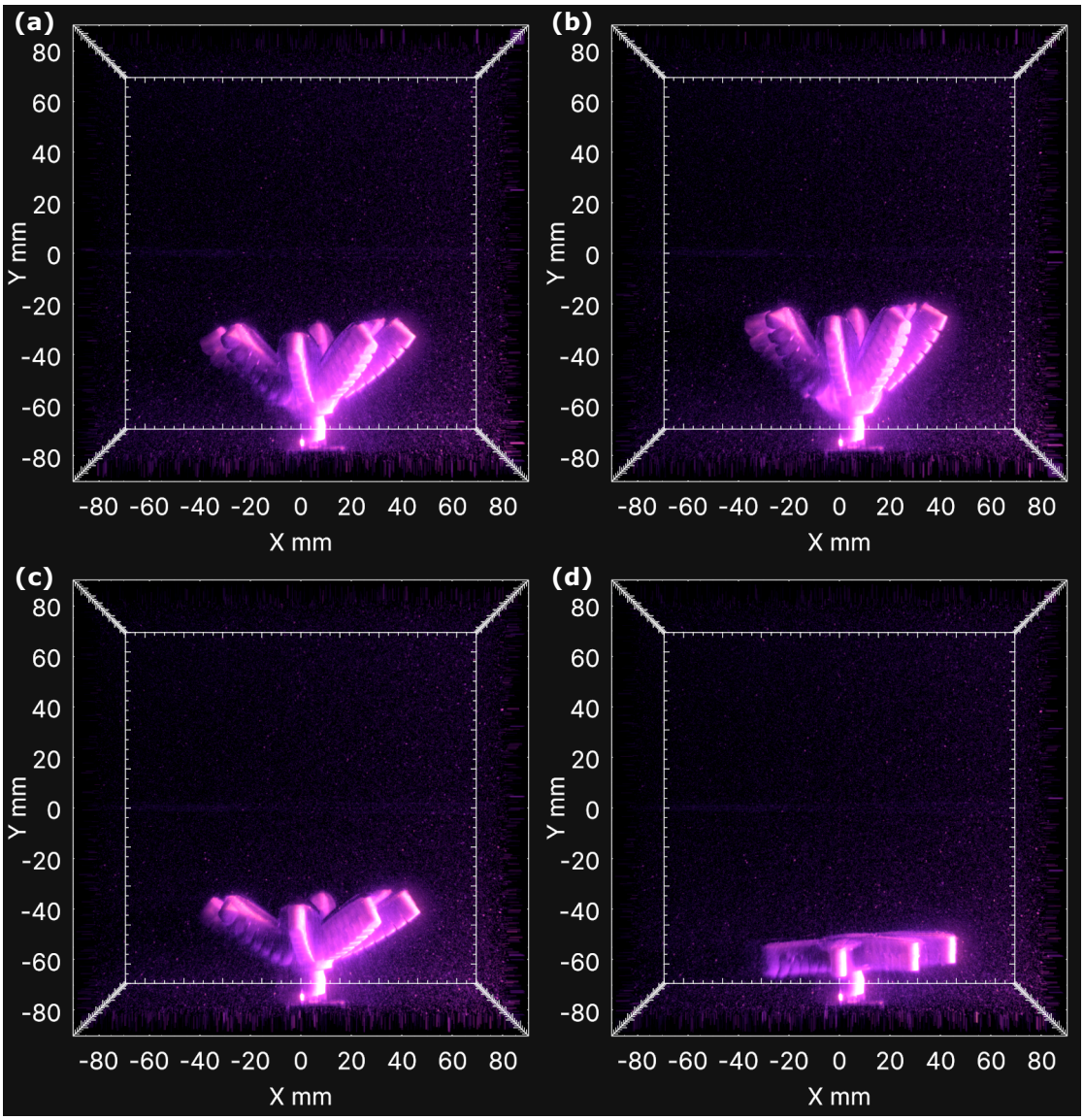


Figure 5.3: Raw data of the robot at different stages at its pulsing period **(a)** taken at $t = 0.25T$, **(b)** $t = 0.50T$ **(c)**, $t = 0.75T$ and **(d)** $t = T$.

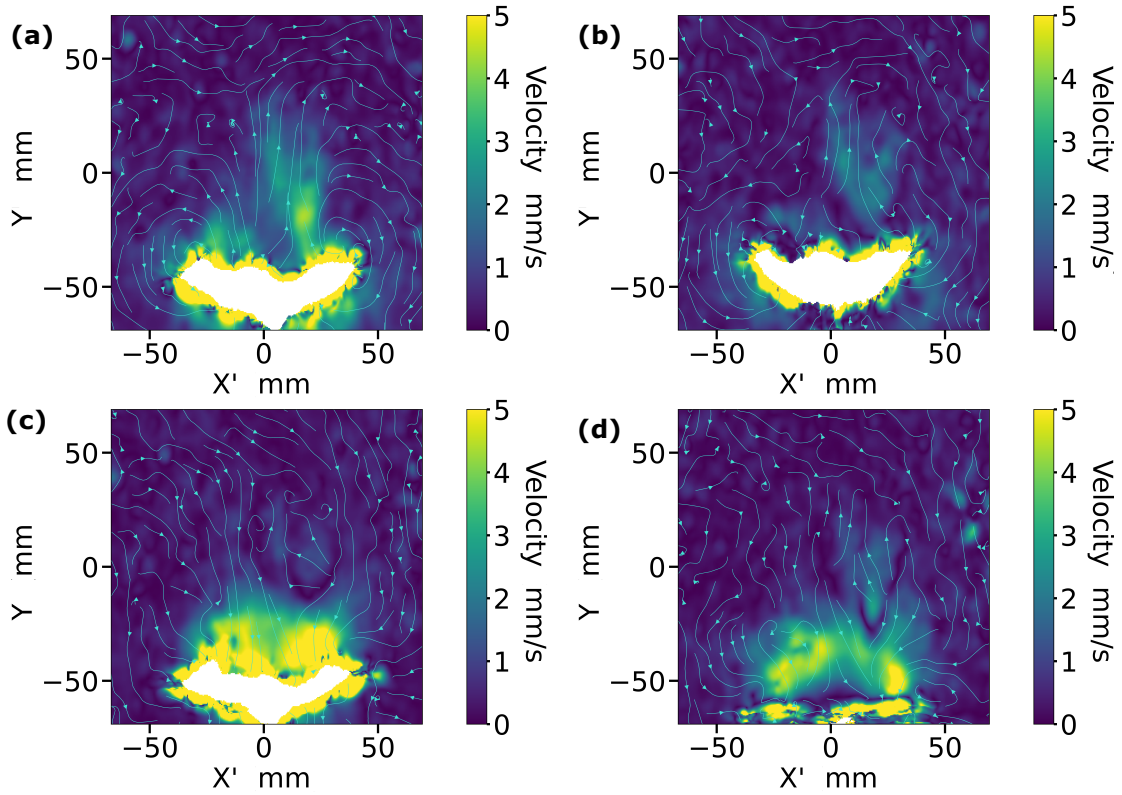


Figure 5.4: A vertical slice at $L_z = 2$ mm, rotated 40° about the y-axis, is taken throughout the pulsing period ($t = T$) for a corresponding Reynolds number of $Re = 100$. The results show the planar velocity magnitude overlaid with streamlines (turquoise color) **(a)** taken at $t = 0.25T$, **(b)** $t = 0.50T$ **(c)**, $t = 0.75T$ and **(d)** $t = T$. Back flow is observed near the end of the pulsing period.

5.3.1 Velocity and vorticity fields

We use our custom-built windowed polynomial (WP) interpolator (see Chapter 4) to reconstruct a dense velocity field from particle tracking velocity trajectories. Additionally, we compute the analytical derivatives of the velocity field to obtain the vorticity field, as detailed in Appendix B.

The presence of an upward jet is observed throughout the robot's contraction phase as shown in fig. 5.4. We plot the planar velocity magnitude and the planar vorticity magnitude for a vertical slice at $L_z = 2$ mm (through the center of the robot) rotated 40° about the y-axis. Significant backflow is observed during the expansion phase, as denoted by the streamlines. This suggests, that the paddle-like tentacles or the kinematics of the robot results in poor fluid advection.

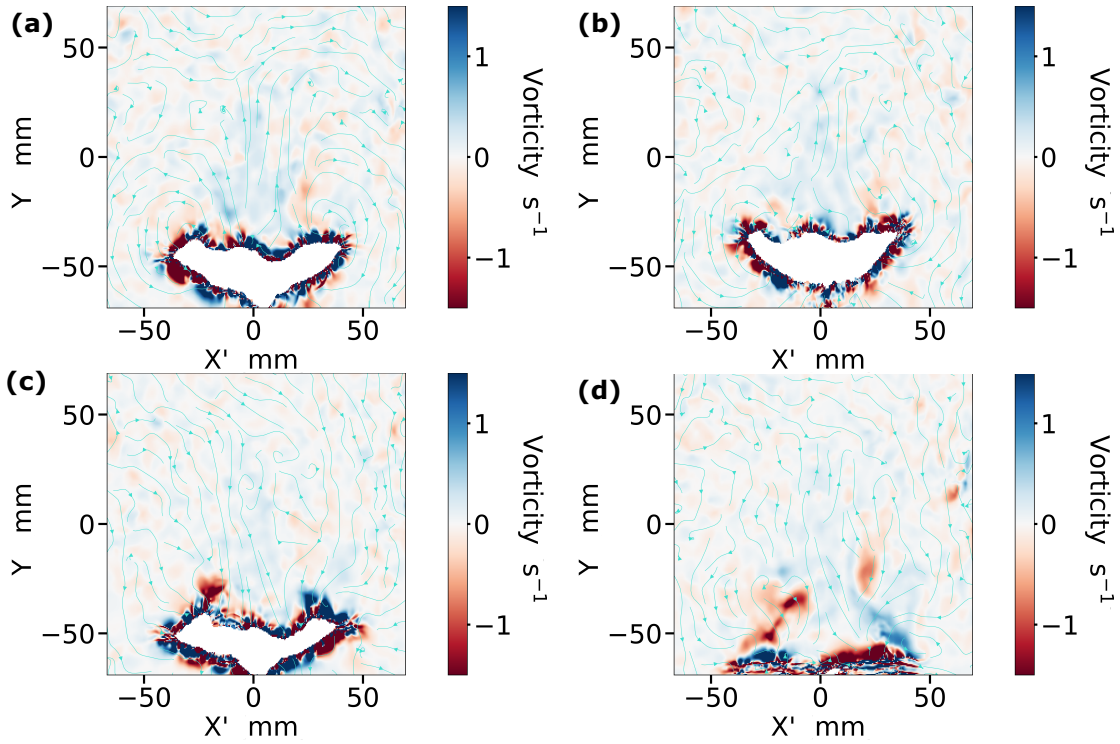


Figure 5.5: A vertical slice at $L_z = 2$ mm, rotated 40° about the y-axis, is taken throughout the pulsing period for a corresponding Reynolds number of $Re = 100$. The results show z-vorticity component overlaid with streamlines (turquoise color) **(a)** taken at $t = 0.25T$, **(b)** $t = 0.50T$ **(c)**, $t = 0.75T$ and **(d)** $t = T$. The strongest vorticity is observed at the end of the pulsing period.

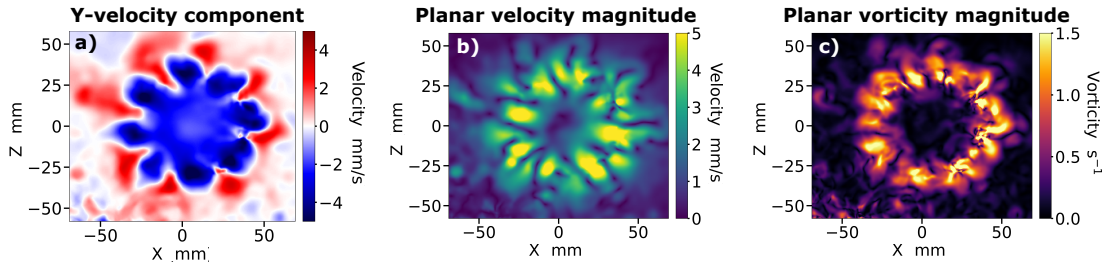


Figure 5.6: A horizontal slice at $L_y = -50$ mm is taken at the end of the pulsing period ($t = T$) for a corresponding Reynolds number of $Re = 100$. The results show **(a)** the y-velocity component, **(b)** the planar velocity magnitude, and **(c)** the planar vorticity magnitude. At the end of the pulsing period, backflow is observed, as indicated in **(a)**. A vortex ring is shed from the robot's tentacles, as shown in **(c)**.

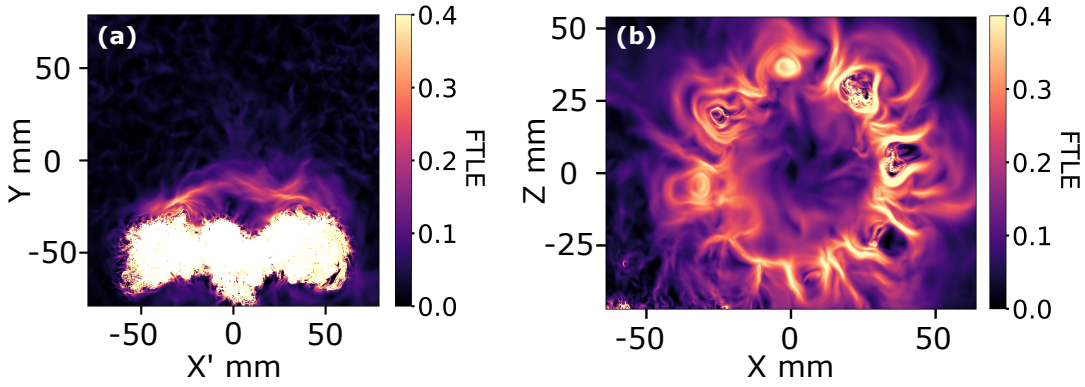


Figure 5.7: **(a)** A vertical slice at $L_z = 2$ mm, rotated 40° about the y-axis. **(b)** A horizontal slice at $L_y = -25$ mm. The results show the forward FTLE field for the expansion phase, from $t = 0.5T$ to $t = T$.

However, more rigorous experiments are needed to confirm this. We observe that the strongest z-vorticity component is present during the end of the pulsing period, $t = T$ (see fig. 5.5), prompting us to investigate this further.

The y-velocity component, planar velocity magnitude, and planar vorticity magnitude are plotted for a horizontal slice $L_y = -50$ mm at the end of the pulsing period $t = T$ (see fig. 5.6). We observe and confirm the presence of strong backflow and the formation of a coherent vortex ring at the end of the expansion phase. Interestingly, this coherent vortex shedding is not present in the live corals [1]. Although, vortex shedding can be advantageous in enhancing fluid mixing its absence might suggest that the energy required to pulse at high frequencies might be too costly for the actual animals.

5.3.2 FTLE fields

We plot the forward FTLE field corresponding to the expansion phase of the robot from $t = 0.5T$ to $t = T$ (see fig. 5.7). In the vertical slice $L_z = 2$ mm, we observe the formation of a ridge near the top surface of the robot (see fig. 5.7**(a)**). This ridge aligns with the approximate location of the vortex ring. However, shadowing effects that result from the robot's opacity prevents us from visualizing the FTLE field at the exact location of the vortex. We investigate this further

by taking a horizontal slice $L_y = -25$ mm (see fig. 5.7**(b)**). Here, the largest stretching, indicated by the magnitude of the FTLE field, appears directly above the location of vortex ring visualized in fig. 5.6**(c)**. This observation aligns with our analysis from chapter 3, where we demonstrated that the most significant stretching occurs near the vortex location.

5.4 Conclusion

Inspired by the flows generated by the pulsing of Xeniidae corals, we designed and fabricated a soft robot that mimics the pulsing motion of these animals. The experimental and computational tools developed throughout this thesis were employed to collect and analyze particle tracking data, enabling us to access and visualize the acquired flow data. To achieve accurate interpolation of the data, we utilized our custom WP interpolator, which provided smooth and precise velocity fields and corresponding derivatives of the interpolated fields (e.g., vorticity field and FTLE field).

The interpolated velocity fields, resolved at different stages of the robot's pulsing period, reveal a strong backflow that appears during the expansion phase. Additionally, our robot sheds a coherent vortex at the end of its pulsing period, a feature that is not observed in the actual corals. The strongest vorticity is observed at the end of the expansion phase of the robot, which also is the when we observe the strongest backflow. A coherent vortex is shed as the robot expands. Lastly, the FTLE field, plotted for the expansion phase of the robot, reveals that the largest stretching occurs near the approximate location of the generated vortex ring.

We have showcased the versatility, applicability, and innovation of our experimental and computational analysis tools for studying a bio-inspired flow. While our focus has been on a specific system, these tools are broadly applicable across a wide range of experimental setups. These tools provide access to 3D flow data, a key characteristic of physical models and real-world flows. While the primary aim of this work is to demonstrate effective methods for capturing and analyzing

flow data, our simplified robotic model establishes a foundation for future studies aimed at improving comparisons between Xeniidae corals and our physical model. However, refining the robot's kinematics and morphology will be necessary to enhance comparisons with live corals. Nonetheless, our imaging technique, analysis tools, and simplified physical model offer a valuable framework for investigating flows generated by rhythmic pulsing.

Chapter 6

Conclusion and future work

In this work, we introduced a novel experimental imaging technique, combined with robust analysis methods to study the fluid dynamics of complex flows. We begun by highlighting the importance of volumetric imaging in experimental fluid dynamics. In chapter 2, we present our high-speed two-color scanning volumetric laser induced fluorescence (H2C-SVLIF) technique which enables direct flow visualization, and the acquisition of particle tracking data. We discuss the fundamental and practical limitations of volumetric imaging to guide researchers that are interested in developing similar imaging systems. To validate our technique, we design a test experiment for flow past a sphere, and directly visualize flow patterns by using fluorescent dye, and acquire particle tracking data using fluorescent labeled particles. The particle tracking data is analyzed and processed to compute the velocity field and the vorticity field. We pair the technique with custom-built software for real-time playback of the large datasets that result from volumetric imaging, and provide calibration scripts to automate the correction of distortion defects for the spatial distortion and the non uniformity present in the fluorescent signal captured by the camera. The technique enables us to study fundamental flow properties at high spatial and temporal resolutions ($512 \times 512 \times 512$ resolutions at ~ 60 Hz) as demonstrated through our validation experiment.

Chapter 3 explores the stretching mechanisms in fluid flows through the lens of vorticity, focusing on how vortex topology and geometry influence the stretching rates of material lines. We develop a simplified model by 'freezing' vortices. Our

findings provide insights into experimentally observed phenomena, where vortex knots tend to stretch and destabilize, leading to unknotting, while vortex rings remain stable and exhibit linear stretching. Our results show that topology significantly impacts stretching dynamics. By visualizing FTLE fields, we show that the stretching of material lines mirrors the deformation of the flow field itself. Although the short-lived nature of knotted vortices limits our understanding of real systems, our simplified model captures the underlying mechanisms of when, why, and how material lines and fluid domains stretch.

Building upon our fundamental understanding of stretching mechanisms of fluid domains, we develop efficient and accurate analysis methods of particle tracking data presented in 4. We present and discuss the two methods for the computation of the flow map, standard velocity field integration (VFI) and flow map compilation (FMC). We pair FMC with a custom windowed polynomial (WP) for the time-efficient and accurate computation of FTLE fields. Our interpolator addresses the limitations of standard radial basis function (RBF) interpolation, which is shown to produce non-smooth derivatives of interpolated fields even with non-noisy data. Our method dynamically adjusts the distanced-based weights of the source points leading to smoother and accurate interpolated fields and derivatives. We validate our approach on the Lamb-Oseen vortex and the double gyre, demonstrating that our interpolator outperforms RBF interpolation in both speed and accuracy, particularly when handling noisy data. Future work could explore incorporating RBF kernels as the basis functions with our weighting scheme to possible enhance accuracy. Preliminary results indicate that computing analytical derivatives of our WP interpolator could yield even better accuracy, though this requires further evaluation. We apply the methods developed throughout this work to analyze flows in a bio-inspired system to showcase their practicality in experiments.

In chapter 5 we apply our imaging system and numerical methods to investigate the fluid dynamics of a soft robot designed to mimic the pulsing behavior of Xeniidae corals. Using our H2C-SVLIF imaging technique, we acquired flow data to compute the velocity, vorticity and FTLE fields at various stages of the robot's pulsing period. The tools developed in this work enables us to handle inherently

noisy experimental data to resolve smooth and accurate fields. Our analysis reveals an upward jet during the contraction phase. Strong back flow is observed during the expansion phase. We observe concentrated vorticity near the top surface of the robot and at the end of the expansion phase. Further refinement of the robot's design is needed, particularly in replicating the kinematics and morphology for a more accurate comparison to the actual coral.

Our tools facilitate the study and analysis of both simulated and experimental data sets, addressing two main themes: 1) developing versatile imaging and analysis methods to process noisy experimental particle tracking data, and 2) investigating fundamental flow properties. We have advanced high-speed volumetric imaging and developed computational tools for the time-efficient and accurate computation of fields derived from these data sets. This work advances the integration of computational and experimental studies of complex flows, particularly in understanding stretching in fluid domains.

Appendix A

Mathematical Definitions

We lay out a consistent notation for tensors and their derivatives (using standard notation [121]). Note that the most important rule to keep in mind is that vectors and vector operations (i.e., dot and cross products) *never* cross the tensor product symbol, $\otimes$.

$$\underline{a} = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} \qquad \text{(general vector)} \quad \text{(A.1)}$$

$$|\underline{a}| = \sqrt{\sum_i a_i^2} \qquad (L^2 - \text{norm}) \quad \text{(A.2)}$$

$$\boldsymbol{a} = \sum_i a_i \hat{\boldsymbol{e}}_i = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \qquad \text{(spatial vector, e.g., } \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}) \quad \text{(A.3)}$$

$$\dot{\boldsymbol{u}} = \frac{d\boldsymbol{u}}{dt} \qquad \text{(time derivative)} \quad \text{(A.4)}$$

$$\mathbf{A} = \sum_{ij} A_{ij}\ \hat{\boldsymbol{e}}_i \otimes \hat{\boldsymbol{e}}_j = \begin{pmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{pmatrix} \qquad \text{(rank 2 tensor)} \quad \text{(A.5)}$$

$$= \begin{pmatrix} \mathbf{A}_{1*}^{\top} \\ \mathbf{A}_{2*}^{\top} \\ \mathbf{A}_{3*}^{\top} \end{pmatrix} \qquad \text{(row vectors)} \tag{A.6}$$

$$\text{where } \boldsymbol{A}_{i*} \equiv \sum_i A_{ij} \hat{\boldsymbol{e}}_i = \begin{pmatrix} A_{i1} \\ A_{i2} \\ A_{i3} \end{pmatrix} \qquad \text{(row vector } i\text{)} \tag{A.7}$$

$$= \begin{pmatrix} \mathbf{A}_{*1} & \mathbf{A}_{*2} & \mathbf{A}_{*3} \end{pmatrix} \qquad \text{(col. vectors)} \tag{A.8}$$

$$\text{where } \boldsymbol{A}_{*j} \equiv \sum_j A_{ij} \hat{\boldsymbol{e}}_j = \begin{pmatrix} A_{1j} \\ A_{2j} \\ A_{3j} \end{pmatrix} \qquad \text{(column vector } j\text{)} \tag{A.9}$$

$$\boldsymbol{\nabla} = \sum_i \hat{\boldsymbol{e}}_i \partial_i = \begin{pmatrix} \partial_1 \\ \partial_2 \\ \partial_3 \end{pmatrix} \qquad \text{(“del”)} \tag{A.10}$$

$$\nabla^2 \boldsymbol{v} = \sum_i \hat{\boldsymbol{e}}_i \left(\partial_1^2 v_i + \partial_2^2 v_i + \partial_3^2 v_i \right) = \begin{pmatrix} \partial_1^2 v_1 + \partial_2^2 v_1 + \partial_3^2 v_1 \\ \partial_1^2 v_2 + \partial_2^2 v_2 + \partial_3^2 v_2 \\ \partial_1^2 v_3 + \partial_2^2 v_3 + \partial_3^2 v_3 \end{pmatrix} \qquad \text{(Laplacian)} \tag{A.11}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{v} = \sum_i \partial_i v_i = \partial_1 v_1 + \partial_2 v_2 + \partial_3 v_3 \qquad \text{(divergence)} \tag{A.12}$$

$$\boldsymbol{\nabla} \times \boldsymbol{v} = \sum_{ijk} \epsilon_{ijk} \partial_i v_j \; \hat{\boldsymbol{e}}_k = \begin{pmatrix} \partial_2 v_3 - \partial_3 v_2 \\ \partial_3 v_1 - \partial_1 v_3 \\ \partial_1 v_2 - \partial_2 v_1 \end{pmatrix} \qquad \text{(curl)}$$

(A.13)

$$\boldsymbol{\nabla} \otimes \boldsymbol{v} = \sum_{ij} \partial_i v_j \; \hat{\boldsymbol{e}}_i \otimes \hat{\boldsymbol{e}}_j = \begin{pmatrix} \partial_1 v_1 & \partial_1 v_2 & \partial_1 v_3 \\ \partial_2 v_1 & \partial_2 v_2 & \partial_2 v_3 \\ \partial_3 v_1 & \partial_3 v_2 & \partial_3 v_3 \end{pmatrix} \qquad \text{(tensor deriv.)}$$

(A.14)

$$= \begin{pmatrix} \boldsymbol{\nabla} v_1 & \boldsymbol{\nabla} v_2 & \boldsymbol{\nabla} v_3 \end{pmatrix} \qquad \text{(as columns)}$$

(A.15)

$$= \begin{pmatrix} \partial_1 \boldsymbol{v}^\top \\ \partial_2 \boldsymbol{v}^\top \\ \partial_3 \boldsymbol{v}^\top \end{pmatrix} \qquad \text{(as rows)}$$

(A.16)

$$\mathbf{A} \cdot \boldsymbol{v} = \sum_{ij} A_{ij} v_j \hat{\boldsymbol{e}}_i \qquad \text{(A.17)}$$

$$= \sum_{i} \boldsymbol{v} \cdot \boldsymbol{A}_{i*} \hat{\boldsymbol{e}}_i \qquad \text{(A.18)}$$

$$= \mathbf{A}\boldsymbol{v} \qquad \text{(matrix mult. form)}$$

(A.19)

$$= \begin{pmatrix} \boldsymbol{A}_{1*}^{\mathrm{tr}} \\ \boldsymbol{A}_{2*}^{\mathrm{tr}} \\ \boldsymbol{A}_{3*}^{\mathrm{tr}} \end{pmatrix} \cdot \boldsymbol{v} \qquad \text{(A.20)}$$

$$= \begin{pmatrix} \boldsymbol{A}_{1*} \cdot \boldsymbol{v} \\ \boldsymbol{A}_{2*} \cdot \boldsymbol{v} \\ \boldsymbol{A}_{3*} \cdot \boldsymbol{v} \end{pmatrix} \qquad \text{(i.e. rows } \cdot \boldsymbol{v}\text{)}$$

(A.21)

$$\text{Note: } (\boldsymbol{\nabla} \otimes \boldsymbol{a}) \cdot \boldsymbol{b} = \begin{pmatrix} (\partial_1 \boldsymbol{a}) \cdot \boldsymbol{b} \\ (\partial_2 \boldsymbol{a}) \cdot \boldsymbol{b} \\ (\partial_3 \boldsymbol{a}) \cdot \boldsymbol{b} \end{pmatrix} \qquad \text{(A.22)}$$

$$\boldsymbol{v} \cdot \mathbf{A} = \sum_{ij} v_i A_{ij} \hat{\boldsymbol{e}}_j \tag{A.23}$$

$$= \sum_{j} \boldsymbol{v} \cdot \boldsymbol{A}_{*j} \hat{\boldsymbol{e}}_j \tag{A.24}$$

$$= \mathbf{A}^\top \boldsymbol{v} \qquad \text{(matrix mult. form)} \tag{A.25}$$

$$= \begin{pmatrix} \boldsymbol{A}_{*1}^\top \\ \boldsymbol{A}_{*2}^\top \\ \boldsymbol{A}_{*3}^\top \end{pmatrix} \cdot \boldsymbol{v} \tag{A.26}$$

$$= \begin{pmatrix} \boldsymbol{v} \cdot \boldsymbol{A}_{*1} \\ \boldsymbol{v} \cdot \boldsymbol{A}_{*2} \\ \boldsymbol{v} \cdot \boldsymbol{A}_{*3} \end{pmatrix} \qquad \text{(i.e. } \boldsymbol{v}\cdot \text{ columns)} \tag{A.27}$$

$$\text{Note: } \boldsymbol{a} \cdot (\boldsymbol{\nabla} \otimes \boldsymbol{b}) = \begin{pmatrix} \boldsymbol{a} \cdot \boldsymbol{\nabla} b_1 \\ \boldsymbol{a} \cdot \boldsymbol{\nabla} b_2 \\ \boldsymbol{a} \cdot \boldsymbol{\nabla} b_3 \end{pmatrix} = \boldsymbol{\nabla}_{\boldsymbol{a}} \boldsymbol{b} \qquad \text{(directional deriv.)} \tag{A.28}$$

We can take the dot product between two tensors, which is equivalent to the result of a normal matrix multiplication:

$$\mathbf{A} \cdot \mathbf{B} = \sum_{ijk\ell} A_{ij} B_{kj} \hat{\boldsymbol{e}}_i \otimes \underbrace{\hat{\boldsymbol{e}}_j \cdot \hat{\boldsymbol{e}}_k}_{\boldsymbol{\delta}_{jk}} \otimes \hat{\boldsymbol{e}}_\ell \tag{A.29}$$

$$= \mathbf{AB} \qquad \text{(normal matrix mult.)} \tag{A.30}$$

$$\Rightarrow (\mathbf{A} \cdot \mathbf{B})_{ij} = \sum_{k} A_{ik} B_{kj} \tag{A.31}$$

Appendix B

Analytical derivatives and Gram-Schmidt process

Analytical derivatives for the windowed polynomial (WP) interpolation can be explicitly derived to compute the derivatives of the interpolated fields, e.g. vorticity and FTLE fields. These derivatives include both the polynomial basis functions and the weighting functions. Additionally, derivatives from the Gram-Schmidt orthogonalization process are also necessary to compute.

The interpolation function is constructed as:

$$X'_{nm} = w_n \phi_m(\boldsymbol{r}_n) \tag{B.1}$$

(B.2)

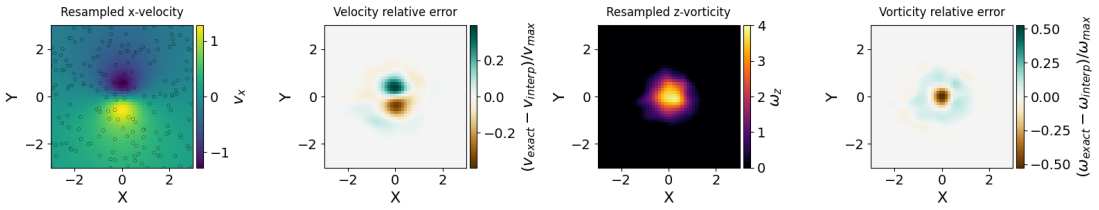


Figure B.1: WP interpolated fields using a Hann window, where the vorticity field is computed using the analytical derivatives of the interpolated velocity field. From left to right: the interpolated x-component of the velocity field component, the relative error between the exact and interpolated velocity fields, the z-component of the vorticity, and the relative error between the exact and the interpolated z-component of the vorticity.

where $\boldsymbol{r}_n = \boldsymbol{x}_n - \boldsymbol{x}_i$ is the relative distance between the interpolated target, $\boldsymbol{x}_n$ and the source point, $\boldsymbol{x}_i$. The polynomial basis function is given by:

$$\phi_m(\boldsymbol{r}_n) = \prod_j r_{nj}^{\alpha_{mj}} \tag{B.3}$$

with α_{mk} is the matrix of powers for the polynomial basis. Taking derivatives of the polynomial basis and the weight function yields:

$$\partial_k \boldsymbol{r}_n = -\hat{\boldsymbol{e}}_{\boldsymbol{k}} \tag{B.4}$$

$$\partial_k r_n = -\hat{\boldsymbol{e}}_{\boldsymbol{k}} \cdot \hat{\boldsymbol{r}}_{\boldsymbol{n}} \tag{B.5}$$

$$\partial_k \phi_m(\boldsymbol{r}_n) = -\alpha_{nk} r_{nk}^{(\alpha_m k - 1)} \prod_{j \neq k} r_{nj}^{\alpha_{mj}} \tag{B.6}$$

$$\partial_k w_n(r_n) = -w_n'(\hat{\boldsymbol{e}}_{\boldsymbol{k}} \cdot \hat{\boldsymbol{r}}_{\boldsymbol{n}}) \tag{B.7}$$

where $w' = dw(r)/dr$. Combining these, the derivative of the interpolated function yields:

$$X'_{nm} = [\partial_k w_n]\phi_m(\boldsymbol{r}_n) + w_n[\partial_k \phi_m(\boldsymbol{r}_n)] \tag{B.8}$$

Gram-Schmdit process ensures orthogonality of the polynomial basis. Given a set of linearly independent vectors $\underline{a}$ we can find a set of orthonormalized vectors $\underline{q}$. We initialize the first orthogonalized vector $\underline{q}_1 = \underline{a}_1$. Subsequent vectors are orthogonalized by subtracting the projections of $\underline{a}_m$ onto the previously calculated orthogonormal vectors. The intermediate vectors $\underline{q}_m$ are computed as:

$$\underline{a}_m^{(0)} = \underline{X}'_m \tag{B.9}$$

$$\underline{a}_m^{(j)} = \underline{a}_m^{(j-1)} - \left(\underline{q}_j \cdot \underline{a}_m^{(j-1)}\right) \underline{q}_j \qquad \text{(note that } j < m\text{)} \tag{B.10}$$

$$\underline{q}_m = \frac{\underline{a}_m^{(m-1)}}{|\underline{a}_m^{(m-1)}|} \tag{B.11}$$

$$\tag{B.12}$$

Derivatives are computed as:

$$\partial_k \underline{a}_m^{(0)} = \partial_k \underline{X}'_m \tag{B.13}$$

$$\partial_k \underline{a}_m^{(j)} = \partial_k \underline{a}_m^{(j-1)} - \left(\partial_k \underline{q}_j \cdot \underline{a}_m^{(j-1)} + \underline{q}_j \cdot \partial_k \underline{a}_m^{(j-1)}\right) \underline{q}_j - \left(\underline{q}_j \cdot \underline{a}_m^{(j-1)}\right) \partial_k \underline{q}_j \tag{B.14}$$

$$\partial_k \underline{q}_m = \frac{\partial_k \underline{a}_m^{(m-1)}}{|\underline{a}_m^{(m-1)}|} - \frac{\underline{a}_m^{(m-1)} \left(\partial_k \underline{a}_m^{(m-1)} \cdot \underline{a}_m^{(m-1)}\right)}{|\underline{a}_m^{(m-1)}|^3} \tag{B.15}$$

$$= \frac{\partial_k \underline{a}_m^{(m-1)} - \left(\underline{q}_m \cdot \partial_k \underline{a}_m^{(m-1)}\right) \underline{q}_m}{|\underline{a}_m^{(m-1)}|} \tag{B.16}$$

The final interpolant function $\tilde{y}$ is given by:

$$\tilde{y} = \frac{\underline{p} \cdot \underline{y}'}{\underline{p} \cdot \underline{w}} \tag{B.17}$$

$$y'_n = w_n y_n \tag{B.18}$$

$$\underline{p} = \underline{q}_M. \tag{B.19}$$

$$\partial_k \tilde{y} = \frac{\partial_k \underline{p} \cdot \underline{y}' + \underline{p} \cdot \partial_k \underline{y}' - \tilde{y} \left(\partial_k \underline{p} \cdot \underline{w} + \underline{p} \cdot \partial_k \underline{w}\right)}{\underline{p} \cdot \underline{w}} \tag{B.20}$$

$$(\partial_w \underline{y}')_n = (\partial_k w_n) y_n + w_n \underbrace{\partial_k y_n}_{0} \tag{B.21}$$

In fig B.1 we compute the interpolated velocity field, and use the analytical derivatives of the interpolated field to calculate the vorticity field for the Lamb-Oseen vortex as we did in chapter 4. The results are consistent with those obtained using discrete differentiation. More rigorous testing is needed to fully validate the analytical derivatives.

Bibliography

[1] Julia E. Samson, Laura A. Miller, Dylan Ray, Roi Holzman, Uri Shavit, and Shilpa Khatri. A novel mechanism of mixing by pulsing corals. *Journal of Experimental Biology*, 222(15), August 2019.

[2] Shashank Kamdar, Seunghwan Shin, Premkumar Leishangthem, Lorraine F. Francis, Xinliang Xu, and Xiang Cheng. The colloidal nature of complex fluids enhances bacterial motility. *Nature*, 603(7903):819–823, March 2022.

[3] Howard C. Berg. Symmetries in bacterial motility. *Proceedings of the National Academy of Sciences*, 93(25):14225–14228, December 1996.

[4] David Merritt. Dynamics of Elliptical Galaxies. *Science*, 259(5103):1867–1871, March 1993.

[5] Yu-Qing Lou. Two-fluid Dynamics in Clusters of Galaxies Pairwise Velocity Statistics of Dark Halos. *Chinese Journal of Astronomy and Astrophysics*, 5(1):6, February 2005.

[6] B W van Oudheusden. PIV-based pressure measurement. *Measurement Science and Technology*, 24(3):032001, March 2013.

[7] Steven J Beresh, John F Henfling, Russell W Spillers, and Seth M Spitzer. 'Postage-stamp PIV': Small velocity fields at 400 kHz for turbulence spectra measurements. *Measurement Science and Technology*, 29(3):034011, March 2018.

[8] S. Tokgoz, G. E. Elsinga, R. Delfos, and J. Westerweel. Spatial resolution

and dissipation rate estimation in Taylor–Couette flow for tomographic PIV. *Experiments in Fluids*, 53(3):561–583, September 2012.

[9] Mac Panah, François Blanchette, and Shilpa Khatri. Simulations of a porous particle settling in a density-stratified ambient fluid. *Physical Review Fluids*, 2(11):114303, November 2017.

[10] P. Sánchez-Martín, J. J. Masdemont, and M. Romero-Gómez. From manifolds to Lagrangian coherent structures in galactic bar models. *Astronomy & Astrophysics*, 618:A72, October 2018.

[11] Henrik Juul Spietz, Mads Mølholm Hejlesen, and Jens Honoré Walther. Iterative Brinkman penalization for simulation of impulsively started flow past a sphere and a circular disc. *Journal of Computational Physics*, 336:261–274, May 2017.

[12] Cameron Tropea, Alexander L. Yarin, and John F. Foss. *Springer Handbook of Experimental Fluid Mechanics*. Gale Virtual Reference Library. Springer, Berlin, 2007.

[13] Yue Wu, Wenjiang Xu, Qingchun Lei, and Lin Ma. Single-shot volumetric laser induced fluorescence (VLIF) measurements in turbulent flows seeded with iodine. *Optics Express*, 23(26):33408, December 2015.

[14] Lin Ma, Qingchun Lei, Jordan Ikeda, Wenjiang Xu, Yue Wu, and Campbell D. Carter. Single-shot 3D flame diagnostic based on volumetric laser induced fluorescence (VLIF). *Proceedings of the Combustion Institute*, 36(3):4575–4583, January 2017.

[15] Michel Versluis. High-speed imaging in fluids. *Experiments in Fluids*, 54(2):1458, February 2013.

[16] Stephen D. Hammack, Campbell D. Carter, Aaron W. Skiba, Christopher A. Fugger, Josef J. Felver, Joseph D. Miller, James R. Gord, and Tonghun Lee. 20 kHz CH $_2$ O and OH PLIF with stereo PIV. *Optics Letters*, 43(5):1115, March 2018.

[17] Xuanyu Zhu, Daniel R. Sabatino, and Tobias Rossman. 3D planar laser-induced fluorescence (PLIF) reconstruction of a hairpin vortex. In *AIAA*, 2019.

[18] P.J. Bryanston-Cross and A. Epstein. The application of sub-micron particle visualisation for PIV (Particle Image Velocimetry) at transonic and supersonic speeds. *Progress in Aerospace Sciences*, 27(3):237–265, January 1990.

[19] P. Palafox, M. L. G. Oldfield, J. E. LaGraff, and T. V. Jones. PIV Maps of Tip Leakage and Secondary Flow Fields on a Low-Speed Turbine Blade Cascade With Moving End Wall. *Journal of Turbomachinery*, 130(1), December 2007.

[20] A. A. Adamczyk and L. Rimai. 2-Dimensional particle tracking velocimetry (PTV): Technique and image processing algorithms. *Experiments in Fluids*, 6(6):373–380, January 1988.

[21] C. Brücker and W. Althaus. Study of vortex breakdown by particle tracking velocimetry (PTV). *Experiments in Fluids*, 13(5):339–349, September 1992.

[22] Martin Elfert, Michael Schroll, and Wolfgang Förster. PIV Measurement of Secondary Flow in a Rotating Two-Pass Cooling System With an Improved Sequencer Technique. *Journal of Turbomachinery*, 134(3), July 2011.

[23] Sven Scharnowski, Matthew Bross, and Christian J. Kähler. Accurate turbulence level estimations using PIV/PTV. *Experiments in fluids*, 60(1):1–12, 2019.

[24] Jan F. G. Schneiders, Giuseppe C. A. Caridi, Andrea Sciacchitano, and Fulvio Scarano. Large-scale volumetric pressure from tomographic PTV with HFSB tracers. *Experiments in Fluids*, 57(11):164, November 2016.

[25] Carlo F. Barenghi, David C. Samuels, Gregory H. Bauer, and Russell J. Donnelly. Superfluid vortex lines in a model of turbulent flow. *Physics of Fluids*, 9(9):2631–2643, September 1997.

[26] A. Leonard. Numerical simulation of interacting three-dimensional vortex filaments. In Robert D. Richtmyer, editor, *Proceedings of the Fourth International Conference on Numerical Methods in Fluid Dynamics*, volume 35, pages 245–250. Springer-Verlag, Berlin/Heidelberg, 1975.

[27] Jacob Bedrossian, Pierre Germain, and Benjamin Harrop-Griffiths. Vortex filament solutions of the Navier-Stokes equations. *arXiv:1809.04109 [physics]*, March 2020.

[28] T. Maxworthy. The structure and stability of vortex rings. *Journal of Fluid Mechanics*, 51(1):15–32, January 1972.

[29] C.-H. Krutzsch, D. Bolster, R. Hershberger, and R.J. Donnelly. On an experimentally observed phenomenon on vortex rings during their translational movement in a real liquid. *Annalen der Physik*, 523(5):360–379, May 2011.

[30] Dustin Kleckner, Louis H. Kauffman, and William T. M. Irvine. How superfluid vortex knots untie. *Nature Physics*, 12:650–655, March 2016.

[31] Kun Li, Chiya Savari, and Mostafa Barigou. Computation of Lagrangian coherent structures from experimental fluid trajectory measurements in a mechanically agitated vessel. *Chemical Engineering Science*, 254:117598, June 2022.

[32] J. Peng and J. O. Dabiri. Transport of inertial particles by Lagrangian coherent structures: Application to predator–prey interaction in jellyfish feeding. *Journal of Fluid Mechanics*, 623:75–84, March 2009.

[33] G. Haller and G. Yuan. Lagrangian coherent structures and mixing in two-dimensional turbulence. *Physica D: Nonlinear Phenomena*, 147(3-4):352–370, December 2000.

[34] George Haller and Themistoklis Sapsis. Lagrangian coherent structures and the smallest finite-time Lyapunov exponent. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(2):023115, June 2011.

[35] Steven L. Brunton and Clarence W. Rowley. Fast computation of finite-time Lyapunov exponent fields for unsteady flows. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 20(1):017503, March 2010.

[36] Samuel G. Raben, Shane D. Ross, and Pavlos P. Vlachos. Computation of finite-time Lyapunov exponents from time-resolved particle image velocimetry data. *Experiments in Fluids*, 55(1):1638, January 2014.

[37] Julia E. Samson, Nicholas A. Battista, Shilpa Khatri, and Laura A. Miller. Pulsing corals: A story of scale and mixing. *BIOMATH*, 6(2):1712169, December 2017.

[38] Kakani Katija, Sean P. Colin, John H. Costello, and Houshuo Jiang. Ontogenetic propulsive transitions by Sarsia tubulosa medusae. *Journal of Experimental Biology*, 218(15):2333–2343, August 2015.

[39] Rachel E. Pepper, Jules S. Jaffe, Evan Variano, and M. A. R. Koehl. Zooplankton in flowing water near benthic communities encounter rapidly fluctuating velocity gradients and accelerations. *Marine Biology*, 162(10):1939–1954, October 2015.

[40] Brooke E. Flammang, George V. Lauder, Daniel R. Troolin, and Tyson Strand. Volumetric imaging of shark tail hydrodynamics reveals a three-dimensional dual-ring vortex wake structure. *Proceedings of the Royal Society B: Biological Sciences*, 278(1725):3670–3678, December 2011.

[41] C. D. Wilga and G. V. Lauder. Function of the heterocercal tail in sharks: Quantitative wake dynamics during steady horizontal swimming and vertical maneuvering. *Journal of Experimental Biology*, 205(16):2365–2374, 2002.

[42] C. D. Wilga and G. V. Lauder. Hydrodynamic function of the shark's tail. *Nature (London)*, 430(7002):850–850, 2004.

[43] Nicholas T. Ouellette. Turbulence in two dimensions. *Physics Today*, 65(5):68–69, May 2012.

[44] Cecilia Ortiz-Dueñas, Jungyong Kim, and Ellen K. Longmire. Investigation of liquid–liquid drop coalescence using tomographic PIV. *Experiments in Fluids*, 49(1):111–129, July 2010.

[45] Philippe Cornic, Benjamin Leclaire, Frédéric Champagnat, Guy Le Besnerais, Adam Cheminet, Cédric Illoul, and Gilles Losfeld. Double-frame tomographic PTV at high seeding densities. *Experiments in Fluids*, 61(2):23, February 2020.

[46] Terrence R. Meyer, Benjamin R. Halls, Naibo Jiang, Mikhail N. Slipchenko, Sukesh Roy, and James R. Gord. High-speed, three-dimensional tomographic laser-induced incandescence imaging of soot volume fraction in turbulent flames. *Optics express*, 24(26):29547–29555, 2016.

[47] D. Krug, M. Holzner, B. Lüthi, M. Wolf, A. Tsinober, and W. Kinzelbach. A combined scanning PTV/LIF technique to simultaneously measure the full velocity gradient tensor and the 3D density field. *Measurement Science and Technology*, 25(6):065301, April 2014.

[48] Jeffrey R. Osborne, Sarah A. Ramji, Campbell D. Carter, Scott Peltier, Stephen Hammack, Tonghun Lee, and Adam M. Steinberg. Simultaneous 10 kHz TPIV, OH PLIF, and CH2O PLIF measurements of turbulent flame structure and dynamics. *Experiments in Fluids*, 57(5):65, May 2016.

[49] Shengxian Shi, Junfei Ding, Callum Atkinson, Julio Soria, and T. H. New. A detailed comparison of single-camera light-field PIV and tomographic PIV. *Experiments in Fluids*, 59(3):46, March 2018.

[50] B. R. Halls, P. S. Hsu, S. Roy, T. R. Meyer, and J. R. Gord. Two-color volumetric laser-induced fluorescence for 3D OH and temperature fields in turbulent reacting flows. *Optics Letters*, 43(12):2961, June 2018.

[51] Sina Kashanj and David S. Nobes. Application of 4D two-colour LIF to explore the temperature field of laterally confined turbulent Rayleigh–Bénard convection. *Experiments in fluids*, 64(3), 2023.

[52] Dustin Kleckner. Kleckner lab github repository: Muvi lab 3d imaging library. `https://github.com/klecknerlab/muvi`, 2021.

[53] Dustin Kleckner. Kleckner lab github repository: Analog-digital synchronization board schematics. `https://github.com/klecknerlab/ad_sync`, 2021.

[54] Siegman Anthony E. Physical Properties of Gaussian Beams. In *Lasers*, pages 1–1. University Science Books, 1986.

[55] P. Ward and E. Jacobson. *Manual of Photography*. Burlington: Focal Press, 9th edition, 2000.

[56] Anthony Gerrard and James M. Burch. *Introduction to Matrix Methods in Optics*. Courier Corporation, January 1994.

[57] James R. Janesick. *Photon Transfer*. SPIE, Bloomington, 2007.

[58] Kyle Gilroy and Toni Lucatorto. Overcoming Shot Noise Limitations in High Speed Imaging with Bright Field Mode, January 2019.

[59] Z. HONG, H. ZHENG, J. CHEN, and J. GE. Laser-diode-pumped Cr4+, Nd3+:YAG self-Q-switched laser with high repetition rate and high stability. *Applied physics. B, Lasers and optics*, 73(3):205–207, 2001.

[60] Will Schroeder. *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. Kitware, New York, 4th ed. edition, 2006.

[61] J. Reznicek. Method for Measuring Lens Distortion by Using Pinhole Lens. *International archives of the photogrammetry, remote sensing and spatial information sciences.*, XL-5(5):509–515, 2014.

[62] L. Eshbal, V. Rinsky, T. David, D. Greenblatt, and R. van Hout. Measurement of vortex shedding in the wake of a sphere at $Re=465$. *Journal of Fluid Mechanics*, 870:290–315, July 2019.

[63] Mengyang Lv, Shuzhen Chen, and Wenjiang Xu. 3D spatial resolution characterization for volumetric computed tomography. *AIP Advances*, 12(3):035322, March 2022.

[64] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer Series in Operations Research and Financial Engineering. Springer, New York, NY, second edition. edition, 2006.

[65] Julien Manin, Scott A. Skeen, and Lyle M. Pickett. Performance comparison of state-of-the-art high-speed video cameras for scientific applications. *Optical Engineering*, 57(12):1, December 2018.

[66] Jürgen Mayer, Alexandre Robert-Moreno, James Sharpe, and Jim Swoger. Attenuation artifacts in light sheet fluorescence microscopy corrected by OPTiSPIM. *Light, Science & Applications*, 7:70, 2018.

[67] Alex Liberzon, Davide Lasagna, Mathias Aubert, Pete Bachant, Theo Käufer, jakirkham, Andreas Bauer, Boyko Vodenicharski, Cameron Dallas, Joe Borg, tomerast, and ranleu. OpenPIV/openpiv-python: OpenPIV - Python (v0.22.2) with a new extended search PIV grid option, July 2020.

[68] Daniel Allan, Thomas Caswell, Nathan Keim, and Casper van der Wel. trackpy: Trackpy v0.3.2, August 2016.

[69] John C. Crocker and David G. Grier. Methods of digital video microscopy for colloidal studies. *Journal of colloid and interface science*, 179(1):298–310, 1996.

[70] Daniel Schanz, Andreas Schröder, Sebastian Gesemann, Dirk Michaelis, and Bernhard Wieneke. 'Shake The Box': A highly efficient and accurate Tomographic Particle Tracking Velocimetry (TOMO-PTV) method using prediction of particle positions.

[71] Andreas Schröder and Daniel Schanz. 3D Lagrangian Particle Tracking in Fluid Mechanics. *Annual Review of Fluid Mechanics*, 55(1):511–540, 2023.

[72] Abraham. Savitzky and M. J. E. Golay. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, 36(8):1627–1639, July 1964.

[73] Thomas Janke, Rüdiger Schwarze, and Katrin Bauer. Measuring three-dimensional flow structures in the conductive airways using 3D-PTV. *Experiments in Fluids*, 58(10):133, October 2017.

[74] Mehdi Parviz and M. Shahram Moin. Multivariate Polynomials Estimation Based on GradientBoost in Multimodal Biometrics. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques*, Communications in Computer and Information Science, pages 471–477. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[75] Tsutomu Kambe. *Elementary Fluid Mechanics*. World Scientific, Hackensack, N.J. ; London, 2007.

[76] Nezih C. Geçkinli and Davras Yavuz. A set of optimal discrete linear smoothers. *Signal processing*, 3(1):49–62, 1981.

[77] A. Sansica, J.-Ch Robinet, F. Alizard, and E. Goncalves. Three-dimensional instability of a flow past a sphere: Mach evolution of the regular and Hopf bifurcations. *Journal of fluid mechanics*, 855:1088–1115, 2018.

[78] John T. Snow. A review of recent advances in tornado vortex dynamics. *Reviews of Geophysics*, 20(4):953, 1982.

[79] Enrico Brunelli and Carlo Marchioro. Vanishing viscosity limit for a smoke ring with concentrated vorticity. *Journal of Mathematical Fluid Mechanics*, 13(3):421–428, September 2011.

[80] S. Douady, Y. Couder, and M. E. Brachet. Direct observation of the intermittency of intense vorticity filaments in turbulence. *Physical Review Letters*, 67(8):983–986, August 1991.

[81] Rupert Klein, Andrew J. Majda, and Kumaran Damodaran. Simplified equations for the interaction of nearly parallel vortex filaments. *Journal of Fluid Mechanics*, 288:201–248, April 1995.

[82] Iwo Bialynicki-Birula, Zofia Bialynicka-Birula, and Cezary Śliwa. Motion of vortex lines in quantum mechanics. *Physical Review A*, 61(3):032110, February 2000.

[83] Alain Pumir and Robert M. Kerr. Numerical simulation of interacting vortex tubes. *Physical Review Letters*, 58(16):1636–1639, April 1987.

[84] Philip McGavin and David I. Pontin. Vortex line topology during vortex tube reconnection. *arXiv:1805.01202 [physics]*, May 2018.

[85] Jack Binysh, Carl A. Whitfield, and Gareth P. Alexander. Stable and unstable vortex knots in excitable media. *Physical Review E*, 99(1):012211, January 2019.

[86] Y Kimura and H K Moffatt. Scaling properties towards vortex reconnection under Biot–Savart evolution. *Fluid Dynamics Research*, 50(1):011409, February 2018.

[87] Dustin Kleckner and William T. M. Irvine. Creation and dynamics of knotted vortices. *Nature Physics*, 9(4):253–258, April 2013.

[88] Martin W. Scheeler, Wim M. van Rees, Hridesh Kedia, Dustin Kleckner, and William T. M. Irvine. Complete measurement of helicity and its dynamics in vortex tubes. *Science*, 357(6350):487–491, August 2017.

[89] Dhawal Buaria, Eberhard Bodenschatz, and Alain Pumir. Vortex stretching and enstrophy production in high Reynolds number turbulence. *Physical Review Fluids*, 5(10):104602, October 2020.

[90] H. K. Moffatt, S. Kida, and K. Ohkitani. Stretched vortices – the sinews of turbulence; large-Reynolds-number asymptotics. *Journal of Fluid Mechanics*, 259:241–264, January 1994.

[91] Robert M. Kerr. Trefoil knot structure during reconnection. *arXiv:1703.01676 [physics]*, May 2017.

[92] Rupert Klein and Andrew J. Majda. Self-stretching of a perturbed vortex filament I. The asymptotic equation for deviations from a straight line. *Physica D: Nonlinear Phenomena*, 49(3):323–352, April 1991.

[93] Yoshifumi Kimura and H. K. Moffatt. A tent model of vortex reconnection under Biot–Savart evolution. *Journal of Fluid Mechanics*, 834:R1, January 2018.

[94] Alberto Villois, Davide Proment, and Giorgio Krstulovic. Evolution of a superfluid vortex filament tangle driven by the Gross-Pitaevskii equation. *Physical Review E*, 93(6):061103, June 2016.

[95] A. Leonard. Curvature and torsion of material lines in chaotic flows. *Fluid Dynamics Research*, 36(4-6):261, April 2005.

[96] A. Leonard. The universal structure of high-curvature regions of material lines in chaotic flows. *Journal of Fluid Mechanics*, 622:167–175, March 2009.

[97] Jean-Luc Thiffeault. Stretching and Curvature of Material Lines in Chaotic Flows. *Physica D: Nonlinear Phenomena*, 198(3-4):169–181, November 2004.

[98] Nikolaos D. Katopodes. Vorticity Dynamics. In *Free-Surface Flow*, pages 516–565. Elsevier, 2019.

[99] Philip L. Boyland, Hassan Aref, and Mark A. Stremler. Topological fluid mechanics of stirring. *Journal of Fluid Mechanics*, 403:277–304, January 2000.

[100] Lukas Bentkamp, Theodore D. Drivas, Cristian C. Lalescu, and Michael Wilczek. The statistical geometry of material loops in turbulence. *Nature Communications*, 13(1):2088, April 2022.

[101] James D. Hanson and Steven P. Hirshman. Compact expressions for the Biot–Savart fields of a filamentary segment. *Physics of Plasmas*, 9(10):4410–4412, October 2002.

[102] J. R. Dormand and P. J. Prince. A family of embedded Runge-Kutta formulae. *Journal of Computational and Applied Mathematics*, 6(1):19–26, March 1980.

[103] Lawrence F. Shampine. Some practical Runge-Kutta formulas. *Mathematics of Computation*, 46(173):135–150, 1986.

[104] George Haller. Lagrangian Coherent Structures. *Annual Review of Fluid Mechanics*, 47(1):137–162, January 2015.

[105] Jean-Luc Thiffeault and Matthew D Finn. Topology, braids and mixing in fluids. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 364(1849):3251–3266, December 2006.

[106] G. Haller. Lagrangian coherent structures from approximate velocity data. *Physics of Fluids*, 14(6):1851–1861, June 2002.

[107] Samuel Gillooly Raben. Determination of Three Dimensional Time Varying Flow Structures. page 131.

[108] Amir E. BozorgMagham, Shane D. Ross, and David G. Schmale. Real-time prediction of atmospheric Lagrangian coherent structures based on forecast data: An application and error analysis. *Physica D: Nonlinear Phenomena*, 258:47–60, September 2013.

[109] Megan M Wilson, Jifeng Peng, John O Dabiri, and Jeff D Eldredge. Lagrangian coherent structures in low Reynolds number swimming. *Journal of Physics: Condensed Matter*, 21(20):204105, May 2009.

[110] Nicholas Rott. On the viscous core of a line vortex. *Zeitschrift für angewandte Mathematik und Physik ZAMP*, 9(5):543–553, March 1958.

[111] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 67–76, New York, NY, USA, August 2001. Association for Computing Machinery.

[112] Cristian Rusu and Virginia Rusu. Radial Basis Functions Versus Geostatistics in Spatial Interpolations. In Max Bramer, editor, *Artificial Intelligence in Theory and Practice*, pages 119–128, Boston, MA, 2006. Springer US.

[113] Wilna Du Toit. *Radial Basis Function Interpolation*. PhD thesis, Stellenbosch : Stellenbosch University, March 2008.

[114] Ke Fabricius and Dw Klumpp. Widespread mixotrophy in reef-inhabiting soft corals:the influence of depth, and colony expansion and contraction on photosynthesis. *Marine Ecology Progress Series*, 125:195–204, 1995.

[115] J. M. Lough. 1997–98: Unprecedented thermal stress to coral reefs? *Geophysical Research Letters*, 27(23):3901–3904, 2000.

[116] Blake L. Spady, William J. Skirving, Gang Liu, Jacqueline L. De La Cour, Cathy J. McDonald, and Derek P. Manzello. Unprecedented early-summer heat stress and forecast of coral bleaching on the Great Barrier Reef, 2021-2022. *F1000Research*, 11:127, November 2022.

[117] Shihori Inoue, Hajime Kayanne, Shoji Yamamoto, and Haruko Kurihara. Spatial community shift from hard to soft corals in acidified water. *Nature Climate Change*, 3(7):683–687, July 2013.

[118] John B. Lewis. Feeding behaviour and feeding ecology of the Octocorallia (Coelenterata: Anthozoa). *Journal of Zoology*, 196(3):371–384, 1982.

[119] Maya Kremien, Uri Shavit, Tali Mass, and Amatzia Genin. Benefit of pulsation in soft corals. *Proceedings of the National Academy of Sciences*, 110(22):8978–8983, May 2013.

[120] Bobak Mosadegh, Panagiotis Polygerinos, Christoph Keplinger, Sophia Wennstedt, Robert F. Shepherd, Unmukt Gupta, Jongmin Shim, Katia Bertoldi, Conor J. Walsh, and George M. Whitesides. Pneumatic Networks for Soft Robotics that Actuate Rapidly. *Advanced Functional Materials*, 24(15):2163–2170, 2014.

[121] Yifei Yu, Yinlin Dong, and Chaoqun Liu. Rules of Tensor and Matrix Operation for Liutex Calculation. In Yiqian Wang, Yisheng Gao, and Chaoqun Liu, editors, *Liutex and Third Generation of Vortex Identification*, pages 35–43, Singapore, 2023. Springer Nature.