Chasing After the Wind:

Flow Structure Detection Strategies for Autonomous Mobile Flow Field Measurements

Thesis by
Tanner David Harms

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy in Aerospace Engineering

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2025
Defended September 3, 2024

ACKNOWLEDGEMENTS

Alone, we can do so little; together, we can do so much.

— Helen Keller

Occasionally, during introductions, the fact surfaces that I am attempting to complete a PhD in aerospace engineering at Caltech. Invariably, the other person remarks to the effect of, "You must be smart!" The sentiment is kind, but I am never sure how to respond (perhaps this is evidence to the contrary?). Usually, I laugh and reply, "just stubborn and curious," since those attributes seem to overlap most between myself and my vision of a scientist. However, the truth of the matter is that I am—and have always been—surrounded by excellent people. My strength is in my community. It is in the union of people who, by example, challenge me to pursue great things; who encourage me when I struggle to overcome seemingly insurmountable obstacles; who rejoice as milestones are crossed; and who love and support me regardless of the letters after my name. It is with great pleasure and sincerity that I acknowledge many of those wonderful individuals.

First, I am sincerely grateful to my outstanding advisor, Beverley McKeon, for providing me the opportunity to grow as a researcher under her guidance. By her stellar example, I have learned what it is to conduct research with excellence, integrity, and professionalism. Not only does Beverley produce world-class research in her field, she has built a laboratory environment that is collaborative, encouraging, and oriented towards outreach and inclusion. Moreover, during the COVID-19 pandemic and, later, a laboratory shift to Stanford, she provided much-needed stability, sheltering me and the rest of her lab from the chaos that can often result from such perturbations. For these reasons, and many more, Beverley has been a wonderful advisor. I am exceedingly fortunate to be a member of her academic lineage.

I am also indebted to the members of my dissertation committee: John Dabiri, Steven Brunton, and Morteza Gharib. Aside from the keen insights which have shaped my work and character as a researcher, these men have each stepped out of their way to include me in their respective spheres of the scientific community. I am sincerely grateful for their wisdom and hospitality; I hope to embody the generous and kind spirit they have shown me in my own career.

I thank the current and former members of the McKeon group for their engagement in my work and, especially, for their enduring friendship. The postdocs and senior members of the group—Anagha Madhusudanan, Tomek Jarosalwski, Maysam Shamai, Morgan Hooper, Ben Barthel, Salvador Gomez, Yuting Huang, and Jacque Tawney—were instrumental in getting me up to speed in the happenings of the McKeon group. Morgan, Maysam, and Jacque, in particular, graciously oriented me in the NOAH laboratory where most of my experiments were conducted. The friendship of Greg Stroot and Miles Chan, my immediate classmates, was greatly impactful to my graduate experience. Federico Rios-Tascon and Miya Coimbra came later to the team, but were close companions nonetheless. I give special thanks to Yuting Huang who remained at Caltech as co-final McKeon Group member on campus with me. Between all members, I am thankful for the problems solved around a whiteboard and the long lunches at Red Door spent pondering life's mysteries.

Of course, this work would not be where it is today without the engagement of our collaborators. In particular, I thank Leonardo Alves and Davi Bernhard de Souza (Universidade Federal Fluminense), Pavlos Vlachos and Brett Meyers (Purdue University), and Steven Brunton with his lab (for hosting me at the University of Washington). Speaking of which, I am grateful for the funding supplied by the Army Research Office under grant W911NF-17-1-0306 and by the Office of Naval Research under grant N00014-17-1-3022.

I am also grateful to many of Caltech's faculty for valuable insights and for instilling in me a passion for knowledge. First I thank Anthony Leonard who, though emeritus, contributed his time, interest, and insight to the work of the McKeon group. I also thank John Sader, Jane Bae, and Joe Shepherd from GALCIT for their contributions to my work and for their various encouragements. Joanna Austin, the option representative at this time, has been helpful in managing graduate progress; I thank her as well. Because my work heavily relied on concepts from applied math and computer science, I often pursued coursework and advice from professors in those departments. Adam Wierman, Eric Mazumdar, Konstantin Zuev, and Georgia Gkioxari all contributed to my project direction and academic formation.

Many other members of GALCIT contributed to the quality of my work. From Mory Gharib's group, I thank Scott Bollt, Sean Devey, and Alexander Rusch. From John Dabiri's lab, I thank Nathan Wei, Peter Gunnarson, Nina Mohebbi, and Simon Anuszczyk. I also thank Rahul Arun and Micah Nishi-

moto for their valuable insights. Discussions with these peers and others were instrumental in the formation of my work. I am fortunate to conduct research in such an encouraging and collaborative environment.

Truly, none of this work would have been possible without the constant support of Liza Bradulina, who was the McKeon group administrative liaison during my time in GALCIT. Not only did she ensure that all of my administrative needs were handled without a hitch, but she also made sure that I was involved in scientific outreach as much as (and sometimes more than) was possible. Her passion for exposing local youth to science is admirable and infectious.

I also owe gratitude to many other members of Caltech staff and administration. I thank the GALCIT administrators who often assisted my journey through the program. Christine Ramirez, Jamie Meighen-Sei, Sarah Pontes, Paula Mark, and Karima Guthrie were always a joy to interact with. I also thank the GALCIT machinists Ali Kiani and Brandon Nilles for contributing their expertise to my experiments. I wish to acknowledge Mary Herrera, Kitty Calahan, Lance Hayashida, and the others who helped organize and facilitate my outreach lecture with Caltech Science Journeys, which was one of the highlights of my time at Caltech. I also thank Brian Brophy for his guidance in the art of scientific presentation and for his general interest in my wellbeing over the last few years. Finally, I thank GALCIT's excellent custodians Barry Reed and Glenda Cisneros. These people afforded me the ability to focus on technical work without worrying about anything else, which is a rare blessing.

Scholars, scientists, and engineers external to Caltech have also had considerable influence on my academic development and interests. First, I owe my fascination with fluid mechanics to Jonathan Naughton, who advised me during a Master's degree at the University of Wyoming prior to matriculation at Caltech. I remember the time in his laboratory with great fondness. I also acknowledge my colleagues from Wyoming—Pourya Nikoueeyan, Arash Hassanzadeh, Chris Rumple, Gideon Baldwin, Yi Han, and others—to whom I owe much of my knowledge of the practical aspects of research in fluid mechanics. I also acknowledge Cameron Wright, who helped show me the joys of data analysis through image processing, and Francis Su, whose generous friendship and deep reflections on the virtues of mathematics have been formative to my scientific worldview.

I owe a debt of gratitude to David Mukai, who provided the primary inspiration to pursue a PhD. Through shared interests in faith, engineering, and wrestling, David has been a constant friend, mentor, and advisor since the very beginning of my academic journey. Through countless hours spent in his office over coffee, David showed me the influence that a professor can have on his students. His example has helped me to calibrate my own ambitions as a researcher, showing me the importance of self-sacrifice for lasting impact.

Through various challenges, I have found that shared trials lead to lasting friendships. That has certainly been my experience in GALCIT. I am incredibly thankful for my first-year cohort, especially: Brayden Aller, John Pederson, Miles Chan, Niyati Desai, Sorina Lupu, Peter Gunnarson, Nikhil Ranganathan, Sahangi Dassanayaki, Ying Luo, Jimmy Ragan, and Donner Schoeffler. I also acknowledge Liam Heidt and Quentin Chevalier as members of this crew, though they were not strictly in the cohort. These friendships were forged by many long nights spent grappling with difficult concepts in the first-year office (or on Zoom, during the pandemic). I am certain that many of them will endure long beyond our time in Pasadena. I am additionally thankful for the friendship of other members of the Caltech community, including that of Joel and Angela Lawson, Wesley Yu, Noel Esparza-Duran, Harsha Reddy, Utku Hatipoglu, Mert Yuksel, Thomas Henning, and others.

Various other communities in Pasadena have been invaluable to my experience there. I begin by thanking my friends from Caltech's Graduate Christian Fellowship (GCF): Nathan Wei, Albert Wandui, Widi Moestopo, Rachel Gehlhar Humann, Maria Camarca, Hannah Manetsch, Andy Ylitalo, and others. Gathering to discuss theology and philosophy was a refreshing addition to the largely technical content consumed at Caltech.

I am also thankful for the members of our home Bible study which has formed in recent years: John Pederson, Brayden Aller, Kate Gieschen, Salvador Gomez (and occasionally Heather Ruiz), Tinashe Handina, and Will Dembski. I am frequently astounded by the depths of wisdom these men and women possess; our discussions shared collectively over Margherita Monday pizzas or individually over coffee have been incredibly edifying. They have been instrumental to my understanding of life, faith, and work.

Members of the grappling community have embraced me (often literally) during my time in Pasadena. From Titan Mercury Wrestling Club, I thank Morris Bird, Jordin Humphrey, and Lee Schneiderman; from Pasadena Fight Academy, I thank Savant Young, Jason Parry, and the other coaches and committed athletes at the gym. It may seem disjointed to acknowledge the grappling community in a dissertation, but wrestling is, in a way, therapeutic to me. Headlocks and choke-holds help me to breathe freely when the stranglehold of technical material constricts.

My local church community at Grace Pasadena has been a fount of continual encouragement. Numerous members of the congregation have invested their lives in mine. First, I thank the leadership of the church for creating such a community-oriented environment: Ben Ewen, Marc Choi, Caroline Huxell, John Stone, and Brannin and Tanya Pitre. I also thank the members of our Wednesday night City Group—Ben and Whitney Ewen, Anders and Ashley Corey, Zach and Rachel Lee, Dennis and Nancy Finnerman, Connor Otheim, Mark and Elaine Reid, Tim Hawkinson and Patty Wickman, Bryan Estreller, Francis Su and Natalie Wang, Matthew and Christine Natividad, and many others—who have been consistent companions through life's various fluctuations. Whether at the commune, Bungalow Heaven, or the Finn Inn, these meetings have been a steady anchor point in my weekly rhythms. Members of the men's group have also been a great support. Including some already mentioned, I thank Josh Brake, Zeke Zechman, Dave MacFadzean, Gene Fahnestock, Noah Sierrota, and others for their consistent prayer, encouragement, and advice throughout the trials inherent to a PhD. There are many more to acknowledge from Grace, and I could not name them all, but of the rest, I highlight Caroline Paulles, Ben and Meredith Barnes, Abbey Brake, Nate and Emily Hsieh, and Erissa Stilley who have provided encouragement and inspiration in various capacities.

I am grateful to many other close friends who have supported me during my studies. First, I am grateful to Scott Habermehl, who was my all-star roommate during the first two years at Caltech. Our relationship was forged in the crucible of a tiny apartment during two years of a global pandemic. Under such circumstances, the dross is skimmed away, and a pure friendship remains. I also thank my dear friends from Wyoming, Ian and Tylynn Hall, Jordan Kiler, Abby Lund, Tonya Busse, James and Rachel Musselman, Shane, Claudia, and Harper Woods, and Kyle and Megan Pope for their constant companionship despite our geographical diaspora across North America. I am thankful, too, for the local friendships gladly inherited from my wife. Blake and Caroline Huxell, Kenny and Erin Reyes, and Aaron and Casey Zwingli-Thomas have been thoughtful and supportive beyond expectations. The advice, encouragement, and friendship received from these men and women has been invaluable.

Having spent much of this thesis engaged with the concepts of dynamical systems, the importance of initial conditions on my life's trajectory is starkly evident. Any progress I have made can be traced back to my loving family. I am immensely grateful to my parents, Adley and Jenifer, for their compassionate, supportive, and enduring presence in my life. When I showed interest in science as a child, they fostered it in every way that they could. Whether by throwing NASA themed birthday parties, sending me to summer school at the museum, or buying me books on all things science and engineering, they attempted to satisfy my insatiable curiosity. Though they no longer supply my literature review, I still deeply sense their supportive and loving presence. Their influence is indellibly marked on my character. I am also incredibly thankful for my brother Mason who has become a role model in many ways. I admire his integrity and his creative approach to leadership, and am grateful to have him as a close confidant in life's trials—as the proverb states, "a brother is born for adversity." Our family's value of education goes back generations: I thank Paul and Shirley Harms, Joyce Moore, and James McCaughan for instilling in us the desire to learn. I am also deeply grateful to my wife's family—John and Jeniffer Miller; Jeff, Muriel, Lily, Everett, and Cannon Miller; and Tyler, Joanna, Cahyra, and Tobias Studley—for adopting me into their family and my interests as their own. I am immeasurably grateful for the uplifting influence of my family. The freedom which comes from their support allows me to fly.

Finally, I thank my lovely wife, Jessica Harms, for her constant and profound companionship. This thesis would not have been completed if not for her steadfast support. Jess has been by my side through every moment of the Ph.D. journey. When I was burdened by frustration and disappointment, she was the first to comfort me; when I experienced triumphs, she led the celebrations. Jess thoughtfully listened to every scientific presentation that I prepared (often multiple times) and, though not trained in the mathematical sciences, could tell you all about the research contained in these pages. Her love and patience have motivated me not only to be a better researcher, but also to be a better man.

Above all, I thank my Lord and Savior Jesus Christ for the grace and wisdom He has afforded me in the pursuit of knowledge. In those moments when I feel that the work of my hands is as striving after wind, I am awed by the beauty of creation and am reminded of His great love for us.

> *When I consider your heavens,*
> *the work of your fingers...*
> *LORD, our Lord,*
> *how majestic is your name in all the earth!*
>
> — Psalm 8:3,9

ABSTRACT

Modern flow measurement technology enables studies of fluid motion that, half a century ago, would have seemed unfathomable. However, despite staggering capabilities, measuring many natural flows in the field remains challenging. In particular, resolving coherent flow structures within physical scales ranging from meters to kilometers is not readily achieved. This dissertation proposes autonomous mobile flow field measurements (AMFM) as a paradigm for expanding flow field measurement capabilities into this range of scales. In the AMFM framework, a mobile platform such as a drone would identify critical flow structures and follow them autonomously as they evolve; the device would be taught, in a sense, to chase after the wind for the sake of measuring it. The greatest theoretical challenge to AMFM is that of flow structure detection: what, after all, should be identified in the flow? How is it to be measured? Answering these questions is the overarching motivation of this dissertation. In response, two principal contributions are developed. The first is a theoretical approach to gradient estimation labeled Lagrangian gradient regression (LGR), which enables instantaneous and finite-time flow gradients to be approximated from sparse flow observations. The second is a semantic approach to flow measurement, which provides the ability to discern fluid motion from complex natural images using arbitrarily defined flow tracers. Together, these tools enable a range of studies which would be difficult to conduct otherwise. To demonstrate their combined ability, two experiments are performed. The first examines the motion of imperfect surface tracers measured by the proposed methods relative to sub-surface flows measured by conventional techniques. The second experiment analyzes flow features in the Caltech turtle ponds using only tracers naturally occurring on its surface. While it is demonstrated that the methods and results obtained in this work are meritorious in their own right, they also provide a framework from which future AMFM technologies can be built.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Estimating dynamic flow features in groups of tracked objects,” *Under Review*, 2024. DOI: `10.48550/arXiv.2408.16190`. [Online]. Available: `https://arxiv.org/abs/2408.16190`,
T.D.H. contributed in the development of the semantic kinematic measurement framework, collection of training data, detection model finetuning, tracer tracking implemention, performance of physical experiment, data processing, and writing the manuscript. S.L.B. contributed to the conceptualization of the study, interpretation of the results, and editing of the manuscript. B.J.M. contributed to the conceptualization of the study, funding acquisition, overseeing of physical experiments, interpretation of the results, and editing of the manuscript.

[2] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Lagrangian gradient regression for the detection of coherent structures from sparse trajectory data,” *Royal Society Journal of Open Science*, 2024. DOI: `10.48550/arXiv.2310.10994`. [Online]. Available: `https://arxiv.org/abs/2310.10994`,
T.D.H. made primary contributions in developing the theory of Lagrangian gradient regression, implementing the theory in software, performing numerical experiments, analyzing results, and writing the manuscript. S.L.B. contributed to the conceptualization of the study, theoretical guidance, interpretation of the results, and feedback and editing of the manuscript. B.J.M. contributed to the conceptualization of the study, funding acquisition, theoretical guidance, and editing of the manuscript.

[3] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Direct computation of velocity gradients from particle trajectories,” *15th International Symposium on Particle Image Velocimetry*, no. 1, Jul. 2023. [Online]. Available: `http://hdl.handle.net/20.500.12680/7p88cp96d`,
T.D.H. performed numerical and physical experiments, developed algorithms and implemented them in software, prepared the data, and contributed to the writing of the manuscript. S.L.B. contritubuted to the conceptualization of the study, interpretation of the results, and editing of the manuscript. B.J.M contributed to conceptualization of the study, funding acquisition, overseeing of physical experiments, and editing of the manuscript.

The above articles co-authored by Tanner Harms contain material that is adapted for selected chapters in this dissertation. The work by Harms et al. (2023) [1] has been modified for inclusion in Chapter 4, results from Harms et al. (2024a) [2] have been included in Chapters 3 and 4, and components of Harms et al. (2024b) [3] have been prepared for Chapters 5 and 7. A software package providing general implementation capabilities for Lagrangian gradient regression has also been published [4], and is publicly available at https://doi.org/10.5281/zenodo.13126619.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

> *Measure what is measureable, and make measureable what is not so.*
>
> — Galileo Galilei

From the earliest days of recorded thought, the dynamics of fluid motion has captivated the human mind. The intricate beauty of its eddies and currents has inspired poets and artists across centuries to compose masterpieces that flow like the waters constructing their metaphors. Its mystery and perpetuity have formed the basis of philosophies and systems of thought; indeed, three of the four primeval elements—air, water, and fire—exist within the categories of fluid mechanics. Even modern intellects marvel at the turbulent dance of tongues of flame and the powerful cascades that form as waters flow to the sea.

Understanding and harnessing fluids has been the task of scientists and engineers for millennia. The fruit of their research has benefited humanity greatly. Today, filling a glass of water at home seems mundane; yet, the Romans achieved a miracle in their aqueducts, fountains, and bathhouses. Commuters may grumble at the discomfort of cramped seats in economy class; yet when Orville Wright's feet left the ground at Kitty Hawk in 1903, all of mankind took flight with him. One must not look far to find the benefits which society has reaped from the study of fluids. Whether by engines and turbines, air conditioning and plumbing, rockets and airplanes, or a freshly brewed cup of coffee in the morning, humanity thrives on the fruit of fluid mechanics.

Unlocking the technological potential of fluid motion is driven by the ability to measure it. Advances in measurement almost always bring about advances in scientific understanding and engineering capability—the Wright brothers would not have been able to fly without the force measurements made through wind tunnel experiments. Today, many aspects of fluids, and fluid motion in particular, can be measured. By measuring pressure differences on specifically oriented sensors or the thermal cooling due to flow across a heated wire, velocities can be inferred with high accuracy at a point in space. In the latter part of the twentieth century, advances in computing made it possible to extend flow measurements beyond a single point to an entire region of fluid, effectively rendering the invisible quantifiable. The ability to measure flow fields has since extended to encompass many physical scales important to science and engineering through carefully designed laboratory experiments or through expensive remote measurement technologies like satellite altimetry and radar.

There remains, however, a need to measure flows in circumstances in between the laboratory and satellite extremes. Applications including wind energy, pollution management, physical oceanography, and aerodynamics around large vehicles would all benefit from the ability to measure flows in their real context and with imperfect flow tracers. That challenge represents the guiding motivation of this thesis. Its objective is to make progress in extending fluid measurement technologies to scales and conditions where measurements are presently limited.

1.1 Probing the Measurable Scales of Fluid Motion

Modern technology enables the measurement of resolved flow fields at both small and large scales. Figure 1.1 illustrates these capabilities by displaying flow fields measured in recent experiments alongside the physical dimensions of the measured flow in meters. In Figure 1.1a, the flow field around a Zebrafish larva is displayed [5] and flow structures on the order of millimeters are visible. Flows with features even smaller than this can now be measured. An experimental technique known as micro-PIV images through a microscope to capture structures on the order of 10^{-6}m [6].

The largest scales of flow field measurements using techniques from experimental fluid mechanics tend to be on the order of 10^0 to 10^1 meters. Figure 1.1b displays results of an aerodynamic study of the flow around a professional cyclist [7]. Other large-scale aerodynamic studies include those performed on applications such as a full-scale free flying helicopter in ground effect [8] and vertical-axis wind turbines (VAWTs) [9]. However, imitations on facility size and hardware capabilities make it challenging to perform experiments larger than these.

By very different measurement modalities, it is also possible to observe flow fields at geophysical length scales due to advances in flow sensing technologies originating from physical oceanography and meteorology. Modern satellite altimetry, for example, is able to measure ocean velocities with spatial resolution of 15-30 km [11], and techniques from data assimilation enable the

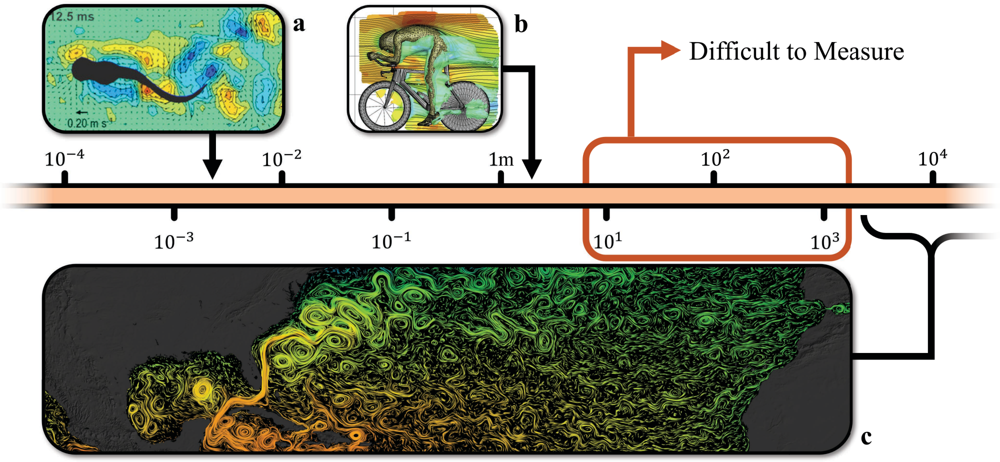


Figure 1.1: Considering the length scales of measurable flow fields. (a) Results from a study performed on a Zebrafish larva, where flow features are on the order of millimeters (Müller et al. [5], reproduced with permission from The Company of Biologists Publication Agreement through CCC.). (b) Results from an aerodynamic study on a full-sized cyclist model , where measurements are on the order of meters (Jux et al. [7], reproduced with permission under CC–BY 4.0.). (c) Ocean surface velocities visualized by the NASA Scientific Visualization Studio [10], where measurements are on the order of kilometers and greater. Flow fields on intermediate scales ranging tens of meters to kilometers currently face significant measurement challenges.

public distribution of global velocity fields at resolution finer than 10 km [12]. An example of this type of data is presented in Figure 1.1c, where a graphic created by NASA Scientific Visualization Studio [10] displays currents in the Atlantic Ocean. Other measurement platforms such as ocean drifters [13], Lagrangian weather balloons [14], weather doppler radar [15], and LiDAR [16], [17] are able measure velocity fields at large physical scales, but become limited at scales smaller than a kilometer due to factors such as technological limitations, measurement sparsity, and atmospheric conditions.

Making robust measurements of flow fields on the scale of $10^0 - 10^3$ meters using commonly available equipment remains a frontier in flow measurement research. The ability to make such measurements would lend insight into critical applications such as wind energy generation, plastic pollution reduction in global waterways, and remote sensing of rivers and streams, among many others. This thesis aims to expand upon measurement techniques in the domain of experimental fluid mechanics using recent developments in computer science and dynamical systems analysis. In particular, it seeks to enable flow structure detection schemes amenable to autonomous mobile flow field measurements (AMFM), an experimental framework which shall be developed and described shortly. Before discussing details of AMFM, however, it is first necessary to briefly cover the basics of flow measurement approaches originating from an experimental fluids perspective.

1.2 Conventional Flow Field Measurement Techniques

Consider, for a moment, that it is necessary to measure the flow around the cylinder indicated in Figure 1.2. To fully examine the flow, it is desirable to construct an experiment that allows for the flow field to be instantaneously observed many diameters of the cylinder downstream and with sufficient breadth in the cross-stream direction. In certain circumstances, it may be possible to design such an experiment. For instance, if the cylinder is relatively small, it could be configured in a laboratory facility and measured over a domain much larger than the object itself.

Many flow measurement applications, however, do not permit observation the flow field over the entire region of interest. In such instances, parts of the flow might be observed in the lab-fixed Eulerian frame over a smaller region of the flow, or in the particle-centric Lagrangian frame through passive tracers.

The Eulerian and Lagrangian Frames

The Eulerian description of flows considers fluid as it moves past a fixed point or a fixed region in space. As time passes, fluid moves through the specified domain while the coordinate frame of reference stays the same. Eulerian analyses are concerned with field variables defined over the entire region. For example a flow in the Eulerian context may be described by the velocity field $\mathbf{v}(\mathbf{x}, t)$, where $\mathbf{x}$ are spatial coordinates that remain unmoved in time and t is a parameterization for time. In Figure 1.2, both the ideal field-of-view and the practical one represent Eulerian domains through which the fluid flows. Clearly, the Eulerian frame is favorable for many laboratory flows since experimental apparatus are often bulky and immobile; the fluid runs through an experimental domain carefully constructed by the researchers.

The alternative approach is to consider fluid motion from the frame of reference of fluid parcels advecting with the flow. This is known as the Lagrangian frame, and is represented in Figure 1.2 by the trajectories of tracers shown traveling downstream. Rather than describing the fluid's behavior by

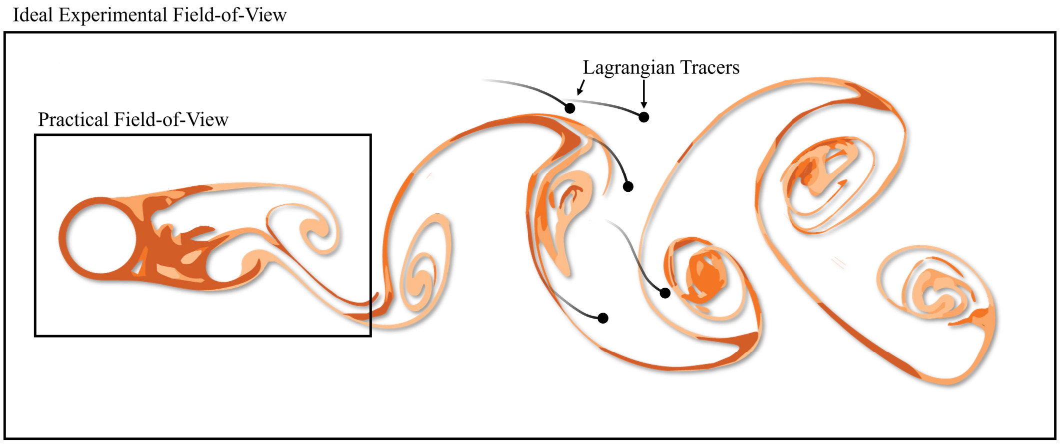


Figure 1.2: Measuring a flow with a fixed field of view. The canonical flow around a cylinder from left to right is depicted in orange as the characteristic Von Karman vortex street. The large box encompassing the flow indicates an ideal field-of-view for an experiment on the flow, while the smaller box indicates what may be practically achievable. Black markers and their tails indicate Lagrangian tracers which may also be used to study the flow.

field quantities, it is defined according to the position histories of the analyzed parcels. Thus, in the Lagrangian frame, the flow is described by the trajectory function $\mathbf{x}(t; t_0, \mathbf{x}_0)$ which indicates a particle's position at time t with respect to initial position $\mathbf{x_0}$ and initial time t. Lagrangian measurements are common in oceanographic applications, where ocean drifters are released and their positions monitored over time. Both the Lagrangian and Eulerian descriptions of a flow are relevant to understanding the basics of experimental flow field measurement techniques.

Velocimetry in Experimental Fluid Mechanics

The task of experimental fluid mechanics is to observe and quantify the motion of matter which is, by common inspection, invisible. In almost all instances, this is achieved by modifying the characteristics of the fluid in such a way as to make it observable under certain circumstances. While various modifications exist, by far the most common is to add numerous observable particles to the fluid and illuminate them in the experimental region of interest. If the response time of the particle is small relative to that of the flow, then it is assumed that the observable tracers can be reliably used to estimate flow velocities (This is usually characterized by the Stokes number, which is defined as $Stk = \frac{t_0 u_0}{l_0}$, where t_0 represents the relaxation time of the particle, u_0 is the

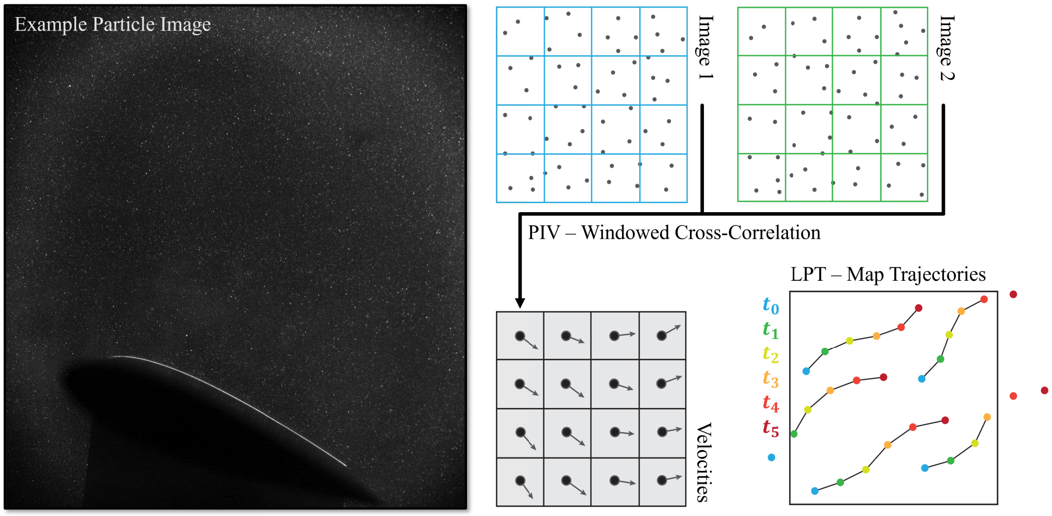


Figure 1.3: Schematics illustrating flow-field estimation approaches commonly employed in experimental fluid mechanics. Virtually all experimental flow field measurements begin with particle images such as seen on the left. PIV involves performing cross-correlation between subsequent frames inside of small windows tesselated over the domain. LPT tracks particle trajectories and computes velocities based on individual particle displacements.

free-stream fluid velocity of the fluid, and l_0 is the the characteristic length-scale of the flow, for example, the diameter of an obstacle. Particles trace the flow well when $Stk \ll 1$.). An example of the particle images which result from this process is presented on the left-hand side of figure 1.3.

Flow visualization by particle images, however, is only the first step of the velocimetry process. Various algorithms have been composed to quantify the observed motion of tracers through an experiment. Particle image velocimetry (PIV, [18]) has traditionally been the most commonly used approach, although, recently, computational and algorithmic advances have made Lagrangian particle tracking (LPT, [19], also denoted Particle tracking velocimetry (PTV)) an attractive alternative. These algorithms are here briefly described.

Particle image velocimetry (or, more specifically, digital particle image velocimetry) was developed at the end of the twentieth century as a computationally efficient means of extracting velocity information from particle images [20]. It is based on the principal of spatial signal correlation, which can be used to find the location of strongest similarity between two images. PIV utilizes correlation by separating the full-sized images into many small windows within which correlations are performed between subsequent frames.

The peak of the correlation within each window represents the position of greatest similarity of the window contents at frame two to the contents at frame one. It indicates the aggregate displacement of illuminated particles over the time step between frames, and can thus be used to estimate the average velocity inside the window. A schematic of this process is presented in Figure 1.3.

While PIV has evolved into a very effective and heavily used algorithm in the fluids community [18], [21], its mode of operation limits the physical scales of the flow which it is able to measure. The effectiveness of PIV depends on being able to sense the motion of recorded particles. As the field-of-view is expanded for a single camera, the motion of particles between frames may drop below pixel-scale and less-brightly illuminated particles may become difficult to distinguish. This leads to a tradeoff between measurable area and measurement accuracy. To overcome this limitation, studies often stitch together the field-of-view of multiple cameras, which is costly, and can be computationally burdensome.

Lagrangian particle tracking also began development towards the end of the twentieth century [22]–[24], but did not experience the same community adoption as PIV due to algorithmic concerns. Where PIV estimates velocity on a grid due to the aggregate motion of windowed particles, LPT seeks to estimate velocity locally at particle positions by mapping their trajectories over time. The algorithmic process involves detecting particle positions, stitching together the trajectories that they form over subsequent frames, computing trajectory-wise velocities, and interpolating to a uniform grid at each time step. This process was limited to sparse distributions of particles until only recently, when algorithmic advances enabled LPT to be performed on the dense particle distributions commonly used with PIV [19], [25]. A simple visualization of the LPT process is also presented in Figure 1.3.

LPT is often advantageous over PIV for 3D measurements and for measurements made with low particle density. However, 2D flow measurements can be challenging using LPT, as particles will often move out of the plane of illumination thereby truncating its recorded trajectory. Another disadvantage of LPT is the computational complexity of modern algorithms. Performing analysis on LPT data can take far longer than for a similar PIV study due to the multiple complex steps in the algorithm.

Both PIV and LPT are limited in their requirement of clean particle images to be effective. The addition of non-flow features in the images can quickly reduce the quality of PIV and LPT measurements. As a result, experimental fluids studies often go to great lengths to ensure a dark background in the experimental field-of-view, and multiple pre-processing steps are typically applied to data to improve quality. As flow measurement technologies are extended to larger scales, it will become necessary to engage with images containing many features that are not relevant to the flow. This is a challenge that will be addressed in this work.

Velocimetry in Computer Vision

Camera-based velocimetry has grown in global relevance in the second and third decades of the twenty-first century due to the proliferation of computer vision and autonomy applications. The ability to estimate the velocity of objects in a scene is crucial for any technology that involves the autonomous navigation of an agent. Self-driving vehicles are a high-profile case in point, where understanding the motion of surrounding objects, vehicles, and pedestrians is safety-critical.

Surprisingly, there has been little cross-talk between the experimental fluids and computer vision communities regarding motion estimation. As a result, the algorithms embraced by the machine perception community are largely different from those commonly used in fluid mechanics. Typically, they fall into the category of optical flow (OF).

Optical flow methods were effectively conceived in the 1940's with the work of psychologist James Gibson [26], who was interested in understanding motion as a piece of a theory on neurological perception of the visual world. Gibson's research laid the groundwork for understanding pixel intensity as a conservative quantity in the perception of motion. For example, one may watch a video of a train moving across a screen and perceive its motion even though the pixels on the screen remain stationary. These psychological concepts were eventually codified algorithmically by researchers in the early 1980's [27], [28]. Since then, many related algorithms based on the same principal have come to be known as OF. In fact, PIV is sometimes considered to be a type of OF algorithm. Practically speaking, however, OF is often viewed as distinct due to the breadth of alternative methods and the widely varying applications.

One of the advantages of OF is its ability to compute dense velocity estimation at each pixel in the image. As a result, fluids researchers have recently developed approaches which fine-tune coarse PIV flow estimation with OF [29]. OF, however, is not particularly well suited to analyze flow data by itself. When applied directly to particle image data, the concentration of brightness at the particle locations inhibits accurate full-field velocity estimation.

1.3 Autonomous Mobile Flow Field Measurements

All of the flow measurement techniques described in the previous section are most naturally executed in the Eulerian frame. Whether in the field or in the laboratory, cameras are almost always placed with a fixed field-of-view relative to the flow, making it possible only to observe fluid behavior within a stationary window. Even LPT studies, which assume a Lagrangian description of the flow through the measurement of tracer trajectories, are limited to a static domain and therefore often interpolate flow fields to a uniform Eulerian grid before performing analyses.

Being constrained to a fixed field-of-view poses severe limitations on the type of studies which can be conducted on advection-dominated flows. This is especially true when observing larger-scale flows in the field. In particular, the ability to analyze the behavior of flow structures as they evolve downstream is limited by the capabilities of the imaging hardware and the capacity to seed measurable tracers. In an attempt to address these challenges, this work introduces the notion of autonomous mobile flow field measurements (AMFM). In the AMFM paradigm, which is illustrated in Figure 1.4, flow structures of interest are identified using available tracers, which may be artificially seeded or natural, and followed autonomously by some mobile platform such as an unmanned aerial vehicle (UAV). In this way, the evolution of flows can be understood over larger spatio-temporal domains within a frame of reference intrinsic to the flow itself.

Similar Flow Field Measurements in Experimental Literature

Measurement apparatuses with a dynamic field of view have been used to measure fluid behavior in a variety of contexts. A common example fixes the optical system to a body that is in motion within the fluid. Studies involving cameras mounted to tow-tank carts (e.g., Beal et al. [30]) fit into this category, as do studies where the field-of-view is affixed to a rotating body, which is often considered in experiments on rotorcraft (e.g., Moaven et al. [31]). Even though the field-of-view dynamically traverses the flow in these cases, it is assumed to

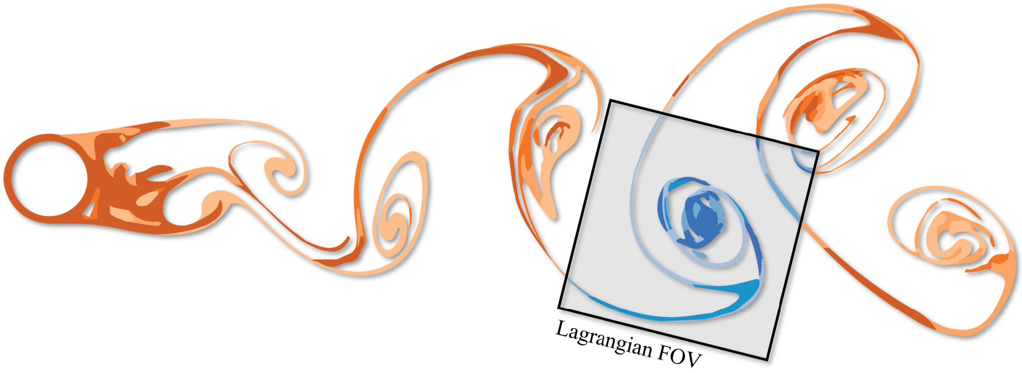


Figure 1.4: An illustration of the autonomous flow field measurement concept. A flow structure of interest (in the figure, a vortex) is identified and followed by a dynamic observer.

be stationary with respect to an immersed body of interest. Other experiments approximate a body-fixed frame of reference by sampling over a larger domain in space and transforming the flow field measurements' coordinates in post-processing [32].

Some studies have been designed with observers that travel with flow features without the point of reference of an immersed body. Examples of so-called Flying-PIV have been implemented in various studies to examine flows that evolve over spatial domains. The work by Zheng and Longmire [33] explores the streamwise development of turbulent vortex packets induced in a boundary layer by cylindrical roughness elements by traversing the optical setup downstream at a fixed rate. The study by Koehler et al. [34] examined flow in an internal combustion cylinder at a fixed distance from the piston over its cycle by traversing the laser sheet with the piston head. These studies are both examples of where the observer moves with nontrivial prescribed dynamics relative to the flow.

Outside of the laboratory, researchers have begun utilizing unmanned systems for flow measurement. Unmanned aerial vehicles (UAVs) in particular show promise for the measurement of mid-range scales of fluid motion in the field. For example, the work by Tauro et al. [35] computes averaged flow measurements on a small stream using a drone in hover. Measurements are taken using natural and artificial flow tracers over multiple repetitions and averaged to reproduce the flow field over a few meters. Other studies such as that by Pinton et al. [36] have measured flow behavior by observing dye added to the water in natural flows. Unmanned systems have become relevant in ocean applications as well, where underwater gliders have been developed to autonomously study ocean characteristics beneath the sea surface [37], [38]. Gliders, however, are similar to ocean drifters [13] in that they do not necessarily provide spatially resolved information about flow fields but rather ocean data along a single trajectory.

Components of AMFM

Autonomous mobile flow field measurements differ from the above studies in that they seek to use flow field information to autonomously navigate the observer in response to the flow. In general, this involves three stages in a cycle:

Detection: The first stage in the AMFM loop is to identify the flow features of interest that are to be tracked and followed. These features could be vortices, transport barriers, or coherent sets of tracers, among others—the particular detection objective depends on the application. Critically, it should be possible to identify the features regardless of the motion of the observer.

Estimation: The estimation stage in the AMFM cycle involves predicting the future state evolution of the detected feature. Using the observation history of the feature in cadence with the surrounding flow field information, it is possible to predict subsequent positions in the feature trajectory. Data assimilation methods such as Kalman Filters and Particle Filters are well suited for this task and already a highly developed technology.

Control: The final stage in the AMFM process is to manipulate the trajectory of the observer so that it follows that of the detected feature. The exact nature of this stage depends on the form of the measurement apparatus. For example, implementation will differ between an observer mounted on a cyber-physical system in a laboratory to one mounted on a drone flying above a river.

Of the three stages of AMFM, structure detection presents the greatest research opportunity. Both state estimation and control in autonomous applications are relatively well understood and are more specific to the particular applications. Therefore, the focus of this thesis is to develop detection strategies that are amenable with the AMFM concept.

1.4 Addressing Challenges in Structure Detection in AMFM

The overarching objective of this thesis is to address principal flow structure detection challenges that arise during the AMFM process. In particular, three prominent theoretical and technical challenges are considered:

The Challenge of Relative Motion: Normally, when optically measuring a flow field, it is assumed that the observer is stationary and fixed with respect to the flow. That assumption must be discarded in the context of AMFM, where the observer's frame of reference is expected to change over time. This induces relative motion between the flow and the observer which is not generally known and which influences the observed flow field. In order to meaningfully detect flow structures with a dynamic observer, the influence relative motion must be addressed.

The Challenge of Feature-Rich Images: Recorded images of flows taken outside of the controlled laboratory environment will typically contain various features which complicate flow field analysis. Rather than working with carefully orchestrated particle images, real-world features such as reflections, terrain, and non-flow areas will occupy many parts of the images. Without careful pre- and post-processing of the data, these features will inhibit the calculation of necessary quantities such as gradients, thereby preventing their use for structure detection purposes. Before AMFM applications in the field are possible, an approach to dealing with such images needs to be developed.

The Challenge of Imperfect Tracers: When making field measurements, the available tracers (whether natural or artificial) will not likely be perfect indicators of the underlying flow. Rather, they will exhibit finite mass, buoyancy, and non-spherical geometry, which will make them respond differently to the flow than an ideal tracer. Understanding the relationship of the tracers to the flow is a challenge that must be considered when attempting AMFM applications.

This list of potential challenges is by no means exhaustive; AMFM presents an audacious attempt to drive forward the state of flow measurement technology. Nevertheless, addressing these issues would pave the foundation for future autonomous flow field measurements. Perhaps, as the technology is developed, chasing after the wind will become more than a mere exercise in futility.

Structure of the Dissertation

The remainder of this thesis is devoted to addressing the three challenges listed above. Chapter 2 provides the theoretical background necessary for making objective analyses of flow data that are invariant to change of frame. This theory is derived from the axioms of Lagrangian coherent structure (LCS) analysis, and poses a new difficulty which must be addressed—namely, how does one identify structures objectively using only limited observations of flow tracers. In response to this new problem, Chapter 3 introduces Lagrangian gradient regression (LGR) as the principal theoretical development of this work. Chapter 4 is then devoted to validating LGR by testing it on sparse analytical data sets. In Chapter 5, the difficulties posed by feature-rich images are confronted. Tools from modern computer vision are implemented to enable robust flow feature detection given semantic information embedded within natural images of flows. Chapter 6 applies the tools developed in Chapter 5 to examine the efficacy of naturally occurring flow tracers for structure detection. Finally, the dissertation concludes in Chapther 7 by synthesizing the insights and methods throughout to study flows measured in the field at Caltech's turtle ponds.

Chapter 2

BACKGROUND THEORY: OBJECTIVE FLOW MEASUREMENTS

In the introduction to this dissertation, the framework of autonomous mobile flow field measurements (AMFM) was proposed as a means for studying increasingly large scale flows in natural contexts. Three challenges to structure detection, it was suggested, must be overcome for AMFM systems to operate. The first of these was the challenge of relative motion. Before further development can ensue, this obstacle must be addressed. This chapter presents the theoretical background required to gain traction towards a solution.

The context of AMFM demands that the relative motion of an observer to a flow be considered in all calculations of flow field quantities. Consider the example presented in Figure 2.1 where the observations of a stationary and a dynamic observer are compared for a generic two-dimensional flow field. The top row of the figure represents the flow as if viewed from a laboratory-fixed point of view. In this case, the relative rotation rate $\dot{\mathbf{q}}$ between the observer and the flow is set to 0 and the observer velocity vector $\dot{\mathbf{p}} = \mathbf{0}$. The bottom row depicts the flow as if viewed from a dynamic observer with non-zero rotation rate and translation. Arbitrarily, the rotation rate is set to $\dot{\mathbf{q}} = 3$ radians per unit time and the translation rate is given as $\dot{\mathbf{p}} = [2,\, -1]^\top$ units space per unit time. As indicated in the figure, the first column displays the horizontal velocity u, the second column displays the vertical velocity v, and the third column displays the vorticity $\omega_z = \dfrac{\partial v}{\partial x} - \dfrac{\partial u}{\partial y}$.

The message of Figure 2.1 is that many common flow measurements are dramatically affected by the relationship of the observer to the flow. In the given example, the stationary velocity fields are drastically different than the dynamic ones. Even the observed vorticity, which depends on spatial variations in the flow field, is modified due to the relative rotation. This can cause a range of practical challenges as one attempts to measure and interact with flows beyond the confines of a controlled laboratory. It is especially challenging in the context of AMFM, where observer mobility is intrinsic to the application. How might one address the issues posed by a dynamic observer? Three potential solutions might be considered.

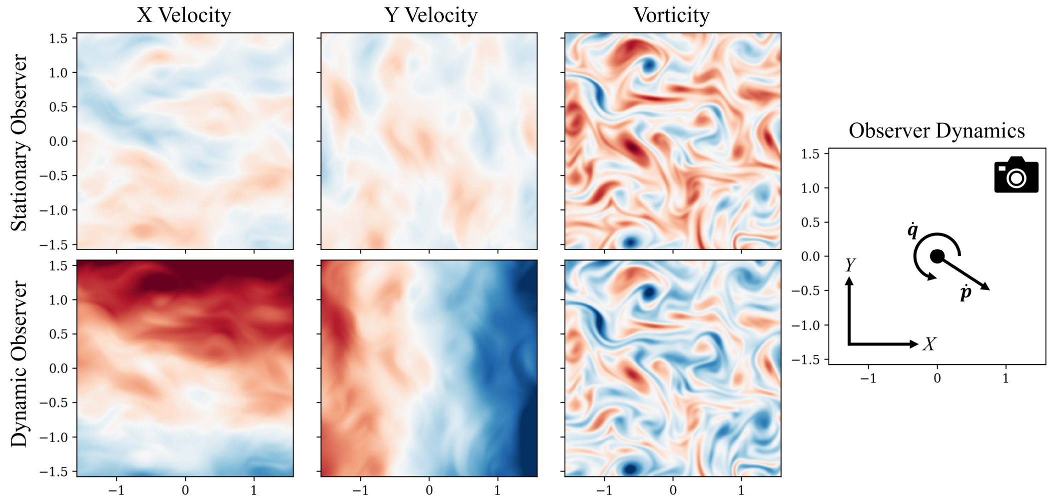


Figure 2.1: Demonstration of the effects of observer motion on measured flow field quantities. The top row presents an arbitrary flow field as if viewed from a stationary observer fixed to the frame of reference of the flow such that $\dot{\mathbf{q}} = 0$ and $\dot{\mathbf{p}} = \mathbf{0}$. The bottom row represents the same flow as observed by a dynamic observer with relative rotation $\dot{\mathbf{q}} = 3$ radians/unit time and $\dot{\mathbf{p}} = [2, -1]^\top$ units space/unit time. In all frames, red indicates positive values and blue negative. It is clear that the observer motion has a strong influence on how a flow is understood.

1. **Ignore relative motion effects.** While certainly the most naive approach, ignoring the effects of relative motion is practically simple. Under certain circumstances it may be appropriate to assume that the relative motion is negligible or that it does not significantly influence the quantities of interest, thus allowing the system to make measurements in the traditional manner without correction. However, in the context of AMFM, this is not generally the case. AMFM seeks to identify and follow specific flow features whose definition depends on the relative motion between the observer and the flow. Following a vortex, for instance, depends on the ability to consistently identify the vortex core and boundaries. However, as was evident in the previous example, while the material motion in the flow is unaffected by the observer, its representation can be drastically influenced by relative motion. Thus, in AMFM measurements, relative motion must not be ignored.

2. **Incorporate a known observer trajectory.** Perhaps the most intuitive approach to correcting for relative motion is to subtract the velocity of the observer from the velocities recorded prior to performing any

analyses. As discussed in the previous chapter, this is already done in practice when velocity fields are *a-posteriori* transformed to the motion of a body in frame. It is also an attractive solution in the laboratory context where the motion of a gantry or a robotic manipulator is precisely known for each experimental recording. However, access to reliable observer trajectories is limited in field experiments. If, for example, a drone were used to measure flow information, its trajectory would be constructed from accelerometer and GPS sources which are notoriously noisy. Therefore, correcting observed velocities using observer trajectories is not an acceptable solution.

3. **Perform objective analyses.** A third alternative is to compute quantities that are invariant under frame of reference changes. While limiting the available analyses that can be done online in the AMFM loop, computing invariant quantities ensures that the flow is objectively considered regardless of the relative motion of the system. This is the approach that is pursued throughout this work. In the remainder of this chapter, the theoretical underpinnings are discussed.

2.1 Lagrangian Coherent Structures

While experimental flows are typically measured in the fixed-frame Eulerian context, a Lagrangian perspective can be helpful when seeking objective flow measurements. The theory of Lagrangian coherent structures (LCS. See Haller [39], Hadjighasem et al. [40], and Allshouse and Peacock [41], for example) provides methods which characterize a flow's behavior according to the collective motion of tracers embedded within it.

Many analyses have been developed since the inception of LCS as a field of research, but almost all of them share two fundamental qualities: they are finite-time and objective. Finite-time deals with the time interval of analysis. Due to their Lagrangian foundations, LCS analyses are derived from the trajectories of tracers in a flow. This implies that any LCS analysis accounts for fluid motion over a finite duration rather than its instantaneous behavior. Eulerian quantities such as vorticity, for example, provide instantaneous information about fluid behavior, but do not lend insight far into the future or past in unsteady flows.

Objectivity

The principle of objectivity is closely linked to the concept of the material-frame invariance from continuum mechanics [42], [43]. The primary statement of objectivity is that the position of an observer should not influence any quantities of interest. An objective quantity is one that is intrinsic to the observed system; its representation is not dependent on how it is measured.

Consider a time-varying vector $\mathbf{x}(t) \in \mathbb{R}^d$ that is subjected to a Euclidean transformation of the form

$$\mathbf{y} = \mathbf{Q}(t)\mathbf{x} + \mathbf{p}(t), \tag{2.1}$$

where $\mathbf{Q}(t) \in \mathbb{R}^{d \times d}$ is a proper orthogonal rotation tensor and $\mathbf{p}(t) \in \mathbb{R}^d$ is a translation. Any objective scalar, vector, or tensor quantity must be invariant under transformations of the form of equation 2.1.

Most common flow measurements are not objective. For instance, it is clear by intuition and by Figure 2.1 that velocity is not objective. Mathematically it can be shown by taking the time derivative of equation 2.1:

$$\tilde{\mathbf{v}} = \dot{\mathbf{y}} = \dot{\mathbf{Q}}\mathbf{x} + \mathbf{Q}\dot{\mathbf{v}} + \dot{\mathbf{p}},$$

where $\tilde{\cdot}$ represents the transformed frame and $\dot{\cdot}$ indicates the time rate-of-change. This transformation law for velocity expresses precisely how the motion of an observer influences the measured velocity. Similar transformation laws can be found for many common flow metrics including vorticity, Q-criterion [44], λ_2-criterion [45], and the Okubo-Weiss parameter [46], [47]. None of these metrics are objective, and therefore would yield misleading information from a dynamic observer. Appendix A contains more details regarding the objectivity of various flow quantities.

Paradigms of LCS Detection

Until now, the notion of a Lagrangian coherent structure has been left ambiguous. This is because a variety of paradigms exist for detecting LCS, each of which admit a different mathematical formalism. Broadly speaking, LCS are regions of a flow that drive material behavior over a specified observation interval. Depending on the approach, these regions may be defined as material surfaces (curves in 2-D space, for example) or as material volumes (some enclosed region in the flow space), and they could represent physical flow characteristics or the probabilistic behavior of observed tracers. In their survey of LCS methods, Allshouse and Peackock categorize analyses as either dense or sparse depending on the quantity of tracer data required to perform the computations [41]. These paradigms are briefly considered below.

Dense Detection Approaches

Dense approaches require a large number of tracers relative to the spatial extent of flow features in their analyses, and are often categorized as either geometric (sometime referred to as gradient-based approaches [39], [48]) or probabilistic [49], [50]. The geometric approach bases its analyses off of flow gradients, and therefore detects features with specific dynamical interpretation. For instance, the finite-time Lyapunov exponent (FTLE, [48]) and the Lagrangian-averaged vorticity deviation (LAVD, [51])—both common geometric LCS analyses—physically represent linear deformation rate and amount of flow rotation over time, respectively. Alternatively, the probabilistic approach to LCS is grounded on the Perron-Frobenius (or transfer) operator [52] and defines structures as regions exhibiting coherent behavior with high probability. While probabilistic measures will often highlight the same spatial domain as corresponding geometric analyses (e.g., Finite-time entropy [50] is a probabilistic proxy for FTLE and highlights the same material features), the field values are probabilistic in nature and are therefore not physically interpretable in the same way as geometric analyses. Both geometric and probabilistic methods are deterministic in the sense that they provide repeatable results when applied to the same flow.

Sparse Detection Approaches

Dense approaches can be difficult to use with practical data sets since they require particle densities greater than what is typically achievable in the field or in the lab. To overcome this limitation, many variations of sparse approaches (i.e., requiring few particles) for LCS detection have been proposed. These include methods based on spectral clustering [53], fuzzy C-means [54], graph coloring [55], [56], finite-element approximation of the dynamic Laplacian [57], [58], network analysis [59], and DBSCAN [60], among others. Sparse LCS tools are able to provide meaningful results with far fewer particles than dense alternatives, but they often lack determinism and do not typically provide quantifiable information of the flow kinematics. Often, users of sparse clustering methods must specify the number of structures to identify *a-priori*, implying prior understanding of the flow. So, while LCS may be identified with limited trajectories via sparse methods, their interpretation requires knowledge of the flow.

2.2 Gradient-Based LCS

Because of their direct relationship to flow dynamics, gradient-based (or geometric) LCS are the focus of the methods developed in this work. Geometric LCS were introduced at the turn of the century [61], [62] and have been consistently developed since. Based on principles from continuum mechanics and dynamical systems theory, geometric LCS seek to identify material surfaces of dynamical significance in a flow over some interval [63]. In the decades since the field's inception, many analyses have been introduced. For a comprehensive treatment, interested readers should refer to the text book by Haller [43] or to the seminal review by the same author [39].

Types of Gradient-Based LCS

All geometric LCS can be classified into one of three categories: hyperbolic (defined as either attracting or repelling manifolds), parabolic (shear-driven manifolds thought of as Lagrangian jet cores), or elliptic (thought of as Lagrangian vortices). The three categories of structures can be visualized in Figure 2.2. While strict definitions of each type of structure are more technical than this work demands, brief, qualitative descriptions are provided below.

Hyperbolic structures represent material surfaces characterized by locally normal flow motion. Repelling hyperbolic surfaces are ones in which particles with initial conditions near the feature rapidly separate from each other over time. Attractive hyperbolic surfaces defined over the same time domain are those which accumulate particles during the flow motion. Figure 2.2 indicates repelling and attracting LCS as orange and blue curves, respectively. Hyperbolic LCS are closely linked to material mixing. Because they represent locally repelling surfaces, fluid will not cross a hyperbolic LCS over the prescribed time interval [48]. The typical analysis used to highlight hyperbolic features is the finite-time Lyapunov exponent (FTLE, [48], [62]), which will be defined in greater detail in later sections.

Parabolic LCS represent Lagrangian jet-cores, or material surfaces dominated by tangential stretching. In other words, they tend to emphasize regions

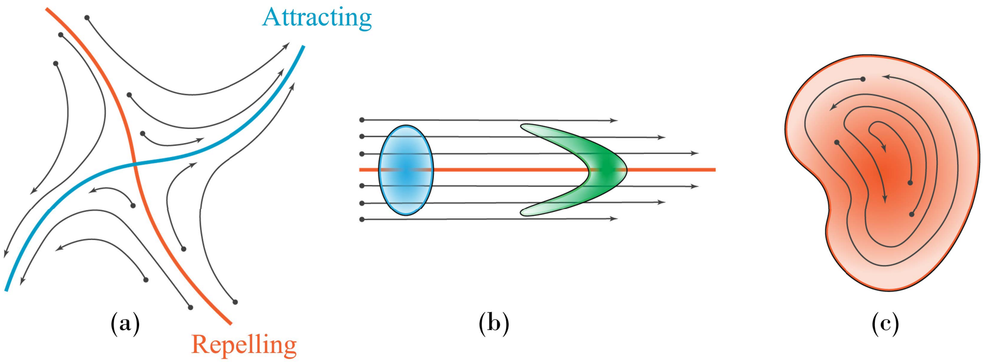


Figure 2.2: Types of LCS as defined by the theory of Geometric LCS. (a) Hyperbolic structures, which are characterized as material surfaces defined by tracer movement normal to the surface. Particles are repulsed from repelling structures and drawn towards attracting surfaces. (b) Parabolic structures, which are defined by stretching due to shear. Can be thought of as Lagrangian jet cores. (c) Elliptic structures, which enclose regions of coherent rotation. Thought of as Lagrangian vortices.

of large shear in a flow, as indicated in figure 2.2. Because parabolic structures and hyperbolic structures are both defined according to local material stretching, it is sometimes challenging to disambiguate the two in LCS an analyses. For instance, ridges in the FTLE field of a flow can be associated with either parabolic or hyperbolic features. However, work has been done to analytically separate the two [39], [63].

Finally, elliptic LCS are defined as material surfaces that enclose regions of coherent rotation [64]. While strictly defined as closed codimension-1 manifolds in the space of the flow, it is convenient to think of elliptic LCS as the entire volume enclosed by the bounding surfaces. As will be shown later, the motion inside of the elliptic boundaries influences the detectability of the boundaries themselves, making them useful targets for AMFM measurements. Some metrics used to identify elliptic LCS include the polar rotation angle (PRA, [65]) and the Lagrangian-averaged vorticity deviation (LAVD, [51]).

Applications of Gradient-Based LCS

Since the inception of the field, LCS analyses have been applied to a broad array of technical problems. Perhaps the most common use of LCS analysis is in the study of geophysical and atmospheric flow (e.g., [66]–[72], among others). They have also been useful in characterizing structures in turbulent and

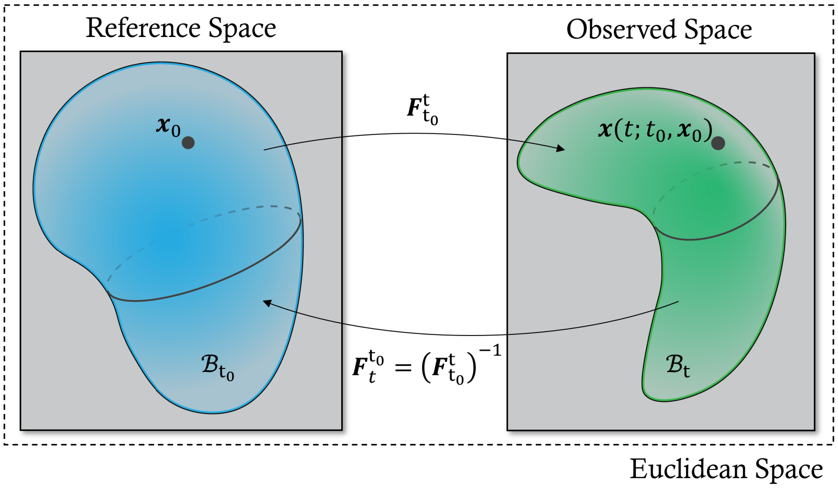


Figure 2.3: Schematic of a material deformation. A continuous region of fluid $\mathcal{B}_{t_0}$ is defined in a reference space and measured in the observed space as a deformed region $\mathcal{B}_t$. The smooth mapping from reference to observation $\mathbf{F}_{t_0}^{t}$ is known as the the flow map, and has an inverse from observation to reference. The mapping of individual particles $\mathbf{x}_0$ within $\mathcal{B}_{t_0}$ to the observed space form trajectories $\mathbf{x}(t; t_0, \mathbf{x}_0)$

unsteady flows of aerodynamic interest [73]–[76] and in biomedical flows [77]–[83]. LCS have even been applied to dynamical systems that are not fluids-specific, such as in the work by Lekien and Marsden [60], [84], and by Husic et al. [56].

2.3 Theoretical Foundations

Having understood the fundamental concepts of gradient-based LCS analyses, it is necessary to define the mathematical framework from which the theoretical contributions of this work are built. In the context of autonomous mobile flow field measurements, both velocity gradients (instantaneous) and flow-map Jacobians (finte-time) will be necessary. This section provides a foundation for the algorithms and theory that will be developed in later chapters.

The Flow Map

The theoretical foundations for making objective flow measurements stem from the concept of the flow map. Consider the schematic presented in Figure 2.3. Here, the deformation over time of a region of flow material is depicted. Specifically, an enclosed region of material $\mathcal{B}_{t_0} \subseteq D \subseteq \mathbb{R}^d$ defined as a subset of the flow domain D in Euclidean space at an arbitrary reference time $t_0 \in \mathbb{R}$ is deformed by the flow to an enclosed spatial region $\mathcal{B}_t \subseteq D \subseteq \mathbb{R}^d$ at some other observed time $t \in \mathbb{R}$. While the dimensionality of the flow d is general in this formulation, for the fluid flows of interest it is restricted to $d = \{2,\, 3\}$.

The motion of all fluid elements governed by the analyzed flow are solutions of the differential equation

$$\frac{d}{dt}\mathbf{x} = \dot{\mathbf{x}} = \mathbf{v}(\mathbf{x}, t), \tag{2.2}$$

where $\mathbf{x} = \mathbf{x}(t) \in \mathbb{R}^d$ is a vector valued function of time representing the position of a material element and $\mathbf{v}(\mathbf{x}, t) : D \to D$ is a smooth vector field describing the rate rate of change of elements in the domain D. The trajectory of a material element, or tracer, is defined by its motion

$$\mathbf{x}(t; t_0, \mathbf{x}_0) = \mathbf{x}_0 + \int_{t_0}^{t} \mathbf{v}(\mathbf{x}(\tau; t_0, \mathbf{x}_0), \tau) d\tau, \tag{2.3}$$

for all times $t \in \mathbb{R}$ from initial time t_0 and position $\mathbf{x}_0 = \mathbf{x}(t_0)$. The tuple $(\mathbf{x}_0,\, t_0)$ is referred to as the initial conditions of the trajectory.

The mapping of a material element from the reference time t_0 to the observed time t is defined by the flow map

$$\mathbf{F}_{t_0}^{t}(\mathbf{x}_0) : D \to D \tag{2.4a}$$

$$\mathbf{x}_0 \mapsto \mathbf{F}_{t_0}^{t}(\mathbf{x}_0) = \mathbf{x}(t; t_0, \mathbf{x}_0), \tag{2.4b}$$

which is also smooth in D. In other words, the position of a tracer at time t is described through the flow map as $\mathbf{x}(t) = \mathbf{F}_{t_0}^{t}(\mathbf{x}_0)$. Then, since the motion of the tracers are defined by the velocity field according to equation 2.3, the flow map satisfies the relationship

$$\frac{d}{dt}\left(\mathbf{F}_{t_0}^{t}(\mathbf{x}_0)\right)\Bigg|_{t=\tau} = \mathbf{v}(\mathbf{F}_{t_0}^{\tau}(\mathbf{x}_0), \tau). \tag{2.5}$$

Thus, the instantaneous rate of deformation of a material region $\mathcal{B}_t$ is governed by the velocity field $\mathbf{v}(\mathbf{x}, t)$ over the domain D.

Importantly, the flow map is a diffeomorphism, and therefore admits an inverse [85]

$$\mathbf{x}_0 = (\mathbf{F}_{t_0}^{t})^{-1}(\mathbf{x}(t)) = \mathbf{F}_{t}^{t_0}(\mathbf{x}(t)). \tag{2.6}$$

This ensures that no two reference points may occupy the same observed point at a given time. Moreover, it makes the arrow of time arbitrary in the sense that the flow map can be defined forwards in time or backwards in time without encountering mathematical barriers. Other useful properties of the flow map stem from the local existence and uniqueness of solutions for initial value problems

$$\mathbf{F}_{t_0}^{t_0}(\mathbf{x}_0) = \mathbf{x}_0 \tag{2.7a}$$

$$\mathbf{F}_{t_0}^{t+s}(\mathbf{x}_0) = \mathbf{F}_{s}^{t+s} \circ \mathbf{F}_{t_0}^{s}(\mathbf{x}_0) = \mathbf{F}_{t}^{t+s} \circ \mathbf{F}_{t_0}^{t}(\mathbf{x}_0) \tag{2.7b}$$

[48]. The first of these properties states that the mapping of a tracer position from one time to the same time is simply the same tracer position. The second states that mappings over adjacent time intervals can be merged to represent the mapping over the entire interval. Thus, the flow map over the full time domain $[t_0,\, t]$ can be constructed as the composition of many intermediate flow maps—a property that will become important later on.

The Flow Map Jacobian

The gradient of the flow map $\mathbf{F}_{t_0}^{t}(\mathbf{x}_0)$ is known as the flow map Jacobian. It represents the spatial variation of material deformations induced by the flow over the observed time interval, and is defined

$$D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0) = \nabla_{\mathbf{x}_0}\mathbf{F}_{t_0}^{t}(\mathbf{x}_0), \qquad DF_{ij}(t; t_0, \mathbf{x}_0) = \frac{\partial x_i(t; t_0, \mathbf{x}_0)}{\partial x_{0,j}},\ \forall i, j \in [1, ..., d]. \tag{2.8}$$

In fact, the flow map Jacobian is also known as the deformation gradient tensor in the field of continuum mechanics [42].

The flow map Jacobian describes the deformation of tracers within an ϵ-neighborhood of the particle $\mathbf{x}_0$. Consider the perturbation $\mathbf{y_0} = \mathbf{x}_0 + \Delta\mathbf{x}_0$, for $\Delta\mathbf{x}_0$ very small. Performing a Taylor expansion of the perturbation centered at $\mathbf{x}_0$ yields

$$\Delta\mathbf{x} = \mathbf{F}_{t_0}^{t}(\mathbf{y}_0) - \mathbf{F}_{t_0}^{t}(\mathbf{x}_0) = D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0)\Delta\mathbf{x}_0 + \mathcal{O}(\|\Delta\mathbf{x}_0\|^2), \tag{2.9}$$

with the norm being Euclidean. As will be seen in the next chapter, equation 2.9 bears significance when estimating flow gradients. Practically speaking it states that, when the perturbation size $\|\Delta\mathbf{x}_0\|$ becomes large, the nonlinearities in the flow map dominate the deformation of local tracers at the future time. Therefore, dense tracer distributions are typically required when computing flow gradients. Overcoming this challenge is the subject of Chapter 3.

For notational convenience, the argument of the tensors will be dropped unless it is ambiguous to do so. Hence, $D\mathbf{F}_{t_0}^{t} \equiv D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0)$, and so on. Throughout this work, other tensors will be introduced, and it will be assumed that the argument is the particle position at the time indicated in the subscript unless otherwise stated.

Connection to The Velocity Gradient Tensor

Where the flow map Jacobian represents the spatial variation of deformations over a finite time, the velocity gradient represents the spatial variation of deformation rate at an instant. Naturally, the two are related. Consider the temporal evolution of the flow map Jacobian:

$$\frac{d}{dt} D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0) = \frac{d}{dt}\left(\frac{\partial}{\partial \mathbf{x}_0}\mathbf{x}(t;t_0,\mathbf{x}_0)\right). \tag{2.10}$$

Exchanging derivatives and applying the chain rule yields

$$\frac{d}{dt} D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0) = \nabla\mathbf{v}(\mathbf{x},t) D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0), \tag{2.11}$$

where $D\mathbf{F}_{t_0}^{t_0}(\mathbf{x}_0) = \mathbf{I}_d$, $\nabla\mathbf{v}(\mathbf{x}(t),t)$ is the velocity gradient tensor located at the particle position $\mathbf{x}$ at time t, and $\mathbf{I}_d$ is the identity tensor in the space of the flow. Thus, the velocity gradient acts as the evolution of the flow map Jacobian along a trajectory [86].

Recognizing the intrinsic link between the velocity gradient and the flow map Jacobian is essential to the methods that are developed in Chapter 3 and implemented in Chapters 4 and 5. It represents a connection between the Lagrangian and the Eulerian frames through the relationship

$$\nabla\mathbf{v}(\mathbf{x}(t),t) = \frac{d}{dt} D\mathbf{F}_{t_0}^{t}(\mathbf{x}_0)\, D\mathbf{F}_{t}^{t_0}(\mathbf{x}(t)), \tag{2.12}$$

of which the implementation using experimental data is addressed in Chapter 3.

2.4 Gradient-Based Flow Metrics

Gradients are the basis for a large proportion of prominent flow diagnostics. Many of these will be implemented throughout this work and it is therefore necessary to introduce them here. The relevant methods are categorized as instantaneous (i.e., based on the velocity gradient tensor) and finite-time (i.e., based on the flow map Jacobian).

Instantaneous Flow Metrics

The velocity gradient tensor can be split into a rotational component and a dilatational component through the decomposition

$$\nabla \mathbf{v} = \mathbf{W} + \mathbf{D}. \tag{2.13}$$

The anti-symmetric spin tensor $\mathbf{W} = \frac{1}{2}(\nabla \mathbf{v} - \nabla \mathbf{v}^\top)$ represents the rate of change of material rotation, and the symmetric rate of deformation tensor $\mathbf{D} = \frac{1}{2}(\nabla \mathbf{v} + \nabla \mathbf{v}^\top)$ represents the rate of stretching in a material element.

Many common flow diagnostics are linked to the stretch and spin tensors. In three dimensions, for example, the rate of fluid rotation is characterized by the vorticity ω, which is defined as the unique vector ω satisfying

$$\mathbf{W}\mathbf{e} = -\frac{1}{2}\boldsymbol{\omega} \times \mathbf{e}, \quad \forall \mathbf{e} \in \mathbb{R}^3. \tag{2.14}$$

Additionally, the principal strain rate may be computed as the maximum eigenvalue of the strain rate tensor

$$\epsilon_1 = \lambda_{max}(\mathbf{D}), \tag{2.15}$$

and represents the maximum rate of stretching for a fluid element. Using both $\mathbf{W}$ and $\mathbf{D}$, the Q-criterion is defined according to

$$Q = \frac{1}{2}\left(\|\mathbf{W}\|_F^2 - \|\mathbf{D}\|_F^2\right), \tag{2.16}$$

where $\|\cdot\|_F$ represents the Frobenius norm [44]. Values of Q larger than zero indicate that fluid motion is dominated by rotation; if it is less than zero, it is dominated by dilation.

Of the metrics listed above, only the principal strain rate ϵ_1 is objective. The vorticity and the Q-criterion both depend on the spin tensor $\mathbf{W}$, whose transformation law

$$\tilde{\mathbf{W}} = \mathbf{Q}\mathbf{W}\mathbf{Q}^\top + \boldsymbol{\Omega} \tag{2.17}$$

depends on the rigid-body angular velocity between observers $\boldsymbol{\Omega} = \dot{\mathbf{Q}}\mathbf{Q}^\top$, known as the frame spin, and is therefore not objective. However, Haller et al. [51] demonstrate that the instantaneous vorticity deviation

$$\mathrm{IVD}(\mathbf{x}, t) = |\omega(\mathbf{x}, t) - \overline{\omega}(t)| \tag{2.18}$$

is objective, where $\overline{\omega}(t)$ is the instantaneous spatial mean of the vorticity at time t over the flow domain.

Finite-Time Flow Metrics

The finite-time metrics often employed in the study of geometric LCS are typically derived from the tensors and decompositions of continuum mechanics. Of particular importance is the right Cauchy-Green strain tensor, which is defined from the flow map Jacobian as

$$\mathbf{C}_{t_0}^{t} = \left(D\mathbf{F}_{t_0}^{t}\right)^{\top} D\mathbf{F}_{t_0}^{t}. \tag{2.19}$$

$\mathbf{C}_{t_0}^{t}$ is a Gramian matrix, and therefore has the useful properties of being symmetric and positive semi-definite. Since the continuum assumption of fluids enforces one-to-one behavior of the flow map, it is guaranteed that $\det(D\mathbf{F}_{t_0}^{t}) > 0$, and therefore that $\mathbf{C}_{t_0}^{t}$ is, in fact, positive definite and has d real eigenvalues.

Physically, the right Cauchy-Green strain tensor represents the squared change in local distances due to the flow map $\mathbf{F}_{t_0}^{t}$. Thus, the maximum magnitude of the perturbation at time t is determined by the L^2 norm on $D\mathbf{F}_{t_0}^{t}$

$$\max \|\Delta \mathbf{x}\| = \left\|D\mathbf{F}_{t_0}^{t}\right\|_2 \|\Delta \mathbf{x}_0\| = \sqrt{\lambda_{\max}(\mathbf{C}_{t_0}^{t})}\, \|\Delta \mathbf{x}_0\|\,, \tag{2.20}$$

where $\lambda_{\max}$ represents the largest eigenvalue of $\mathbf{C}_{t_0}^{t}$. Additionally, due to the properties of the L^2 operator norm, $\left\|D\mathbf{F}_{t_0}^{t}\right\|_2$ may be computed as the largest singular value of $D\mathbf{F}_{t_0}^{t}$ and represents the maximal gain induced by the tensor [87].

The scalar value $\sqrt{\lambda_{\max}(\mathbf{C}_{t_0}^{t})}$ may be viewed as an exponential, such that

$$\sqrt{\lambda_{\max}(\mathbf{C}_{t_0}^{t})} = e^{\sigma_{t_0}^{t}|\Delta t|}, \tag{2.21}$$

where $t_0 + \Delta t = t$. The exponent $\sigma_{t_0}^{t}$ is the finite-time Lyapunov exponent (FTLE), which represents the exponential growth rate of a linear deformation over the observation time. Because $\sigma_{t_0}^{t}$ assumes linearity, it can only be reliably evaluated for finite Δt if $\Delta \mathbf{x}$ is infinitesimal, or for finite $\Delta \mathbf{x}$ if Δt is infinitesimal. Therefore, it is typically computed according to

$$\sigma_{t_0}^{t} = \lim_{\Delta \mathbf{x}_0 \to 0} \frac{1}{|\Delta t|} \ln \sqrt{\lambda_{\max}(\mathbf{C}_{t_0}^{t})}. \tag{2.22}$$

The FTLE is one of the first and most consistently used analyses in the study of LCS [39], [48]. Ridges in the FTLE field indicate hyperbolic LCS features which act as barriers to fluid transport. If computations are performed using tracer propagation forwards in time, the ridges represent repelling manifolds from which particles at time t_0 will tend to separate. If the computations are performed backwards in time, the ridges represent attracting manifolds upon which particles beginning their trajectories at time t_0 will accumulate.

Continuum mechanical analysis of rotation often involves the polar decomposition, which states that $D\mathbf{F}_{t_0}^{t}$ has the unique decomposition

$$D\mathbf{F}_{t_0}^{t} = \mathbf{R}_{t_0}^{t}\mathbf{U}_{t_0}^{t}, \tag{2.23}$$

such that the rotation tensor $\mathbf{R}_{t_0}^{t}$ is proper orthogonal and the right stretch tensor $\mathbf{U}_{t_0}^{t}$ is symmetric, positive definite [42]. $\mathbf{R}_{t_0}^{t}$ represents a solid body rotation of a material element over the interval, and $\mathbf{U}_{t_0}^{t}$ represents its stretching.

The rotation tensor $\mathbf{R}_{t_0}^{t}$ admits a polar rotation angle $\theta_{t_0}^{t}$, which has been used for estimation of polar LCS [65]. As discussed by Haller et al. [51], however, $\mathbf{R}_{t_0}^{t}$ and $\theta_{t_0}^{t}$ are not dynamically consistent. That is, in general,

$$\mathbf{R}_{t_0}^{t} \neq \mathbf{R}_{s}^{t}\mathbf{R}_{t_0}^{s} \tag{2.24a}$$

$$\theta_{t_0}^{t} \neq \theta_{s}^{t} + \theta_{t_0}^{s}. \tag{2.24b}$$

for two connected intervals $[t_0,\, s]$ and $[s,\, t]$ [88]. This is a result of the fact that $\mathbf{R}_{t_0}^{t}$ represents the best fit rotation tensor in the Frobenius sense of the deformation over the interval $[t_0, t]$. The possible rotation histories which could result in $\mathbf{R}_{t_0}^{t}$ are infinite, so it is not possible to sum the rotations of arbitrary sub-intervals.

The dynamic polar decomposition (DPD) was developed by Haller [88] to overcome these deficiencies of the PRA and others. The DPD splits any tensor defined by a linear process (such as in equation 2.11) into a rotational process with zero rate of strain and an irrotational process with no vorticity. Specifically, the decomposition is defined

$$D\mathbf{F}_{t_0}^{t} = \mathbf{O}_{t_0}^{t}\mathbf{M}_{t_0}^{t}, \tag{2.25}$$

where $\mathbf{O}_{t_0}^{t}$ is the proper orthogonal dynamic rotation tensor and $\mathbf{M}_{t_0}^{t}$ is the right dynamic stretch tensor. From this result, the dynamic rotation tensor $\mathbf{O}_{t_0}^{t}$ can be similarly decomposed into a relative rotation tensor $\mathbf{\Phi}_{t_0}^{t}$ and a mean rotation tensor $\mathbf{\Theta}_{t_0}^{t}$ such that

$$D\mathbf{F}_{t_0}^{t} = \mathbf{\Phi}_{t_0}^{t}\mathbf{\Theta}_{t_0}^{t}\mathbf{M}_{t_0}^{t}. \tag{2.26}$$

The authors show that both $\mathbf{O}_{t_0}^{t}$ and $\mathbf{\Phi}_{t_0}^{t}$ are dynamically consistent and $\mathbf{\Phi}_{t_0}^{t}$ is objective [88].

The framework of the DPD provides a robust basis for computing elliptic flow measures. The first metric considered is the intrinsic rotation angle (IRA)

$$\psi_{t_0}^{t}(\mathbf{x}_0) = \frac{1}{2}\int_{t_0}^{t} |\omega(\mathbf{x}(\tau; t_0, \mathbf{x}_0), \tau) - \overline{\omega}(\tau)| d\tau = \frac{1}{2}\int_{t_0}^{t} IVD(\mathbf{x}, \tau) d\tau, \tag{2.27}$$

where $\overline{\omega}(t)$ represents the spatial average of vorticity. Physically, $\psi_{t_0}^{t}$ represents the angle swept by the relative rotation tensor $\mathbf{\Phi}_{t_0}^{t}$ over the interval. The Lagrangian-averaged vorticity deviation is directly related to $\psi_{t_0}^{t}$ as

$$\mathrm{LAVD}_{t_0}^{t} = 2\psi_{t_0}^{t}. \tag{2.28}$$

The dynamic rotation angle (DRA) $\varphi_{t_0}^{t}$ is defined as the angle generated by the dynamic rotation tensor $\mathbf{O}_{t_0}^{t}$. It represents the amount of rotation that a fluid element experiences relative to the observer, and can be computed

$$\varphi_{t_0}^{t}(\mathbf{x}_0; \mathbf{g}) = -\frac{1}{2}\int_{t_0}^{t} \omega(\mathbf{x}(\tau), \tau) \cdot \mathbf{g}(\mathbf{x}(\tau), \tau) d\tau, \tag{2.29}$$

where $\mathbf{g}$ is an axis family related to the observer around which rotations are measured.

Both $\psi_{t_0}^{t}$ and $\varphi_{t_0}^{t}$ are dynamically consistent and can therefore be summed if computed on adjacent intervals. However, because of the dependence on the observer, $\varphi_{t_0}^{t}$ is not generally objective [88]. In 2-D, if $\varphi_{t_0}^{t}$ is computed using the spatial mean-subtracted vorticity, then it is also objective.

Summary of Flow Quantities

The metrics defined above play a prominent role in the analyses and demonstrations to follow. To keep track of them all, it is convenient to collate them in one place and briefly discuss their properties. Table 2.1 contains useful information regarding the most significant of the quantities introduced above, including mathematical properties, whether it is finite-time, and whether it is objective.

Methods for computing gradients and gradient-based quantities will be addressed in Chapter 3. Even so, it is helpful to visualize many of the metrics summarized in Table 2.1 here to understand their use cases. To this effect, the unsteady double gyre flow is introduced as a demonstration case which will be examined frequently throughout this work. Defined analytically by the velocities

$$u = -\pi A \sin\left(\pi f(x,t)\right) \cos(\pi y), \tag{2.30a}$$

$$v = \pi A \cos\left(\pi f(x,t)\right) \sin(\pi y) \frac{\partial f(x,t)}{\partial x}, \tag{2.30b}$$

where

$$f(x,t) = a(t)x^2 + b(t)x, \qquad a(t) = \epsilon \sin(\alpha t), \qquad b(t) = 1 - 2\epsilon \sin(\alpha t),$$

the unsteady double gyre was designed to study particle transport in Rayleigh-Bénard convection [89], [90]. In later years, it has been used as a baseline case in the study of LCS. Figure 2.4 indicates the behavior of the double gyre flow. As can be seen by the contour lines indicating fixed values of vorticity, the double gyre flow dictates the dynamics of two counter-rotating vortices. The motion of the vortices is determined by the parameters A, ϵ, and α. The parameter A defines the strength of the vortices—larger values mean stronger rotation. The parameters ϵ and α define the oscillation of the centerline between the vortices. Raising ϵ increases the amplitude of the centerline oscillation, and raising α increases the rate of oscillations. Unless otherwise stated, examples using the double gyre employ the parameter set $\{A = 0.1,\, \epsilon = 0.1,\, \alpha = 2\pi/10\}$.

Figure 2.5 displays scalar metric fields for selected instantaneous metrics.

Table 2.1: Summary of important flow quantities that will be relevant to the discussions of subsequent chapters. “FT” stands for finite time, and “Obj.” stands for objective. If a tensor is said to be > 0, that means that it is positive-definite. Tensors in $SO(d)$ are orthogonal with $+1$ determinant. More details on objectivity of quantities can be found in Appendix A.

Quantity	Size, Properties	FT	Obj.	Formula
$\mathbf{v}(\mathbf{x},t)$	$\mathbb{R}^d$	✗	✗	eq. 2.2
$\nabla\mathbf{v}(\mathbf{x},t)$	$\mathbb{R}^{d\times d}$	✗	✗	eq. 2.12
$\mathbf{W}(\mathbf{x},t)$	$\mathbb{R}^{d\times d}$, $\mathbf{W} = -\mathbf{W}^\top$	✗	✓	eq. 2.13
$\mathbf{D}(\mathbf{x},t)$	$\mathbb{R}^{d\times d}$, $\mathbf{D} = \mathbf{D}^\top$	✗	✓	eq. 2.13
$D\mathbf{F}_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}^{d\times d}$	✓	✗	eq. 2.8
$\mathbf{C}_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}^{d\times d}$, $\mathbf{C}_{t_0}^t > 0$, $\mathbf{C}_{t_0}^t = \left(\mathbf{C}_{t_0}^t\right)^\top$	✓	✓	eq. 2.19
$\mathbf{R}_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}^{d\times d}$, $\mathbf{R}_{t_0}^t \in SO(d)$	✓	✗	eq. 2.24a
$\mathbf{U}_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}^{d\times d}$, $\mathbf{U}_{t_0}^t > 0$, $\mathbf{U}_{t_0}^t = \left(\mathbf{U}_{t_0}^t\right)^\top$	✓	✓	eq. 2.24a
$\omega(\mathbf{x},t)$	$\mathbb{R}^{\binom{d}{2}}$	✗	✗	eq. 2.14
$\epsilon_1(\mathbf{x},t)$	$\mathbb{R}$	✗	✓	eq. 2.15
$Q(\mathbf{x},t)$	$\mathbb{R}$	✗	✗	eq. 2.16
$\mathrm{IVD}(\mathbf{x},t)$	$\mathbb{R}$, $\mathrm{IVD} >= 0$	✗	✓	eq. 2.18
$\sigma_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}$	✓	✓	eq. 2.22
$\mathrm{LAVD}_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}$, $\mathrm{LAVD}_{t_0}^t \geq 0$	✓	✓	eq. 2.28
$\psi_{t_0}^t(\mathbf{x}_0)$	$\mathbb{R}$, $\psi_{t_0}^t \geq 0$	✓	✓	eq. 2.27
$\varphi_{t_0}^t(\mathbf{x}_0;\mathbf{g})$	$\mathbb{R}$	✓	✗	eq. 2.29

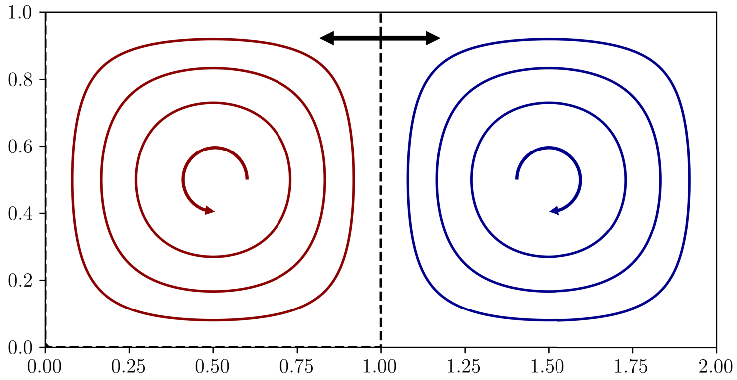


Figure 2.4: Schematic of the unsteady double gyre flow. The flow involves two counter-rotating vortices which fluctuate in size according to a time dependent function. Typically, the interface between the vortices (black dashed line) is defined to oscillate sinusoidally.

Figure 2.5(a) presents the out-of-plane vorticity; 2.5(b) presents the IVD, which is defined in equation 2.18; 2.5(c) displays the principal strain rate ϵ_1 from equation 2.15; and 2.5(d) presents the Q-criterion of equation 2.16. Considering these fields together helps gain intuition for their usage.

Vorticity is a very common metric used in flow-field analysis. As previously discussed, it represents the rate of rotation of a fluid element around an orthogonal axis—in 2-D, as in this case, out of the page. As evinced by figure 2.5, it is the only of the considered instantaneous metrics which indicates the sign of rotation of a flow element. The IVD, on the other hand, indicates the magnitude of rotation, but not the sign. In this case, the magnitude of the IVD is commensurate with that of the vorticity since the spatial average of vorticity over the domain is very close to zero in the double gyre. Although the IVD loses information about the direction of rotation, it has two primary advantages over unaltered vorticity: first, it is objective, where vorticity is not. Second, vorticity is only scalar-valued in two dimensions, whereas the IVD is always a positive scalar. These attributes make it a convenient measurement when analyzing flows in an unknown frame.

Unlike vorticity and IVD, the principal strain rate and the Q-criterion both indicate the amount of stretching occurring in the flow. In fact, the principal strain rate only indicates stretching without being influenced by the flow's rotation, and is therefore objective. The Q-criterion, on the other hand, pits rotation against dilatation, and assigns rotation-dominated regions of the flow a positive value. This is particularly valuable when seeking to identify vortices,

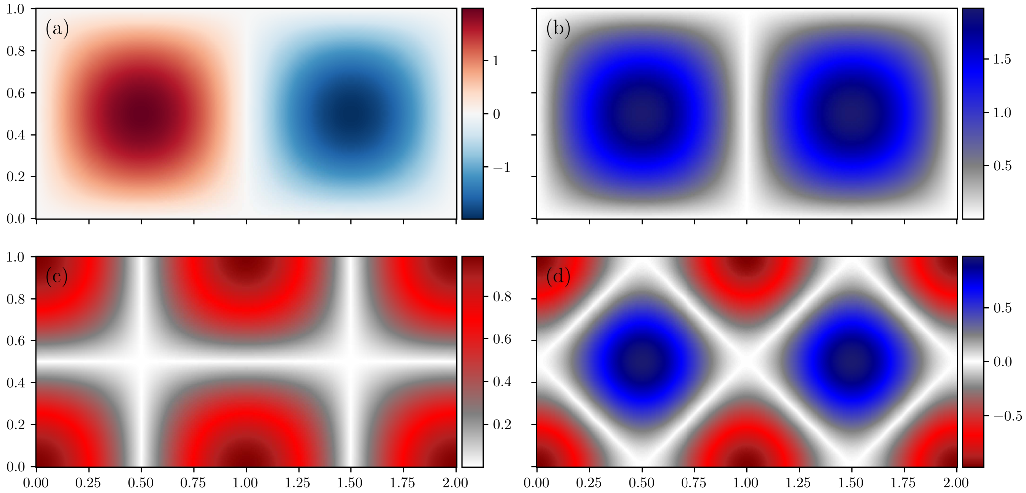


Figure 2.5: Instantaneous flow metrics computed on the double gyre flow at time $t = 0$. (a) Vorticity ω_z. (b) Instantaneous vorticity deviation (IVD) computed according to equation 2.18. (c) Principal strain rate ϵ_1 computed by equation 2.15. (d) Q-criterion computed using equation 2.16.

as any positive region over a threshold can be said to exist in a rotational part of the flow. However, because it depends on the spin tensor, $\mathbf{W}$, it is not objective.

Figure 2.6 presents examples of important finite-time metrics computed on the double gyre flow. Figures 2.6(a) and 2.6(b) display the FTLE field computed forwards and backwards, respectively, over the interval $t \in [0, \pm 15]$ by equation 2.22. Figure 2.6(c) represents the IRA field computed forwards in time over the interval $t \in [0, 15]$ by equation 2.27, and 2.6(d) indicates the DRA field of equation 2.29 over the same duration. Both 2.6(c) and 2.6(d) have been scaled to represent complete rotations of the tracer along its trajectory. Since LAVD is defined as twice the IRA, the IRA will typically be preferred for its convenient physical interpretation.

Forward and backward FTLE as displayed in Figures 2.6(a) and 2.6(b) are possibly the most commonly used finite-time metric. Large values in the FTLE fields represent fluid elements that experience significant stretching over the time domain. Thus, the ridges that appear in the LCS fields are the features of interest in FTLE analyses, as they indicate flow transport barriers. Over the duration of the flow, very little fluid (though, not all: see the flux analysis done by Shadden et al. [48]) will cross these ridges. When performed in forward time, FTLE ridges represent repelling manifolds which drive particles

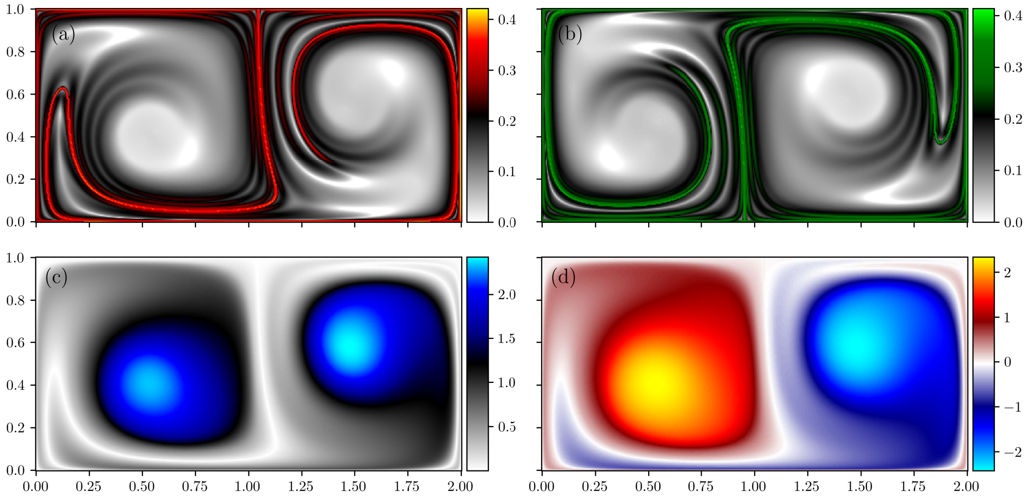


Figure 2.6: Finite-time flow metrics computed on the double gyre flow at time $t_0 = 0$ with $t = t_0 + \Delta t$ and $|\Delta t| = 15$. (a) Forward-time FTLE $\sigma_{t_0}^{t}$ described by equation 2.22. (b) Backward-time FTLE $\sigma_{t_0}^{-t}$ described by equation 2.22. (c) Intrinsic rotation angle $\psi_{t_0}^{t}$ defined by 2.27 scaled to represent complete tracer revolutions. (d) $\varphi_{t_0}^{t}$ defined by 2.29 with similar scaling.

away from themselves. In backwards time, however, they represent attracting manifolds which, in forwards time, tend to accumulate particles from across the domain. The intersections of attracting and repelling ridges represent saddle points in the flow.

While useful for monitoring flow transport barriers, FTLE ridges do not give information about finite-time flow rotation. For that, it is convenient to compute the LAVD (or IRA) and the DRA fields, which are shown for the double gyre in Figures 2.6(c) and 2.6(d), respectively. The LAVD is a scaled sum of the IVD and the DRA is a scaled sum of the vorticity. Thus, they have the same tradeoffs that were discussed regarding vorticity and IVD—namely, that the DRA indicates the direction of rotation but is not, in general objective while the LAVD field is objective but does not indicate rotation direction.

Chapter 3

LAGRANGIAN GRADIENT REGRESSION

In the previous chapter, a framework for making consistent and interpretable flow field measurements under mobile observation conditions was constructed. Rather than ignoring or correcting for the relative velocity between the observer and the flow, tools from dynamical systems theory and continuum mechanics were introduced to define objective metrics which are unaffected by the changing frame of reference. The suggested measurements were classified as either instantaneous (i.e., based on the velocity gradient at a moment in time) or finite-time (i.e., based on the gradient of the mapping from one time to another).

While a variety of viable flow metrics were presented, the approach to computing them has not yet been addressed. That is the objective of this chapter. The following discussion begins with a presentation of the standard approaches for computing velocity gradients and flow map Jacobians. Velocity gradients are typically computed in the study of experimental fluid mechanics and of physical oceanography, though by different means due to varied modality of the collected data. Both paradigms will be introduced in the first section. Next, standard approaches for approximating the flow map Jacobian will be introduced from the theory of Lagrangian coherent structures. The conventional gradient estimation methods, however, are severely limited when only sparse tracer data is available and can be computationally expensive otherwise. While a variety of sparse approaches have been developed, the results are typically unsatisfactory.

All of this sets the scene for the first technical contribution of this work: Lagrangian gradient regression, or LGR. LGR utilizes the data-driven machinery of kernel-weighted regression to approximate the operator driving deformations over short intervals. In the sparse conditions intrinsic to autonomous mobile flow measurements (AMFM), it is critical that the analyzed interval

---

The contents of this chapter have been adapted from Harms, Brunton, and McKeon (2024) [2], which has been accepted to be published by the Royal Society Journal of Open Science under the title *Lagrangian Gradient Regression for the Detection of Coherent Structures from Sparse Trajectory Data* and is available as a preprint on arXiv.

be very short relative to the fastest time-scales of the flow, since deformations remain approximately linear in that regime. While this may seem a stringent condition, it is, in fact, much more lenient than requiring a dense sampling of tracers over the flow domain. Real data can typically be sampled quickly but not with high spatial resolution. (This point is intuitive. In oceanographic studies, retrieving data from ocean drifters can be done many times in a single day, but placing many drifters in close proximity is not cost effective or productive. LPT studies often sample at hundreds of Hertz, but achieving high particle density is technically challenging in many studies.) In the final sections of this chapter, the theoretical framework for LGR will be developed, along with two additional metrics that are convenient to compute by it.

3.1 Methods for Approximating The Velocity Gradient

The velocity gradient is arguably one of the most important quantities in fluid mechanics, and it is therefore one of the most commonly measured. How the velocity gradient is approximated from data depends largely on the type of data that is collected. Here, two paradigms are examined: particle image data collected for use in experimental fluid mechanics, and drifter data collected in physical oceanography. In the former, velocity gradients are universally computed from gridded velocity fields using numerical differentiation. On the other hand, methods from the domain of physical oceanography typically aim to approximate gradient quantities from the Lagrangian motion of the tracers. In the following discussion, prominent gradient approximation methods from both fields are conceptually introduced.

Paradigms in Experimental Fluids

Velocity gradient estimation in the field of experimental fluid mechanics is almost always accomplished through particle image velocimetry (PIV) or through Lagrangian particle tracking (LPT). The pipeline for each of these is sketched in Figure 3.1. The first step in both cases is to carefully acquire particle images of the flow being examined. This is not a trivial step, and often requires expensive equipment, fastidious setup, and somewhat of an artistic touch. The quality of flow field measurements are largely dependent upon the quality of the images collected.

The most common form of experimental flow field estimation is particle image velocimetry (PIV, [18], [21]), which is a windowed, correlation-based

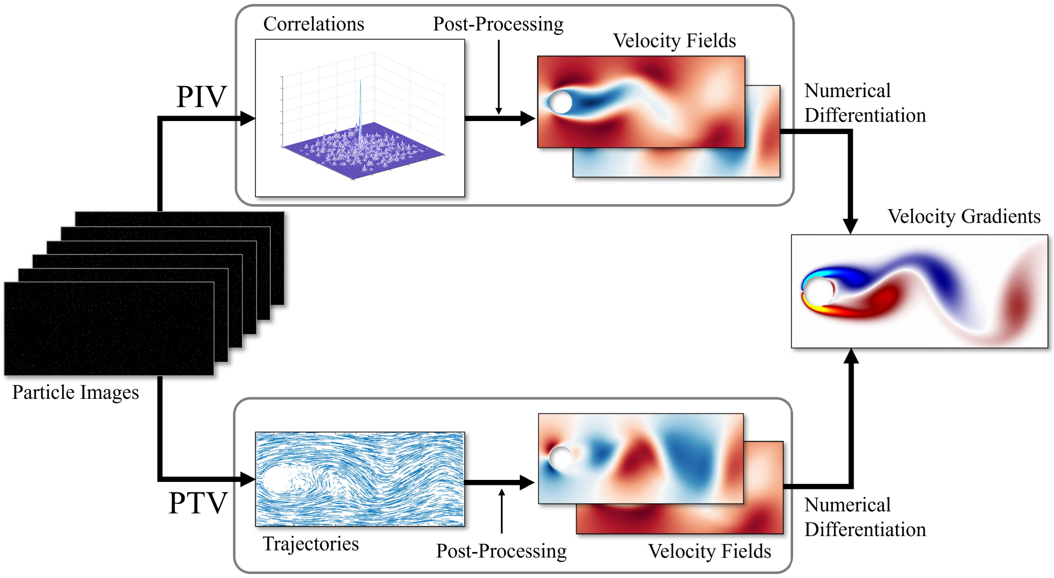


Figure 3.1: Typical velocity gradient estimation pipeline in experimental fluid mechanics. The top branch follows the PIV procedure and the bottom branch follows LPT/PTV methodology.

approach that segments an image into cells and performs signal correlation between frames inside them. The average velocity in each cell is estimated as the location-off-center of the correlation peak divided by the time step between images. Typically, many passes of overlapping windows are performed to increase the resolution of the resulting vector field. Because PIV is correlation-based, it is sensitive to any image content unrelated to the flow. Additionally, it typically does not perform well in regions of high shear due to various effects such as large velocity gradients and out-of-plane motion. These considerations lead to noise in the measured flow fields.

Higher-order methods have also been developed to improve velocimetry results in paradigms similar to PIV. Some of these methods are discussed in a review by Scarano [91]. One such method, named image correlation velocimetry (ICV), was proposed by Tokumaru and Dimotakis (1995) [92]. Instead of realizing displacements by cross-correlation, the Taylor expansion of the Lagrangian deformation of fluid elements was provided as the objective function of a global minimization over image intensities. As a result, higher-order flow properties such as deformation and velocity gradients can be directly computed, albeit at the cost of exponentially increasing computational load with series truncation order. Other methods like the direct measurement of vorticity (DMV) approach of Ruan et al(2001) [93] employ other approaches directly access aspects of velocity gradients and overcome some of the limitations of traditional, cross-correlation-based PIV. Other methods are included in the review by Scarano [91].

Lagrangian particle tracking (LPT, also known as particle tracking velocimetry (PTV); [19], [22], [24]) has recently gained traction in the flow measurement community. While traditional algorithms have struggled to identify trajectories in dense particle images, recent advances have significantly improved performance on dense data and on 3D data sets. Because LPT identifies particle trajectories, it boasts efficient data storage compared to PIV [18] which requires all computed quantities to be stored on a dense grid of pixels or voxels. Additionally, because the velocities computed by LPT are defined point-wise on a particle rather than as the spatial average of all particles in a window they may be more representative of the flow along the particle trajectory.

Prior to computing gradients, velocity information must be transferred to a Cartesian grid. Since PIV is a windowed approach, this constraint is automatically satisfied. LPT data, however, must be interpolated. While standard interpolation is often used in practice, it increases computation time and can incur additional error. As a result, recent studies have developed data-assimilation methodologies which leverage physical principals to achieve fine-scale resolution from LPT tracks. Such methods include FlowFit [94] and VIC+ [95], and their variants. While these methods have been seen to be effective at reconstructing flow fields and, ultimately, gradients, they are known to require significant post-processing effort on top of trajectory identification. They also generate large amounts of data, as all of the measured quantities are stored at every grid cell.

From measured velocity fields, a variety of approaches might be used to estimate velocity gradients. The most basic of these is to compute first-order finite-differences over the field. This, however, amplifies noise in the flow field and, since there are many sources of error in PIV [18] and LPT algorithms [96], the result is often difficult to use for quantitative analyses. To overcome such errors, higher order finite-difference schemes such as Richardson extrapolation might be used. If direct differentiation schemes are insufficient for computing velocity gradients and related quantities, various flow field post-processing techniques may be used. For instance, Gaussian smoothing is often used on noisy velocity fields prior to differentiation and, in flows that exhibit periodicity, phase-averaging can be used to improve gradient calculations. Additionally, other non-derivative quantities such as the Γ_1 and Γ_2 of Graftieaux et al. have been proposed as a proxy for vorticity [97]. All of these methods, however, require Cartesian velocity fields as inputs.

Paradigms in Physical Oceanography

Unlike the convention in experimental fluids, where velocity fields are recorded from image sequences, velocity data in physical oceanography is often recorded via Lagrangian tracers such as ocean drifters (for example, Lumpkin and Centurioni [13]). Thus, the oceanography community has developed a variety of methods for approximating gradients, divergence, and vorticity, some of which have recently been surveyed [98], [99]. In some instances, specific kinematic properties are computed directly. Divergence, for example, is often computed by the change in area of polygons formed by tracers (e.g., Saucier [100]), and vorticity is estimated by evaluating the circulation around the centroid of a polygon (e.g., Kawai [101]). Recent studies, however, have identified the linear least squares approach of Molinari and Kirwan [102] to be a more effective means of estimating ocean velocity gradients [98], [99], [103]. This approach solves separate least squares problems in the latitudinal and longitudinal (x and y) directions using tracer velocities to approximate gradients along each axis at the centroid of the particles used.

3.2 Methods for Approximating The Flow Map Jacobian

Flow map Jacobians are typically approximated by one of two methods: finite-differences or regression. In both cases, tracer positions are required at initial and final times, which implies that trajectories must be known over the entire duration. In this section, a short description of prominent methods is provided here along with illustrations in Figure 3.2.

The finite differences approach to computing $D\mathbf{F}_{t_0}^{t}$ was the first technique to be developed [48], [61] and begins by artificially seeding massless particles throughout the flow domain in a fine mesh. Velocity information for the flow—in the form of an analytical expression or snapshots from simulation or experiment—is then used to propagate the positions of the particles over some interval $[t_0, t]$ to their deformed positions using a numerical integrator. The gradient of the deformation (which is the flow map Jacobian at the initial time) is then approximated at each particle with respect to the initial positions

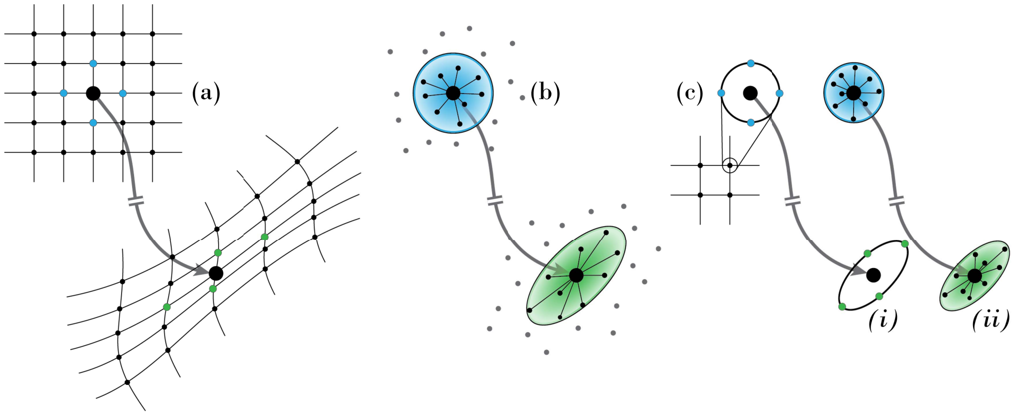


Figure 3.2: Diagrams of computation schemes for the flow map Jacobian in 2D. (a) Finite-differences approach. (b) Regression approach. (c) Planet-satellite approach, which can be accomplished using either (i) finite-differences or (ii) regression.

using a finite-differencing scheme. This process is illustrated in Figure 3.2(a).

In some instances, the finite-differences algorithm for computing $D\mathbf{F}_{t_0}^{t}$ is inefficient. If tracers are given on an unstructured mesh, for instance, additional steps are required to compute the LCS. To address this, Lekien and Ross developed a method for computing $D\mathbf{F}_{t_0}^{t}$ using Voronoi cell-weighted least-squares regression on unstructured meshes [104]. This method was later adapted to experimental particle flows by Raben et al. [105], where regression was used on the observed particle trajectories themselves to compute LCS quantities. The regression approach to LCS approximation is given as a schematic in Figure 3.2(b).

For this work, a third scheme was developed with the intention of directly comparing various numerical approximation techniques. This method, referred to as the planet-satellite approach and displayed in Figure 3.2(c), involves specifying an array of test locations—perhaps on a uniform grid—then seeding particles around around them to locally perform computations either by finite-differences or by regression. The resulting evaluation of $D\mathbf{F}_{t_0}^{t}$ is then provided at the test locations on the initial grid. The primary advantage of this approach is that it allows multiple strategies to be evaluated on a grid of test points and compared directly.

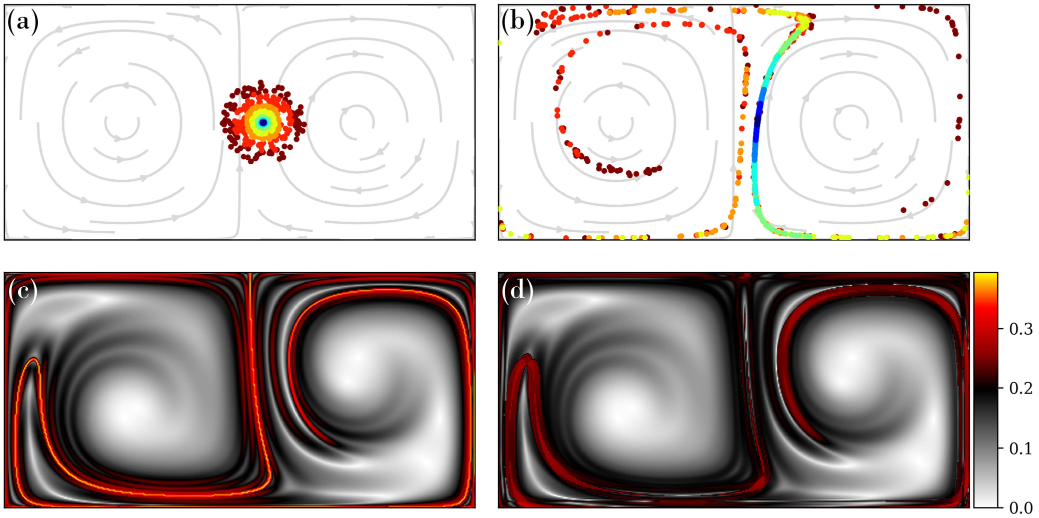


Figure 3.3: A demonstration of the nonlinear influence of the flow on accuracy of the Jacobian for particles with large initial radius. (a) Particles organized by initial radius. (b) Location of particles from (a) after 15 time units in the unsteady double gyre flow. (c) FTLE field evaluated at the level of grid spacing $\Delta\mathbf{p}_0$ ($\Delta\mathbf{p}_0 = \Delta\mathbf{x}_0 = 0.005$: light blue particles). (d) FTLE field evaluated with $\Delta x_0 = 5\Delta\mathbf{p}_0$ ($\Delta\mathbf{x}_0 = 0.025$: green particles). All of the computations are performed using the planet-satellite approach with finite differences (Figure 3.2(c.i))

3.3 The Consequences of Sparsity

Computing accurate Jacobians strongly depends on the distribution of particles at the initial time t_0. Specifically, equation 2.9 shows that accurate computations require tracer spacing $\Delta\mathbf{x}_0$ to be exceedingly small. If an LCS analysis is performed when $\Delta\mathbf{x}_0$ is large, the estimated Jacobian will likely be a poor representation of the tracer deformation as the flow nonlinearities compound.

To demonstrate the influence of initial tracer spacing on approximating $D\mathbf{F}_{t_0}^{t}$, tracer neighborhoods with varying radii are used to compute FTLE fields on the double gyre flow (equation 2.30). Figure 3.3(a) shows a cloud of particles at time t_0 centered at $\mathbf{x}_0$ with varying radii indicated by color. The particles are advected to the final positions (which are displayed in Figure 3.3(b)), where it is clear that the deformation of particles with large initial radii (green to red particles) cannot be reasonably approximated by a linear transformation. However, as the initial radius decreases (blue particles), the deformation of the particles approximates an ellipsoid and can therefore be sufficiently described by a linear operator. It is important to note that, as the integration time increases, the radius that can be accurately approximated by the flow map Jacobian decreases.

It is not clear simply by observing particle deformations how significantly the initial radius influences quantities used in identifying LCS. Figures 3.3(c) and (d) display the forward-time FTLE field computed using different initial neighborhoods with varying radii. In both cases, the planet-satellite method is used with finite differences (Figure 3.2(c.i)), where the set of evaluation locations $\{\mathbf{p}\}$ is a uniform grid and is kept the same between computations. In Figure 3.3(c), the FTLE computations are performed using satellite spacing $\Delta\mathbf{x}_0 = 0.005$, which is equivalent to the grid spacing $\Delta\mathbf{p}$ and the radius of the light blue particles in (a). Here, the FTLE field displays the sharp ridges that are consistent with the literature (see, for example, [41]). The computations in Figure 3.3(d) use $\Delta\mathbf{x}_0 = 5\Delta\mathbf{p} = 0.025$, which is the outer radius of the green particles in (a), and the graphical representation uses the same color mapping as (c). By visual inspection, it is apparent that the FTLE ridges in (d) are muted and distorted. The distortion is so drastic that the dominant ridge that appears in the center of the domain in (c) is difficult to identify in (d). As $\Delta\mathbf{x}_0$ and Δt increase, the quality of the FTLE field decreases.

Progress Towards Sparse Gradient-Based LCS Detection

It is well known that Jacobians are difficult to approximate from sparse trajectories, which has led researchers to develop a variety of methods that demonstrate improved performance under such conditions. To overcome the dependency on a structured array of particles, Lekien and Ross implemented least-squares regression to compute the Jacobian on an unstructured mesh of particles [104]. Brunton and Rowley developed an algorithm utilizing flow map composition to speed up the calculations of FTLE fields, which decreased FTLE computation time for successive frames by reusing prior computations [106]. Raben et al. built upon on these advances by developing strategies for computing FTLE directly from experimental particle trajectories [105]. Recently, Mowlavi et al. proposed a noise-robust method which augmented the regression approach of Lekien and Ross [104] by increasing the number of particle connections within the regression neighborhood and by adding Tikhonov regularization [60].

A separate approach to the sparse identification of gradient-based LCS exists in the recent work by Haller and collaborators (2021-2023) where the trajectory stretching exponent (TSE) and trajectory rotation angle (TRA) have been developed as quasi-objective metrics that approximate traditional metrics like FTLE and polar rotation angle (PRA) but can be computed from single particle trajectories [107], [108]. The theoretical development of the deformation velocity by Kaszas et al. has enabled the TSE and TRA to be computed objectively when using reference velocity information [109], [110]. These methods were shown to be effective at visualizing LCS on sparse data sets, but they remain only an approximate indicator of the actual FTLE, PRA, and LAVD. Moreover, they do not provide the ability to approximate the gradient itself.

3.4 Lagrangian Gradient Regression

Lagrangian gradient regression (LGR) is the principal theoretical development of this work. It refers to a class of techniques used to estimate flow gradients—including velocity gradients and flow map Jacobians—using regression as the backbone with emphasis on improved performance using sparse trajectories. Instead of computing gradients based off of a local linearization in space (as do conventional methods), LGR linearizes in time via temporal discretization in the gradient estimation procedure. The Jacobian of many small deformations are independently calculated and synthesized through composition of flow maps, an operation that is nicely suited for realistic data modalities, which usually abound in temporal resolution while lacking in the ability to densely sample in space. This section introduces the theory enabling LGR to approximate both instantaneous velocity gradients and finite-time flow map Jacobians.

The Linear Deformation Operator

Recall from equation 2.9 that fluid deformations over an arbitrary interval can be described by a linear operator (the flow map Jacobian) if the neighborhood of the deformation is infinitesimally small. However, if that neighborhood is large, the material deformation is dominated by nonlinear flow effects and cannot be represented linearly. This is illustrated in Figure 3.4 through the gradual warping of a circular cloud of particles (shown in blue at the bottom left) into an deformed, banana-shaped filament of tracers (depicted in red at the top right). When a linear operator is fit to the complete deformation, the result inherently misrepresents the true transformation.

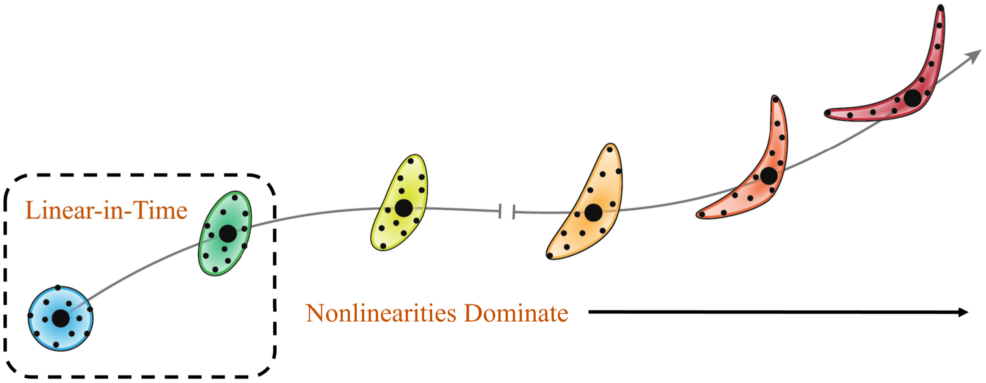


Figure 3.4: An illustration of the deformation of a circular cloud of particles over time in a flow.

However, if the deformation is recorded over an interval relatively small to the fastest time scales of the flow $\mathcal{T}$, then the operator mapping the transformation can still be reasonably approximated as linear. In the figure above, this is indicated by the region labeled *Linear in Time*. Over these very small intervals, the flow map Jacobian accurately represents the deformation. This can be shown mathematically by considering a small perturbation in time Δt at the location of a tracer:

$$\mathbf{F}_t^{t+\Delta t}(\mathbf{x}) = \mathbf{F}_t^t(\mathbf{x}) + \frac{d}{dt}\left(\mathbf{F}_t^t(\mathbf{x})\right)\Delta t + \mathcal{O}\left(\Delta t^2\right). \tag{3.1}$$

As was stated in equation 2.7a, $\mathbf{F}_t^t(\mathbf{x}) = \mathbf{x}$. Therefore,

$$\frac{d}{dt}\left(\mathbf{F}_t^t(\mathbf{x})\right) = \frac{d}{dt}\left(\mathbf{x}\right) = \mathbf{v}(\mathbf{x}, t) \tag{3.2}$$

is just the velocity, and the perturbation is, to first order,

$$\mathbf{F}_t^{t+\Delta t}(\mathbf{x}) = \mathbf{x} + \mathbf{v}(\mathbf{x}, t)\Delta t, \tag{3.3}$$

which also follows from equation 2.5. Taking the gradient of both sides, an expression for the deformation operator is identified as the flow map Jacobian:

$$D\mathbf{F}_t^{t+\Delta t}(\mathbf{x}) = \mathbf{I}_d + \nabla\mathbf{v}(\mathbf{x}, t)\Delta t. \tag{3.4}$$

It is helpful to notice that equation 3.4 is the temporal discretization of the linear process defined by equation 2.11 for $t = t_0$.

This analysis sets the foundations for Lagrangian gradient regression, whose fundamental building block is the instantaneous deformation operator (i.e., the flow map Jacobian defined over short times). Because it is always

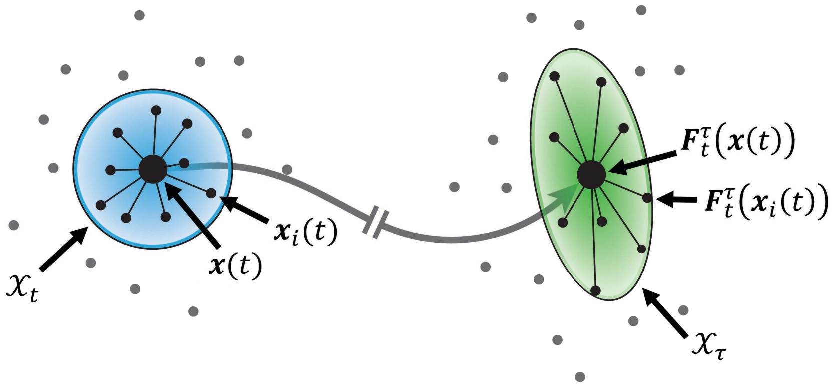


Figure 3.5: An illustration of the components used to form the regression matrices $\mathbf{X}_t$ and $\mathbf{X}_\tau$. A particle $\mathbf{x}(t)$ is tracked from one time step to the next along with all of its neighbors $\mathbf{x}_i(t)$. At the first time step, the differences between the neighbors to the center particle form the columns of $\mathbf{X}_t$, and their deformed differences $\mathbf{F}_t^\tau(\mathbf{x}_i(t)) - \mathbf{F}_t^\tau(\mathbf{x}(t))$ form the columns of $\mathbf{X}_\tau$.

linear, deformations at time $\tau = t + \Delta t$ (again, for small Δt) can be described by the transformation

$$\mathcal{X}_\tau = D\mathbf{F}_t^\tau\left(\mathcal{X}_t\right), \tag{3.5}$$

where $\mathcal{X}_t \subseteq D$ and $\mathcal{X}_\tau \subseteq D$ represent a material region at times t and τ. If only flow-embedded tracers are observable, then $\mathcal{X}_t$ and $\mathcal{X}_\tau$ can be approximated by matrices $\mathbf{X}_t \in \mathbb{R}^{d\times n}$ and $\mathbf{X}_\tau \in \mathbb{R}^{d\times n}$ recording the distances from a central particle to all n neighbors within some neighborhood

$$\mathbf{X}_t = \begin{bmatrix} | & | & & | \\ \Delta\mathbf{x}_1(t) & \Delta\mathbf{x}_2(t) & \cdots & \Delta\mathbf{x}_n(t) \\ | & | & & | \end{bmatrix},$$

$$\mathbf{X}_\tau = \begin{bmatrix} | & | & & | \\ \Delta\mathbf{x}_1(\tau) & \Delta\mathbf{x}_2(\tau) & \cdots & \Delta\mathbf{x}_n(\tau) \\ | & | & & | \end{bmatrix}, \tag{3.6}$$

such that $\Delta\mathbf{x}_i(t) = \mathbf{x}_i(t) - \mathbf{x}(t)$ and $\Delta\mathbf{x}_i(\tau) = \mathbf{F}_t^\tau(\mathbf{x}_i(t)) - \mathbf{F}_t^\tau(\mathbf{x}(t))$ represent the distance from the i^{th} neighbor particle to the center particle at times t and τ, respectively. A schematic of this process is provided in Figure 3.5.

Now the deformation is cast as a linear matrix equation

$$\mathbf{X}_\tau = D\mathbf{F}_t^\tau\,\mathbf{X}_t. \tag{3.7}$$

The formulation is the same as that of dynamic mode decomposition (DMD, [111], [112]), where the operator is solved through regression. Thus, the regression that drives LGR is finally obtained:

$$D\mathbf{F}_t^\tau = \mathbf{X}_\tau \mathbf{X}_t^\dagger, \tag{3.8}$$

where $\mathbf{X}^\dagger = \mathbf{X}^\top \left(\mathbf{X}\mathbf{X}^\top\right)^{-1}$ is the right Moore-Penrose pseudoinverse.

Kernel-Weighted Least Squares Regression of Deformations

The linear deformation operator $D\mathbf{F}_t^\tau$ can be directly approximated through least-squares regression as stated in equation 3.8. This is the solution to the mimization problem

$$D\mathbf{F}_t^\tau = \underset{D\mathbf{F}_t^\tau}{\arg\min} \left(\frac{1}{2} \left\|\mathbf{X}_\tau - D\mathbf{F}_t^\tau \mathbf{X}_t\right\|_F^2\right) \tag{3.9}$$

[113]. It is similar to the approach developed by Lekien and Ross to compute Jacobians on unstructured meshes [104], but differs principally in the duration of the computation interval. Whereas Lekien and Ross perform their regression over arbitrary durations, this work limits the regression of the Jacobian to only short intervals so that the identified linear model is an accurate representation of the observed deformation with sparse tracer density. The work of Mowlavi et al. builds on that of Lekien and Ross by incorporating all connections within some neighborhood to the data matrices rather than only the connections with the center particle [60]. However, like Lekien and Ross, they do not restrict computations to short durations, and therefore are still subject to the influence of flow nonlinearity when operating on sparse data.

Aside from restricting regressions to short intervals, this work also proposes to weight the regression problem with a kernel matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$. The kernel matrix is designed by the user to prioritize the influence of certain tracers within the neighborhood represented by the matrix $\mathbf{X}_t$. In this framework, the minimization problem is cast as

$$D\mathbf{F}_t^\tau = \underset{D\mathbf{F}_t^\tau}{\arg\min} \left(\frac{1}{2} \left\|\mathbf{K}^{\frac{1}{2}} \left(\mathbf{X}_\tau - D\mathbf{F}_t^\tau \mathbf{X}_t\right)\right\|_F^2 + \frac{\gamma}{2} \left\|D\mathbf{F}_t^\tau\right\|_F^2\right), \tag{3.10}$$

where the second term in the minimization is a regularizer with regularization constant γ. The solution of the optimization in equation 3.10 is then

$$D\mathbf{F}_t^\tau = \mathbf{X}_\tau \mathbf{K} \mathbf{X}_t^\top \left(\mathbf{X}_t \mathbf{K} \mathbf{X}_t^\top + \gamma n \mathbf{I}_d\right)^{-1} \tag{3.11}$$

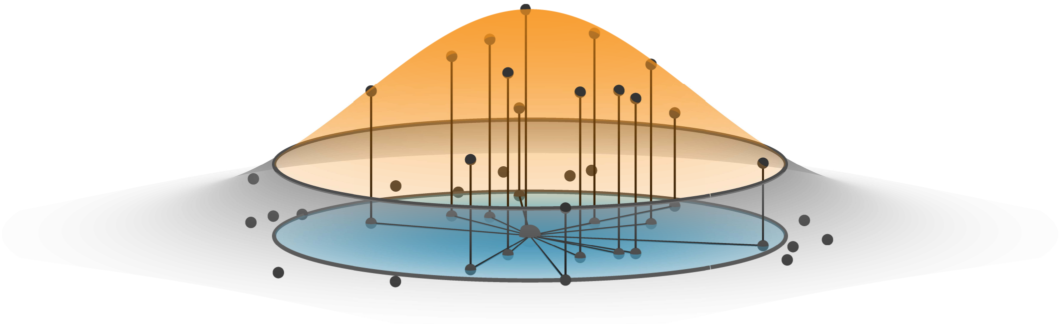


Figure 3.6: An illustration of the radial Gaussian weighting kernel described in equation 3.12. Particles are indicated as gray dots on the plane of the blue circle, which is viewed at a 45° angle from above. The weighting function is indicated as an orange surface over the particles. The weight of each particle to the regression computation is indicated by its value on the orange surface.

[113].

While the addition of the regularizer may help to reduce the influence of noise in the regression computation [60], it will also bias the regressed deformation operator to smaller-than-real values. In this work, only very small values of γ are used in computations in order to maintain numerical stability.

Designing a kernel matrix $\mathbf{K}$ can significantly influence the operator identified by equation 3.11. Many construction strategies exist. Kernels are valid as long as $\mathbf{K}$ is symmetric and positive semi-definite [114]. Typically, one defines a kernel matrix according to a kernel function $k(\Delta\mathbf{x}, \Delta\mathbf{x}') : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}^+$, which computes the distance between data in the regression.

In this work, $\mathbf{K}$ is either set to be the identity matrix $\mathbf{I}_n$, which ascribes equal weight to particles any distance from the center particle, or particles are given weights according to a Gaussian function over the radius. In the latter case, the kernel function is defined

$$\mathbf{k}_{RG}(\Delta\mathbf{x}_i(t), \Delta\mathbf{x}_j(t)) = \begin{cases} \alpha^2 \exp\left(-\frac{(\Delta\mathbf{x}_i(t))^2}{2l^2}\right), & \text{if } i = j \\ 0, & \text{otherwise} \end{cases}, \tag{3.12}$$

where the output scaling α^2 and the input variance l^2 are hyperparameters. An illustration of this this weighting scheme is provided in Figure 3.6. In practice, allowing $\mathbf{K} = \mathbf{I}_n$ is preferable when the particle spacing is already dense relative to the spatial scales of the flow or when the uncertainty of particle trajectories is larger. By equally weighting over the entire radial domain, the influence of errant trajectories is reduced.

Approximating The Velocity Gradient

Now, with the ability to accurately compute short-time flow map Jacobians via regression, it is a simple matter to then compute velocity gradients. Rearranging equation 3.4 yields

$$\nabla \mathbf{v}(\mathbf{x}, t) = \frac{D\mathbf{F}_t^{\tau} - \mathbf{I}_d}{\Delta t}, \tag{3.13}$$

where the result applies at time t. As a side note, this result can also be reached by allowing $t \to t_0$ in equation 2.12 and discretizing.

Importantly, equation 3.13 implies that an approximation of the velocity gradient is achievable using only trajectory information; it does not require any velocity information or finite-differencing. Instead, the robust, data-driven machinery of regression drives the operation. An algorithm to approximate velocity gradients via LGR is provided in Appendix C.

Approximating The Flow Map Jacobian

So far, LGR has been developed to approximate short-time flow map Jacobians and instantaneous velocity gradients. In order to extend the framework of LGR to finite observation times, the theory of flow map composition must be considered.

The use of flow map compositions for computing $D\mathbf{F}_{t_0}^{t}$ was pioneered by Brunton, Rowley, and Luchtenberg [106], [115], and has been applied in various other studies (e.g., Raben et al. [105]). The theory stems from the process property of the flow map

$$\mathbf{F}_{t_0}^{t_n}(\mathbf{x}_0) = \mathbf{F}_{t_{n-1}}^{t_n} \circ \cdots \circ \mathbf{F}_{t_1}^{t_2} \circ \mathbf{F}_{t_0}^{t_1}(\mathbf{x_0}), \tag{3.14}$$

which states that any flow map from time t_0 to t_n can be defined as the composition of n intermediate flow maps, as long as there are no gaps between intervals. Applying the chain rule to equation 3.14 yields

$$\begin{aligned} D\mathbf{F}_{t_0}^{t_n}(\mathbf{x}_0) &= D\left(\mathbf{F}_{t_{n-1}}^{t_n} \circ \cdots \circ \mathbf{F}_{t_1}^{t_2} \circ \mathbf{F}_{t_0}^{t_1}(\mathbf{x_0})\right), \\ &= D\left(\mathbf{F}_{t_{n-1}}^{t_n}\left(\mathbf{F}_{t_0}^{t_{n-1}}(\mathbf{x}_0)\right)\right) D\left(\mathbf{F}_{t_{n-2}}^{t_{n-1}}\left(\mathbf{F}_{t_0}^{t_{n-2}}(\mathbf{x}_0)\right)\right) \dots D\left(\mathbf{F}_{t_0}^{t_1}(\mathbf{x}_0)\right), \\ &= D\mathbf{F}_{t_{n-1}}^{t_n}\left(\mathbf{F}_{t_0}^{t_{n-1}}(\mathbf{x}_0)\right) D\mathbf{F}_{t_{n-2}}^{t_{n-1}}\left(\mathbf{F}_{t_0}^{t_{n-2}}(\mathbf{x}_0)\right) \dots D\mathbf{F}_{t_0}^{t_1}\left(\mathbf{x}_0\right). \end{aligned} \tag{3.15}$$

Then, recalling from equation 2.4 that $\mathbf{x}(t_i) = \mathbf{F}_{t_0}^{t_i}(\mathbf{x}_0)$, the composition operation can be succinctly stated as

$$D\mathbf{F}_{t_0}^{t_n}(\mathbf{x}_0) = \prod_{i=0}^{n-1} D\mathbf{F}_{t_i}^{t_{i+1}}\left(\mathbf{x}(t_i)\right). \tag{3.16}$$

It is helpful to make some remarks regarding the composition framework:

1. Equation 3.16 applies to every tracer in the flow for all times along its trajectory regardless of the method used to compute $D\mathbf{F}_{t_i}^{t_{i+1}}$. Additionally, the important temporal constraint is that computational time domains are consecutive. Any given interval may progress forward in time, backward in time, or not at all, but all intervals must be connected.

2. While previous methods involving flow map composition achieved their respective goals by performing interpolation to a grid at each time step (e.g., [105], [106]), it is not a requirement for using flow map composition. As long as calculations are performed consecutively along the particle trajectory, no interpolation is necessary.

3. There is no practical advantage to using composition for Jacobian estimation if the same group of tracers is used to approximate each intermediate Jacobian operator. Computing the flow map Jacobians $D\mathbf{F}_{t_i}^{t_{i+1}}$ at each interval and synthesizing to $D\mathbf{F}_{t_0}^{t_n}$ through equation 3.16 yields exactly the same results as if $D\mathbf{F}_{t_0}^{t_n}$ were computed using the initial and final times of those particles alone. Thus, composition alone does not solve the problem of sparse identification of LCS.

To overcome the problem posed in the third remark above, a computation scheme which resamples tracers at each time step is proposed. The process of composition with resampling is presented as a schematic in Figure 3.7. In the resampling paradigm, tracers which have exceeded a threshold radius from the analyzed trajectory are discarded at each time step and replaced by others that are within closer proximity. As discussed in the previous section, if the time increment is small enough ($\Delta t/\mathcal{T} \ll 1$), the material deformation is approximately linear and the short-time Jacobian can be accurately approximated. Then, by applying equation 3.16, the complete Jacobian over time $[t_0, t_n]$ can be accurately constructed. An algorithm which leverages LGR to compute Jacobians is presented in Appendix C.

Practically, resampling is achieved by a variety of methods. If computations are performed using numerical tracers, then resampled tracers may be placed randomly in a neighborhood, at specific azimuthal locations on a constant radius, or by any other means that is convenient for the application. When using pre-existing trajectories from observed tracers, resampling might

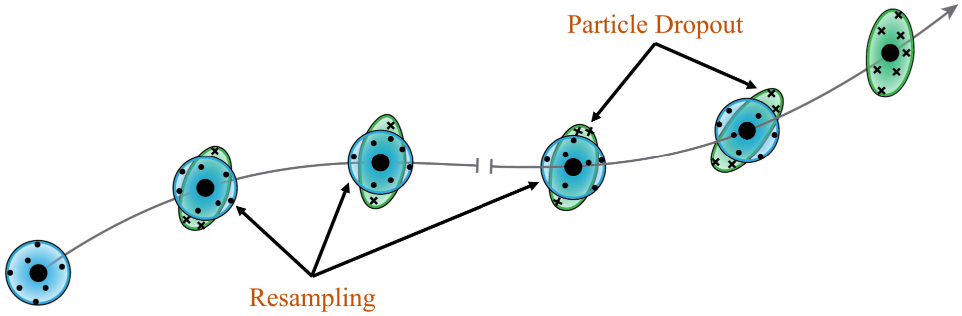


Figure 3.7: Schematic of the resampling procedure. At each time step, particles used in regression are resampled to ensure that each regression stays locally linear in time.

be accomplished by selecting all tracers within a neighborhood around the analyzed particle at each time step or by selecting its k-nearest neighbors. Examples of resampling procedures are illustrated in Figure 3.8. Of course, those discussed are not exhaustive.

Aside from enabling more accurate Jacobian estimation (as will be clearly demonstrated in subsequent chapters), composition with resampling relaxes a significant constraint endemic to Lagrangian methods—namely, it reduces the need for tracer persistence in the frame. Because Lagrangian methods all depend on observing tracer trajectories over time, most Lagrangian methods require all analyzed particles to remain visible (and tracked) over the entire duration of analysis. Resampling, however, reduces this need by only requiring that the center particle—the one whose trajectory is the basis for computations—be tracked over the entire interval. The neighboring particles at any given time step are only required to have defined trajectories from one increment to the next. This makes LGR much more sparsity robust than other methods used to estimate the flow map Jacobian.

3.5 Integrated Flow Metrics Using LGR

Traditional Lagrangian analyses often limit the amount of information that contributes to the computed quantity by only incorporating the initial and final positions of tracers. FTLE, for example, represents the linear rate of deformation between initial and final positions, but entirely ignores the behavior of the flow in between. This is unlike LAVD, which integrates vorticity deviation over the entire trajectory.

By computing and storing gradients at each time-step, the LGR algorithm

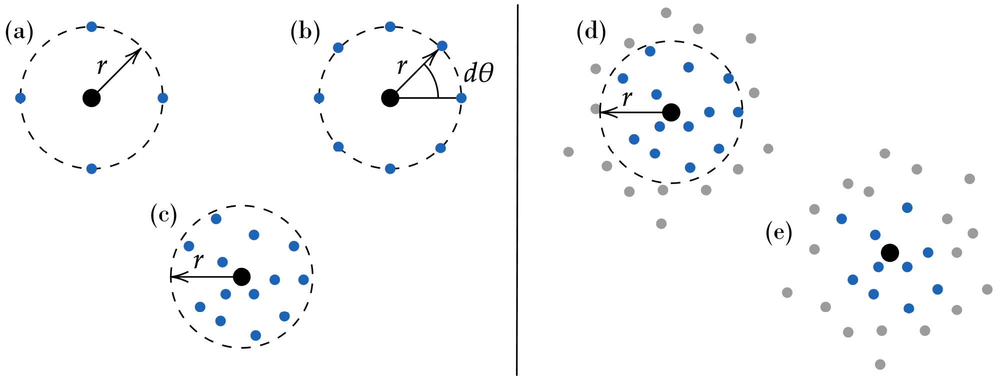


Figure 3.8: Examples of tracer resampling approaches. Examples (a), (b), and (c) represent approaches one might consider if numerically sampling tracers. (a) corresponds to tracer arrangement typical of finite-differences, (b) augments (a) with additional tracers with equal radial spacing, and (c) indicates random sampling inside a neighborhood. Examples (d) and (e) represent potential schemes to be used with observed tracer data. (d) depicts the identification of all tracers at a given time inside a specified radius, and (e) represents the identification of k-nearest neighbors centered at a particle.

leverages the temporal resolution of modern data modalities to naturally enable the computation of quantities that depend on the temporal evolution of gradients. In this section, two metrics are introduced.

Trajectory-Integrated Strain

Recall from equation 2.28 that the LAVD represents the trajectory-integrated vorticity deviation. By the same logic, it possible to compute a trajectory-integrated measure of strain using the template

$$\Lambda_{t_0}^{t} = \int_{t_0}^{t} \|\mathbf{D}(\mathbf{x}(\tau), \tau)\| \, d\tau, \tag{3.17}$$

where the type of norm used dictates the physical interpretation. If the spectral norm $\|\mathbf{D}\|_2 = \sigma_{\max}(\mathbf{D})$ is used, where $\sigma_{\max}$ is the maximum singular value, then the metric represents the maximum strain experienced by the fluid element over the time domain in any direction. On the other hand, if the Frobenius norm $\|\mathbf{D}\|_F = \sqrt{\sum_i \sigma_i^2(\mathbf{D})}$ is used, where σ_i are the singular values of $\mathbf{D}$, the metric represents the total strain experienced by the fluid element over the trajectory in the intrinsic frame of reference for of the particle. Using this second formulation, the trajecory-integrated strain magnitude (TISM) is defined as

$$\mathrm{TISM}_{t_0}^{t} = \int_{t_0}^{t} \|\mathbf{D}(\mathbf{x}(\tau), \tau)\|_F \, d\tau. \tag{3.18}$$

In the context of autonomous mobile flow measurements (AMFM), replacing $\mathbf{D}$ with the traceless deviatoric rate-of-strain tensor

$$\mathbf{D}_{dev} = \mathbf{D} - \mathbf{D}_{hyd}, \tag{3.19}$$

may be useful. The hydrodynamic rate-of-strain tensor

$$\mathbf{D}_{hyd} = \frac{\mathrm{tr}\left(\mathbf{D}\right)}{d}\mathbf{I}_d \tag{3.20}$$

with d the dimension of the flow represents uniform strain experienced in all directions. When observing planar flows with a mobile observer considering removing the hydrodynamic strain may be useful since it could help account for out-of-plane observer motion. If the observer moves away from the flow in the out-of-plane direction it will appear that the tracers are contracting, and if it moves towards the flow on the same axis it will appear that the tracers are expanding. Removing $\mathbf{D}_{hyd}$ from $\mathbf{D}$ helps to mitigate this aberration. To that effect, the trajecory-integrated deviatoric strain magnitude ($\mathrm{TISMd}_{t_0}^{t}$) is

$$\mathrm{TISMd}_{t_0}^{t} = \int_{t_0}^{t} \left\| \mathbf{D}_{dev}(\mathbf{x}(\tau), \tau) \right\|_F \, d\tau. \tag{3.21}$$

A demonstration of the TISM on the double gyre flow is provided in Figure 3.9(a). The flow parameters are the same as was used in the demonstrations of Chapter 2. Comparing the results with the forward-time FTLE field of Figure 2.6(a), it is evident that the ridge of the TISM is the same as the ridge in the FTLE field. However, the the gradient in the TISM field is less sharp and more aptly highlights regions characterized by stretching rather than a discrete separatrix.

Kinematic Action as an Objective Vortex Criteria

The ability to objectively identify vortices in messy data is one of the key objectives of AMFM. While methods such as LAVD and DRA provide a notion of rotation in a Lagrangian frame, identifying vortex boundaries using them requires dense data [51]. The typical procedure involves finding closed contours in level-sets of the LAVD field that satisfy a specified convexity threshold. While this method has been shown to be effective at finding vortex boundaries and cores in dense data sets, it is not a process that is easily implemented with sparse data.

There are, however, a variety of Eulerian metrics which allow a user to easily discern by a simple heuristic whether or not a portion of the flow is

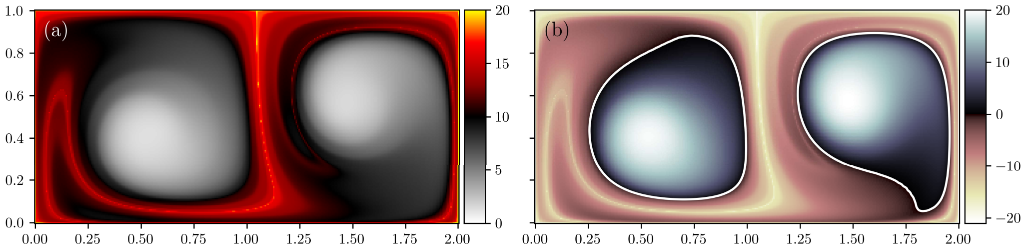


Figure 3.9: Integrated metrics that are conveniently computed using the LGR framework computed on the double gyre flow. (a) Trajectory-integrated stretching magnitude (TISM), as defined by equation 3.18. Brighter color represent regions of greater integrated stretching. (b) Kinematic action $Q_{t_0}^t$ as defined by equation 3.22. Positive values represent regions of the flow dominated by rotation, while negative values represent regions dominated by stretching. The white contour is drawn where $Q_{t_0}^t = 0$.

rotation dominated. One of the most common of these metrics is the Q-criterion, which was introduced in equation 2.16 [44]. As discussed in that section, the Q-criterion decomposes the velocity gradient tensor into rate-of-strain and spin tensors, and measures their magnitude against each other. Thus, positive values of Q-criterion represent rotationally dominated flow and negative values represent stretch-dominated regions. However, as mentioned, the Q-criterion is not objective.

While objective variants of the Q-criterion have been developed [116], [117] as well as other objective measures of rotation (e.g., [118]), these have not been extended to finite times. Making this extension, however, is straightforward using LGR as a baseline. Here, the kinematic action $Q_{t_0}^t(\mathbf{x}_0)$ it is proposed that an objective, finite-time vortex-identification criterion be defined according to

$$Q_{t_0}^t(\mathbf{x}_0) = \int_{t_0}^{t} \left(\left\| \mathbf{W}(\mathbf{x}(\tau),\tau) - \overline{\mathbf{W}}(\tau) \right\|_F - \|\mathbf{D}(\mathbf{x}(\tau),\tau)\|_F \right) d\tau, \tag{3.22}$$

where $\mathbf{D}$ can be interchanged with $\mathbf{D}_{dev}$ to mitigate out-of-plane motion effects. If $Q_{t_0}^t > 0$, then the flow is rotation dominated over the interval at $\mathbf{x}_0$. By removing the spatial mean of the spin, the metric is made objective. By integrating, it is made finite-time.

The differences in the integrand of $Q_{t_0}^t$ with the traditional Q-criterion are designed so that the result is more closely aligned with other finite-time metrics being examined in this work. In particular, $Q_{t_0}^t$ as specified represents a direct combination of LAVD and TISM

$$Q_{t_0}^t(\mathbf{x}_0) = \frac{1}{\sqrt{2}}\mathrm{LAVD}_{t_0}^t - \mathrm{TISM}_{t_0}^t. \tag{3.23}$$

To show this, consider the definition of the IVD from equation 2.18, reproduced here:

$$\mathrm{IVD}(\mathbf{x}, t) = |\omega(\mathbf{x}, t) - \overline{\omega}(t)| = \sqrt{\sum_i (\omega_i - \overline{\omega}_i)^2}. \tag{3.24}$$

Furthermore, recall that the spin tensor $\mathbf{W}$ is anti-symmetric with off-diagonal terms related to the vorticity $|W_{ij}| = \frac{1}{2}|\frac{\partial u_j}{\partial x_i} - \frac{\partial u_i}{\partial x_j}|$. Therefore, the Frobenius norm of the spin deviation is

$$\begin{aligned}
\left\|\mathbf{W} - \overline{\mathbf{W}}\right\|_F &= \sqrt{\sum_i \sum_j \left|W - \overline{W}\right|_{ij}^2} \\
&= \sqrt{2 \sum_k \left(\frac{1}{2} \left|\omega - \overline{\omega}\right|_k\right)^2} \\
&= \frac{1}{\sqrt{2}} \left|\omega - \overline{\omega}\right| \\
&= \frac{1}{\sqrt{2}} \mathrm{IVD}.
\end{aligned}$$

Integrating this result over time leads to the LAVD by equation 2.27. Adding this to the results introduced above with TISM leads to the relation in equation 3.23.

The kinematic action is demonstrated on the double gyre flow in Figure 3.9(b) using the same parameters as usual. Here, the flow is partitioned into rotational and dilatational domains by the value of $Q_{t_0}^t$. If $Q_{t_0}^t < 0$, the flow is dominated by stretching over the time interval. If $Q_{t_0}^t > 0$, it is dominated by rotation. By this analysis, the location of the two vortices in the double gyre are clearly evident and their boundaries over the $t = [0,\ 15]$ interval are easily defined as level sets of $Q_{t_0}^t \geq 0$. As will be discussed in more detail later, this metric is convenient for dealing with sparse data, as it provides an easy criteria for locating vortical regions without access to dense fields for convexity computations.

Chapter 4

EVALUATION OF LAGRANGIAN GRADIENT REGRESSION ON SPARSE DATA

In Chapter 2, background theory from the domain of Lagrangian coherent structures (LCS) was introduced in response to the problem of measurement objectivity faced by autonomous mobile flow field measurements (AMFM). How does one make consistent measurements of a system when its relationship to the observer is potentially unknown? One way, it was shown, is to compute deformation gradients from the motion of observable tracers. However, in Chapter 3, the difficulty of actually estimating such gradients from realistic data was addressed. Typically, the density of tracers required by the numerics for reliable approximation exceeds what is available in real-world observations. In that chapter, Lagrangian gradient regression (LGR) was developed to enable sparse gradient estimation, and thereby an objective means for identifying flow features in the AMFM context.

The aim of the present chapter is to evaluate LGR as a tool for studying flows characterized by sparse trajectories. As was discussed in Chapter 3, LGR enables the approximation of velocity gradients instantaneously and deformation gradients (which have also been referred to as flow map Jacobians) over a finite interval. Therefore, this chapter is organized to assess each of these categories separately.

Section 4.1 addresses LGR's ability to approximate the velocity gradient from sparse, random data. This is first tested using perfect trajectories that are numerically simulated, where LGR is observed to reproduce analytical values of vorticity. This analysis is then extended to numerically simulated particle images. LGR is compared with particle image velocimetry (PIV) and Lagrangian particle tracking (LPT) in its ability to compute gradients. Using

---

The contents of this chapter have been adapted from Harms, Brunton, and McKeon (2023) [1], which was published in the proceedings of the 15th International Symposium on Particle Image Velocimetry under the title *Direct Computation of Velocity Gradients from Particle Trajectories*, and from Harms, Brunton, and McKeon (2024) [2], which has been accepted to be published by the Royal Society Journal of Open Science under the title *Lagrangian Gradient Regression for the Detection of Coherent Structures from Sparse Trajectory Data* and is available as a preprint on arXiv.

vorticity as the gradient-based metric for comparison, LGR is seen to perform comparably, and in some cases better, than PIV and LPT.

Section 4.2 attempts to address the notion of sparsity in the LGR process. Specifically, it discusses LGR's sensitivity to spatial scales. If tracers are spread far across a flow's domain, it is necessary to understand what is being resolved when LGR is applied to the data. Some theory is provided along with a discussion based on pre-computed PIV data collected from experiments completed at Caltech.

Finally, Section 4.3 tests LGR's ability to estimate finite-time gradients and their associated metrics. First, finite-time Lyapunov exponent (FTLE) fields are computed by a variety of methods and it is observed that LGR achieves better accuracy than the others for sparse data. Next, integrated metrics are considered, including trajectory-integrated stretching magnitude (TISM), Lagrangian-averaged vorticity deviation (LAVD), dynamic rotation angle (DRA), and kinematic action (KA). These are seen to be especially robust under sparse conditions, and may have useful attributes for structure detection in AMFM.

A majority of the analyses in this chapter are computed on the simple Double Gyre flow (see equation 2.30). While many more complex benchmark cases exist, the Double Gyre is easily interpreted and commonly available in the literature (e.g., see [41], [48]). In Chapter 5, more advanced examples will be considered. For now, the testbed of the naive Double Gyre provides an apt environment to consider the ability of LGR relative to other gradient estimation schemes.

4.1 Estimating Velocity Gradients using LGR

LGR is first tested for its ability to approximate velocity gradients. As discussed in Chapter 2, a variety of metrics are based on the velocity gradient, including vorticity. Since vorticity is commonly sought from experimental flow field data, it is used as the demonstration metric throughout this section. Principal strain rate, Q-criterion, or a variety of other metrics could equivalently have been chosen, and would yield similar results and conclusions.

Performance on Sparse, Simulated Data

To begin the suite of $\nabla \mathbf{v}$ analyses performed on the Double Gyre, LGR is computed on clean, numerically simulated trajectory data. To apply LGR to randomly distributed trajectories such as this, Algorithm 1, which is detailed Appendix C, is implemented. Having thus computed $\nabla \mathbf{v}$ at each particle location and for each time step, vorticity ω_z is calculated by the off-diagonal terms of the velocity gradient tensor

$$\omega_z = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = \nabla \mathbf{v}[1,\, 0] - \nabla \mathbf{v}[0,\, 1]. \tag{4.1}$$

Figure 4.1 presents the results a numerical experiment performed using the procedure delineated above for simulated trajectory distributions varying in particle density. In each frame of the figure, the number shown on the lower right-hand corner indicates the number of tracers simulated in the flow. The locations of these tracers at the analyzed snapshot ($t = 0$) are represented as gray dots. LGR is applied to the trajectories using the 15 nearest neighbors of each tracer during regressions with kernel weighting defined by a Gaussian function of radial distance from the test trajectory with standard deviation of $s = 0.125$ (see equation 3.12). Regressions are performed over time increments of $\Delta t = 0.1$. The resulting vorticity is fully Lagrangian—it is defined only at the particle positions along their trajectories—so it is interpolated to a uniform grid for visualization purposes. Results in each frame are defined at time $t = 0$, and the chosen color mapping is based on the analytically computed vorticity (shown previously in Figure 2.5a) at that time.

Figure 4.1 demonstrates that vorticity—and more generally, the velocity gradient—can be accurately computed directly from Lagrangian trajectories recorded over short intervals. For the cases using 1000, 500, and 250 tracers, the interpolated vorticity fields closely resemble the ground truth results presented in Figure 2.5a. When computations are performed with 100 tracers, the shape of the two dominant vortices is still evident, although the interpolated fields begin to appear less clean. This trend continues as particle density decreases. When computations are performed using 50 or 25 tracers, the sign and location of the vorticity is evident, but its shape is altered and peak magnitude dampened. These visual discrepancies over the field are largely the result interpolation effects, though, as particle density is reduced, the point-wise accuracy diminishes as well.

Comparing Gradients from PIV, LPT, and LGR

Continuing tests on the Double Gyre flow, vorticity computed by LGR is compared with vorticity estimated by particle image velocimetry (PIV) and

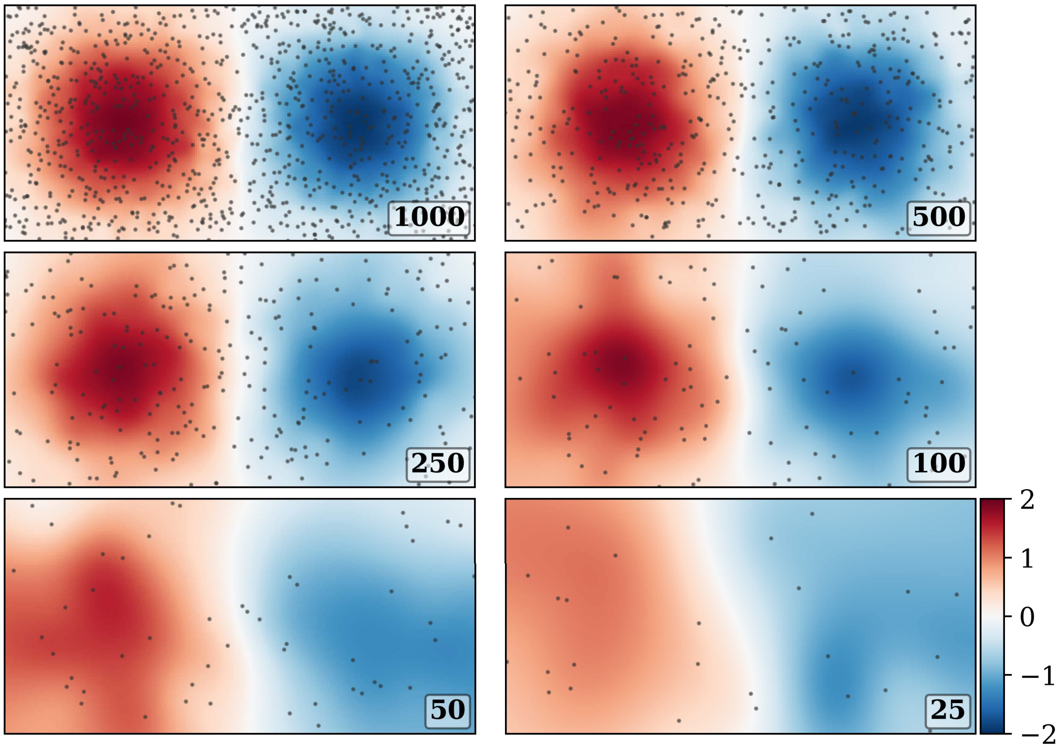


Figure 4.1: Computing vorticity ω_z on sparse data using LGR. Computations are performed at time $t = 0$ on the Double Gyre flow using the 15 nearest neighbors of each tracer with radial-Gaussian weighting using a standard deviation of $s = 0.125$. The number of simulated particles is indicated in the lower right hand of each frame, and the location of those particles at $t = 0$ are indicated by gray dots. Sparse values are interpolated to a field using RBF interpolation and colored for visualization purposes.

by Lagrangian particle tracking (LPT, often referred to as particle tracking velocimetry (PTV)). To illustrate how all three of these approaches for gradient estimation relate, Figure 3.1 is revisited in Figure 4.2 with the addition of the LGR process. Computing gradients by LGR begins with the same trajectories identified in the PTV process from recorded particle images. However, instead of interpolating Lagrangian velocities to Eulerian fields, velocity gradients are computed in a fully Lagrangian manner along each tracer trajectory. Therefore, with LGR, no explicit differentiation is required, nor is any interpolation. This affords a decrease in computational burden and storage demand while simultaneously circumventing procedures that are known to induce error (for example, see the works of Etebari and Vlachos and of Beresh et al., which evaluate differentiation methods for spatial gradient estimation in experimental fluids data [119], [120]).

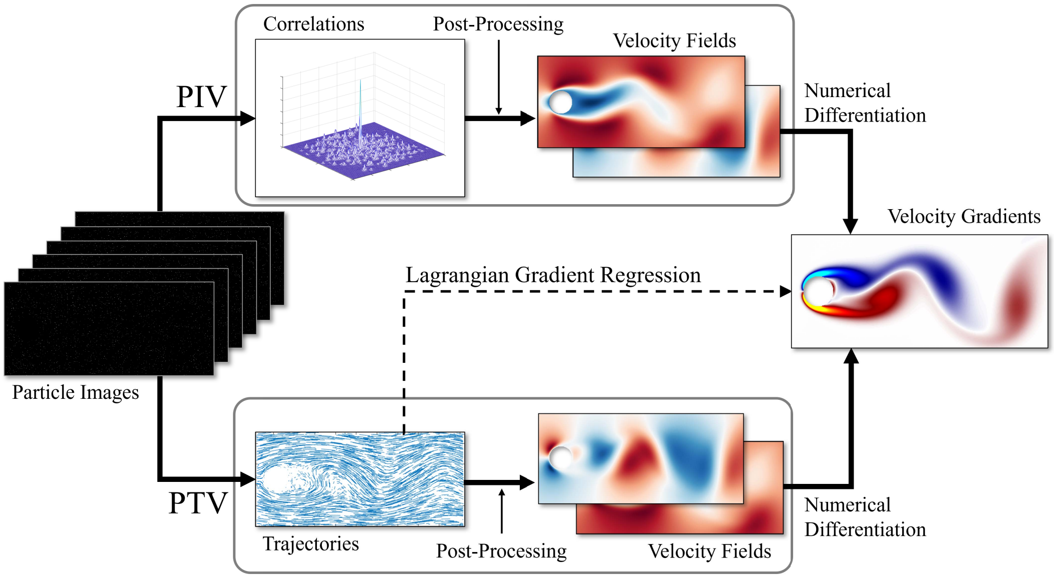


Figure 4.2: Visualization of the processes commonly employed to compute velocity gradients from particle images. The PIV pipeline follows the top branch. The PTV (also commonly referred to as LPT) pipeline follows the bottom branch. LGR begins with PTV, but does not interpolate to velocity fields. Rather, it directly approximates velocity gradients from identified trajectories.

Independent sets of trajectories were generated by integrating randomly distributed particles using equation 2.30. The sets of trajectories were organized according to particle density where 25 particles throughout the domain was the sparsest set generated and 5000 particles was the most dense. An in-house designed synthetic particle image generator was used to create image sequences from which PIV and LPT could be computed. The image generator allowed for particle size and illumination to vary within reasonable limits and generated sequences of 1000 images with 1000 × 500 square pixels at a framerate of 20Hz. Particle illumination profiles were modelled as Gaussian. Samples from the synthetic image generator are displayed in Figure 4.3.

Particle image velocimetry was performed on the synthetic images using the open-source python package OpenPIV [121]. While many variations of parameters can be used when performing PIV, the intent of this study was to process the data earnestly, without withholding steps often used during processing. The sequences of images were treated as time-resolved, so every adjacent pair was processed. Multiple PIV passes were performed for each image pair—a first pass using 64 × 64 pixel square windows with 50% overlap and a second pass using 32 × 32 square windows with 50% overlap. Subpixel

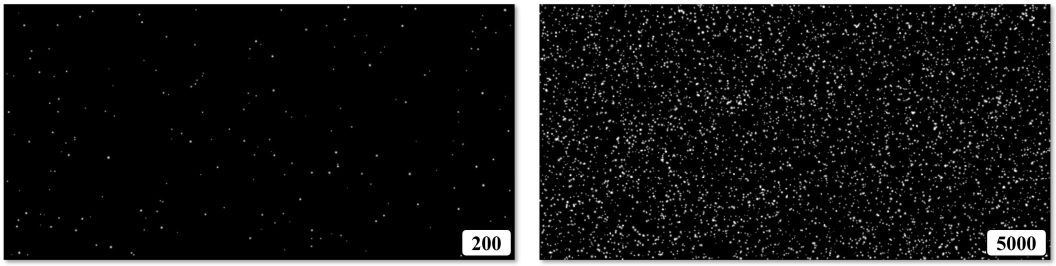


Figure 4.3: Examples of synthetic PIV images generated on the Double Gyre flow. Left: 200 particles in frame. Right: 5000 particles in frame.

interpolation was performed using a Gaussian method. Vectors were validated on both passes, and all empty or rejected vectors were filled using a third order interpolation scheme. Outliers were replaced using a local mean approach with four iterations. No smoothing of the velocity fields was performed. Velocity gradients, and therefore vorticity, were computed by first-order finite differencing on the computed velocity fields as is commonly performed with experimental data. Example snapshots of the PIV processing results for various particle densities are displayed in the left-hand column of Figure 4.4.

Lagrangian particle tracking was also performed on the synthetic data sets as a baseline. The particular implementation of particle tracking used for this study is freely available online from Professor Nicholas Ouellette's laboratory web page [122], [123]. The LPT algorithm output particle trajectories with associated velocities computed at each time step. Lagrangian velocities were interpolated to Eulerian fields in both the x and y axes of the the flow using radial basis function (RBF) interpolation if the number of simulated tracers was less than 2000 and cubic interpolation otherwise. From the interpolated velocity fields, velocity gradients were approximated by first-order finite differencing. Vorticity was extracted from the result using the same finite-differencing approach employed with PIV data. Processed LPT vorticities are presented in the center column of Figure 4.4 for the same data and conditions as are presented for PIV results.

The velocity gradients for each set of particle trajectories were also computed directly from the trajectories using LGR. Regressions were performed on deformations of the 15 nearest neighbors of each analyzed trajectory weighted using the radial-Gaussian approach from equation 3.12 with a standard deviation $s = 0.125$. Approximations of the flow map Jacobian $D\mathbf{F}_{t_0}^{t}$ were made over the short interval $\Delta t = 0.1$ using equation 3.11. From this, velocity gradients were estimated by equation 3.13, and vorticity computed by equation 4.1. For visualization purposes, vorticities were interpolated to a field using RBF interpolation if less than 2000 tracers were being simulated, or cubic interpolation otherwise. No additional processing was performed on the LGR data. Results from LGR are presented in the right-hand column of Figure 4.4.

It is necessary to note that many variations of PIV and LPT parameters and algorithms may be applied to a given data set to varied effect. Practitioners of these methods will often attempt to optimize the particle density observed in their images to effect the best possible outcome. In this case, however, it was deemed that varying parameters between particle densities introduced unnecessary variability into the analysis at hand. This analysis compares all tracer densities given a fixed set of processing parameters for each chosen algorithm to prioritize the comparison over individual case performance. It is recognized that improved accuracy against analytical results could likely be achieved for each tracer density condition for each algorithm (including LGR).

Additionally, a note is added here on the choice of interpolation functions. As RBF interpolation tends to provide clean, smooth results and allows for ease of extrapolation beyond the convex hull of the data, it was preferred in the sparser cases. However, the computational complexity of RBF interpolation scales $\mathcal{O}(n^3)$ for the number of particles in frame, and therefore becomes computationally burdensome when more tracers were present.

The visual comparison of LGR gradients to those of PIV and LPT is helpful but, as a former lab member used to say, “A pseudocolor plot can cover a multitude of sins.” To supplement the plotted fields in Figure 4.4, a statistical analysis of estimated velocity gradients was also performed. Instead of comparing vorticity fields, Lagrangian vorticities are computed at tracer positions and compared. Since PIV and LPT do not naturally approximate vorticities at particle locations, an additional interpolation step was required to transfer vorticity field data from a regular grid to the scattered particle positions. Since the vorticity fields are already densely defined from previous calculations, a linear interpolation scheme was sufficient. Thus, PIV and LPT pointwise vorticities are linearly interpolated to particle positions from the pre-computed vorticity fields.

Since the Double Gyre flow is analytical, the ground truth velocity gradient at every time and position is known exactly. Therefore, predicted vorticities

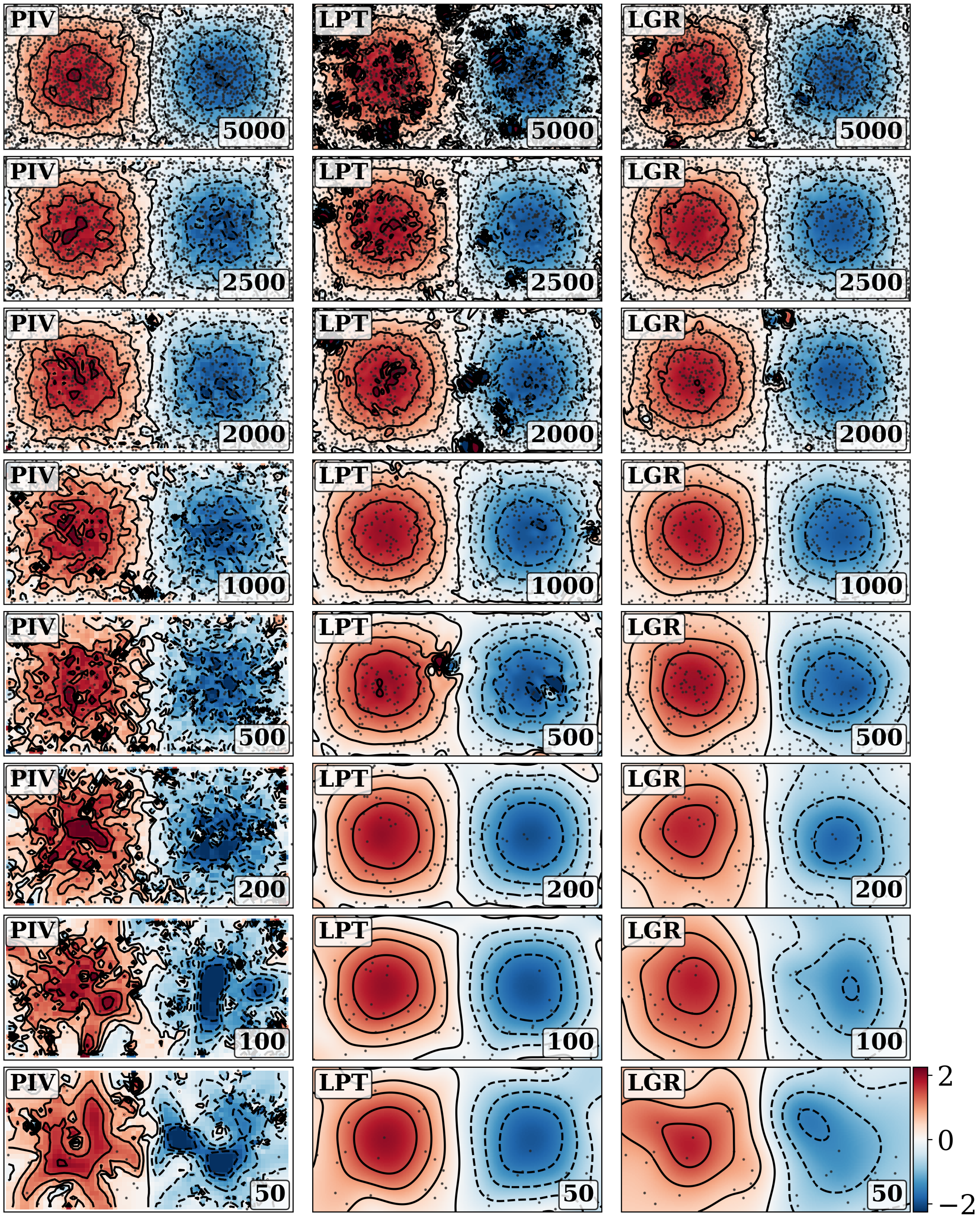


Figure 4.4: Visual comparison of vorticity computed using LGR with vorticity estimations from PIV and LPT. The column at left displays results from a standard PIV gradient pipeline. The center column displays vorticity computed using fields interpolated from LPT tracers. The right column displays fully Lagrangian vorticity estimations made by LGR. Solid and dotted curves indicate iso-contours of vorticity. The number of simulated particles is indicated in the lower right-hand corner of each frame.

$\omega_{z,\,pred}(t)$ from PIV, LPT, and LGR are compared in terms of absolute error from the true value $\omega_{z,\,true}(t)$

$$e = \left|\omega_{z,\,pred}(t) - \omega_{z,\,true}(t)\right|. \tag{4.2}$$

Absolute errors are computed for each recorded particle in every frame of the simulated videos for all three methods, ensuring that all samples occur on each approach. From the compiled data, the mean absolute error is estimated

$$\mathrm{MAE}(\omega) = \frac{1}{n}\sum_{i=1}^{n} e_i, \tag{4.3}$$

as are the standard deviation of the absolute error

$$\mathrm{S}(e) = \left(\frac{1}{n}\sum_{i=1}^{n}\left(e_i - \mathrm{MAE}(\omega)\right)^2\right)^{\frac{1}{2}} \tag{4.4}$$

and the quartile values. All of this information is tabulated for various particle densities in Table 4.1, and the MAE(ω) values are plotted in Figure 4.5.

The results of the comparative study on vorticity approximation admit a few observations. First, it is evident that the present implementation PIV performs best with many tracers, and that of LPT performs best with few. This is consistent with the mechanics of the two algorithms: when there are many tracers present, the correlation signal within a boxed PIV window is strong. Correspondingly, it is weak when there are few tracers observed. As a result, many spurious vectors are formed when PIV is performed on sparse sample data. LPT, on the other hand, is primarily limited by its ability to track independent trajectories. When there are many tracers observed in a flow, distinguishing between particles is challenging. The spurious trajectories that form from mis-associated tracer detections contain false velocities that corrupt the interpolated fields. That is why, for LPT, the frames with many simulated tracers are splotchy; the aberrations are the result of misidentified tracers leading to bad velocities and differentiation error when finite differences are used to compute vorticities. This is further supported by the large standard deviation that LPT experiences at high tracer counts.

A second observation is that LGR does not seem to be afflicted by the pock marks which plague the LPT results at high tracer counts, even though its analysis is based off of the same flawed trajectory detections. Once again, this is likely due to the mechanics of the algorithms. LPT involves two steps that amplify the error of bad trajectories through to velocity gradients: velocity field interpolation and numerical differentiation. On the other hand, the regression process which LGR relies on for gradients can serve to smooth out errant trajectory data by incorporating many neighboring particles and aggregating the effect of their motion. This is likely why LGR outperforms LPT for the large tracer count tests.

A final observation from the comparative study is that, while LGR repre-

Table 4.1: Absolute error statistics of vorticity for PIV, LPT, and LGR. Error analyses were computed using the Double Gyre flow with varying numbers of observable tracers simulated as PIV particles. The number of collected samples is displayed at left along with the mean and standard deviation of absolute error in vorticity and quartile values.

Particles	Samples	Approach	MAE(ω)	S(e)	25%	50%	75%
25	8569	PIV	0.478	0.491	0.153	0.349	0.662
		PTV	**0.173**	**0.137**	**0.043**	**0.103**	**0.238**
		LGR	0.397	0.356	0.143	0.316	0.539
50	21868	PIV	0.359	0.365	0.115	0.259	0.487
		PTV	**0.107**	**0.141**	**0.020**	**0.050**	**0.131**
		LGR	0.255	0.212	0.106	0.207	0.350
100	43189	PIV	0.299	0.305	0.095	0.213	0.405
		PTV	**0.058**	**0.103**	**0.012**	**0.027**	**0.060**
		LGR	0.186	0.150	0.072	0.152	0.265
200	86905	PIV	0.298	0.316	0.094	0.209	0.394
		PTV	**0.044**	**0.143**	**0.009**	**0.021**	**0.041**
		LGR	0.162	0.129	0.065	0.137	0.226
500	206759	PIV	0.246	0.331	0.075	0.165	0.307
		PTV	**0.054**	**0.199**	**0.010**	**0.024**	**0.047**
		LGR	0.100	0.106	0.032	0.068	0.131
1000	401867	PIV	0.176	0.250	0.051	0.113	0.216
		PTV	**0.057**	0.221	**0.011**	**0.024**	**0.047**
		LGR	0.077	**0.087**	0.025	0.054	0.100
2000	732684	PIV	0.112	0.171	0.034	0.073	0.136
		PTV	0.090	0.364	**0.015**	**0.034**	**0.071**
		LGR	**0.058**	**0.082**	0.018	0.039	0.072
2500	732684	PIV	0.100	0.143	0.031	0.067	0.123
		PTV	0.111	0.652	**0.016**	0.038	0.083
		LGR	**0.057**	**0.094**	0.017	**0.037**	**0.069**
5000	732684	PIV	0.074	**0.110**	0.024	0.052	0.091
		PTV	0.204	1.061	0.025	0.060	0.142
		LGR	**0.064**	0.161	**0.015**	**0.035**	**0.068**

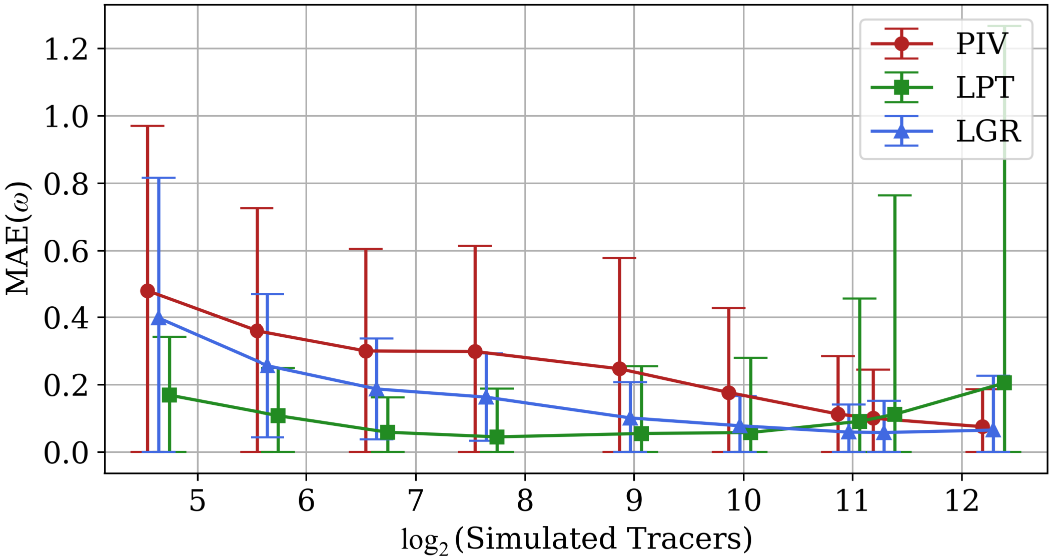


Figure 4.5: Absolute error statistics of vorticity approximated using PIV, LPT, and LGR. The base-2 logarithm of the number of particles is displayed on the abscissa and the mean absolute error (MAE) of vorticity is displayed on the ordinate with standard deviations clipped at zero.

sents improved performance over PIV at all tracer densities, it under-performs LPT for low tracer densities. This discrepancy may result from the specific implementation of LGR in this study. Regressions on tracer neighborhood deformations were performed using the 15 nearest neighbors for all particle densities. This means that, for low tracer densities, the spatial extent of the trajectory neighborhood is much larger than for higher tracer densities, possibly even spanning both vortices at once. Applying the radial-Gaussian kernel weighting helps alleviate this issue, but does not entirely mitigate it.

4.2 Spatial Scale Sensitivity of LGR

During the analysis of the previous section, the impact of the spatial distribution of tracers included in LGR regressions was alluded to. Indeed, understanding which spatial scales of the flow contribute to the gradient approximation along a trajectory is important when working with sparse or natural data, where dominant scales of motion may not be known *a-priori*. By examining the relative positions of regression tracers to the analyzed trajectory, LGR provides a mechanism for assessing the spatial scale sensitivity of computed gradients.

Consider, for example, two tracers embedded in a flow separated by some initial distance $r(t) = \|\Delta \mathbf{x}(t)\|_2$. The approximated gradients are defined by the relative motion of the tracers to one another, which can only be observed at a length scale proportional the initial separation of the two tracers. By the Nyquist-Shannon sampling theorem, flow movements smaller than $2r(t)$ will not be accurately sensed, and the lower-bound of physical scales that can be sensed by LGR is $2r_{min}(t)$. In other words, at least two tracers need to exist in a flow feature (e.g., a vortex) for it to be sensed by LGR. Movements at scales smaller than this will not be sensed and may contribute to measurement error.

Determining the mean contributing spatial scale of the tracer cloud motion to the regressed gradient is possible with the present description. This is most easily illustrated when the kernel-weighting matrix $\mathbf{K} = \mathbf{I}_d$, as the mean contributing scale is calculated as the mean distance to neighboring tracers

$$r_{avg}(t) = \frac{1}{n}\sum_{i=1}^{n} r_i(t), \tag{4.5}$$

where $r_i(t) = \|\Delta\mathbf{x}_i(t)\|_2$.

In light of this, the kernel matrix $\mathbf{K}$ may be viewed as a filter for which flow scales are sensed by the LGR operation. Weights can be tuned to highlight specific scales of motion in the flow. The effective scale contribution of kernel-weighted tracers can be calculated by considering the matrix of weighted tracer positions $\mathbf{Y}_t = \mathbf{X}_t\mathbf{K}$. Each column $\Delta\mathbf{y}_i(t) \in \mathbb{R}^d$ in this new array represents the kernel-weighted relative position of the tracer $\mathbf{x}_i(t)$. The proportional influence of each neighboring particle to the estimated gradient after weighting is given by

$$p_i(t) = \frac{\|\Delta\mathbf{y}_i(t)\|_2}{\sum_{j=1}^{n} \|\Delta\mathbf{y}_j(t)\|_2}, \tag{4.6}$$

such that $0 \leq p_i(t) \leq 1$ and $\sum_{j=1}^{n} p_i(t) = 1$. Multiplying this with the true physical distances $r_i(t)$ yields weighted distances, which can be averaged to compute the mean scale contribution under weighting

$$r_{avg,\,weighted}(t) = \sum_{i=1}^{n} p_i(t) r_i(t). \tag{4.7}$$

As an example of the filtering effect of kernel weighting on effective scale sensitivity, the radial implementation of the Gaussian kernel from equation 3.12 will always reduce the mean contributing scale of LGR to estimated gradients, as it preferentially weights tracers near the central particle.

Demonstration on PIV data

To explore the spatial sensitivity of LGR further, an analysis was performed on experimental flow field data collected on an asymmetric wake studied in the NOAH laboratory at GALCIT. The flow was recorded at a large enough Reynolds number to contain eddies at various scales of motion, making it a favorable candidate for examining the spatial sensitivity of LGR. Complete details regarding the experimental setup and parameters are contained in Appendix B Section B.4. An abbreviated description of the experiment is provided here.

The experimental flow is of a split-stream past a blunt trailing edge resulting in an asymmetric wake which evolves into a shear layer. Upstream blockage is placed to enforce a velocity ratio of $U_2/U_1 \approx 0.5$ where U_1 and U_2 are the free-stream velocities on either side of the plate. The splitting plate has thickness $h = 0.0572$ m and is centered in the NOAH water channel, which is 0.457m wide. For the case studied, the Reynolds number defined by plate thickness is $Re_h = U_{mean}h/\nu = 15.02 \times 10^3$ where $U_1 = 0.325$ ms^{-1}, $U_2 = 0.189$ ms^{-1}, $U_{mean} = 0.257$ ms^{-1}, and $U_2/U_1 = 0.58$.

PIV image acquisition was performed at the midline between the floor of the water channel and the free surface perpendicular to the plane of the splitting plate. Time resolved images were taken at 190 Hz for a duration of 19.35 seconds using two Phantom Miro Lab 320 cameras with pixel resolution of 1920 × 1200 px and overlapping fields of view. Neutrally buoyant particles were illuminated by a Photonics DM20-527(nm) YLF laser in single-pulse mode expanded through a cylindrical lens into a sheet. PIV processing details are provided in Appendix B.

The present analysis is performed on a single snapshot of the asymmetric wake flow. The fields of streamwise (U) and spanwise velocities (V) from the analyzed snapshot are displayed in Figure 4.6a and 4.6b, respectively. The axes of the flow are nondimensionalized by the thickness of the plate, h. The trailing edge of the plate is flush with the left edge of the plot boundary. The faster flow U_1 is on top of the plate, as viewed in the orientation of Figure 4.6. Hence, the characteristic shear layer vortices form with clockwise rotation.

Gradient computations were performed using finite-differences as a baseline and using LGR with numerical tracers integrated by the pre-computed velocity fields. Gradients are represented by vorticity as computed by equation 4.1. The result of the finite-difference vorticity field is presented in Figure

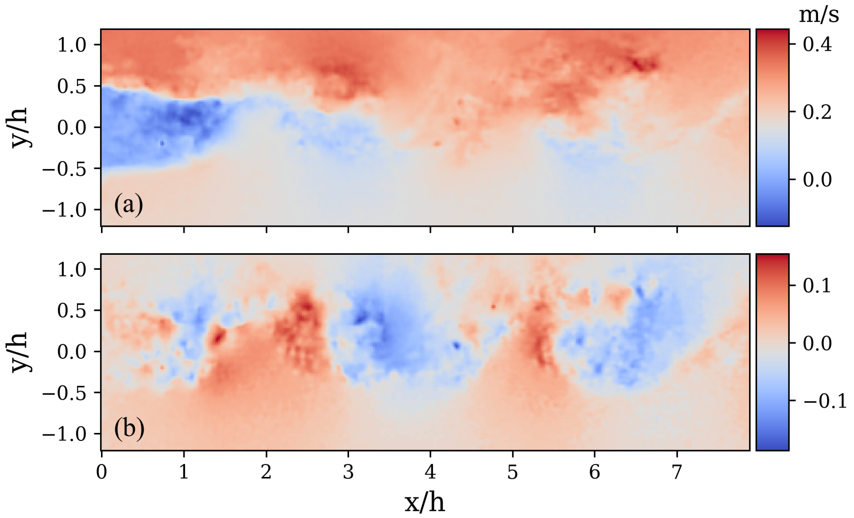


Figure 4.6: A snapshot of PIV-computed velocities from the asymmetric wake experiment discussed in Appendix B section B.4. (a) Streamwise U-velocity from left to right. (b) Spanwise V-velocity from bottom to top.

4.7a. LGR was implemented using numerical particles seeded according to the planet-satellite scheme which was previously discussed. Central particles were seeded on uniform grid defined by the roots of the PIV velocity vectors so that one-to-one comparisons could be made. For each central particle, four satellite particles were seeded around it at constant radius and with equal azimuthal spacing. All tracers were integrated forward in time over one snapshot with duration $\Delta t = 1/190$ s. To examine the influence of initial particle displacement on the computed gradients, the radius from central particles to satellite particles is varied such that $|\Delta \mathbf{x}_0|/h = [0.01, 0.25, 0.75]$. All of the regressions use the identity kernel $\mathbf{K} = \mathbf{I}_2$.

The vorticities computed by LGR with $|\Delta \mathbf{x}_0|/h = 0.01$ are presented in Figure 4.7b, those with $|\Delta \mathbf{x}_0|/h = 0.25$ in Figure 4.7c, and those $|\Delta \mathbf{x}_0|/h = 0.75$ in Figure 4.7d. The schematics to the right of Figures 4.7b-d illustrate the spacing of tracers involved in LGR regressions relative to the spacing of the grid of sample locations. The circle displayed on each frame and associated with the schematics represents the approximate to-scale spacing of tracers used for LGR in that image. For Figure 4.7b, the particle spacing is smaller than the length scale of any motion captured by the PIV, in Figure 4.7c, the spacing is larger than some of the scales of motion, and in Figure 4.7d, the spacing is larger than most scales of fluid motion.

The vorticity computed from the asymmetric wake experiment velocity

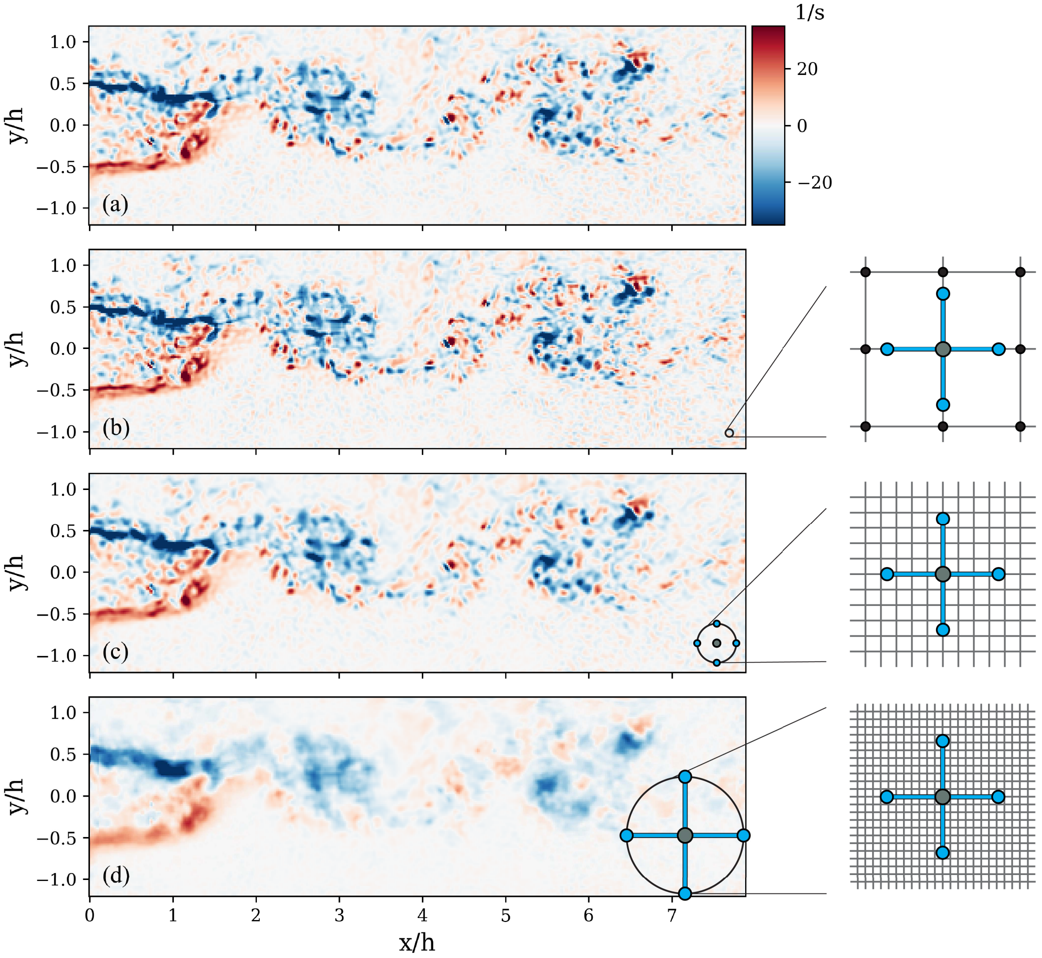


Figure 4.7: Vorticity computed from the experimental velocity fields presented in Figure 4.6. (a) Vorticity computed directly using finite differences. (b-d) Vorticity computed using LGR on deformations of numerical tracers varying initial tracer separation. (b) Initial separation $|\Delta\mathbf{x}_0|/h = 0.01$. (c) Initial separation $|\Delta\mathbf{x}_0|/h = 0.25$. (d) Initial separation $|\Delta\mathbf{x}_0|/h = 0.75$. Illustrations at right approximately indicate initial tracer separation relative to sample grid spacing.

fields highlight important properties of LGR. First, the gradients regressed from a small neighborhood of particles in Figure 4.7b are indistinguishable from those computed by finite-differences (Figure 4.7a). This is unsurprising, since the particles used in the regression were advected on the same velocity fields which were used for numerical differentiation; the result affirms what was already discussed in section 4.1.

Vorticity results displayed in Figures 4.7c and 4.7d are obtained from fluid deformations occurring at flow scales larger than the smallest scales in the flow. Therefore, their results shed light on the spatial sensitivity of LGR. The vorticity features in frames 4.7c and 4.7d do not express the same sharp detail present in frames 4.7a and 4.7b. In other words, the tracer spacing acts as a kinematically defined low-pass filter on the recorded gradients, unable to resolve scales of fluid deformation smaller than the separation of the tracers.

The results of this spatial sensitivity analysis demonstrate that velocity gradients computed by LGR are dependent upon the distribution of neighboring particles and the weights that are associated with them in the kernel matrix $\mathbf{K}$. The initial distribution of particles in a regression neighborhood dictates which scales can be resolved in the outcome. Roughly speaking, if the majority of tracers in that neighborhood exist on a feature in the flow, then it will be detected by LGR. The weights of the kernel matrix influence the outcome by functionally assigning preference to regions of the tracer neighborhood.

4.3 Estimating Finite-Time Metrics using LGR

The evaluation of LGR thus far has focused on approximating velocity gradients instantaneously in time. In this section, its ability to perform finite-time analyses is evaluated. Such analyses come in two varieties: first, there are those based on the flow map Jacobian (equation 2.8) expressed over finite intervals. In the work at present, the only such analysis being considered is the FTLE, which is common to Lagrangian coherent structure (LCS) literature. The second variety of finite-time analysis are those which are integrated along tracer trajectories. Four such metrics are considered here, the LAVD and DRA, which were recently developed by Haller and collaborators [51], [88], and the TISM and kinematic action (KA), which are novel contributions of this thesis. All five of these analyses are available through the framework of LGR, and will be evaluated in this section for use on randomly distributed, sparse data.

A majority of the following examples (with the exception of Figure 4.8) utilize a consistent implementation of LGR. Tracers of varying density are propagated through the Double Gyre flow (equation 2.30 with the associated parameters) on the spatial domain $[0, 2] \times [0, 1]$ from $t = 0$ to $t = 15$. This allows for direct comparison to results presented earlier in the paper and ease of method evaluation. Neighboring tracers are resampled every $\Delta t = 0.1$. Regression between snapshots is performed using $k = 15$ nearest neighbors with the radial-Gaussian weighting function with standard deviation $s = 0.125$ (equation 3.12). The particle locations where values are computed are indicated in each figure as gray dots. Interpolation for the purpose of visualization occurs in between the points. Radial basis function (RBF) interpolation with a multiquadric kernel is used to compute field values on a 200×100 uniform grid in x and y.

Baseline Comparison of FTLE Approximation Methods

Of the five metrics discussed in the introduction to this section, the finite-time Lyapunov exponent (FTLE) is by far the most commonly used in LCS applications. It was the first of the objective LCS detection strategies developed, and has therefore been cemented as a canonical analysis in the field [39], [61]. As discussed at length in Section 3.2 of Chapter 3, the standard approach for estimating FTLE fields requires the numerical integration of a dense, regular grid of tracers over the finite time interval. As the duration of the interval increases, so must the density of the regular grid of tracers. With Figure 3.3, it was shown how increasing tracer sparsity corrupts FTLE results as motivation for the necessity of sparse Jacobian estimation methods. Here, a similar analysis is performed and compared with corresponding results using LGR.

To compare the FLTE results of LGR with traditional approximation methods, four estimation methods are applied to the same flow, sampling at the same initial conditions in all cases. This is possible by interchanging the approximation mechanism into the planet-satellite approximation scheme developed in Section 3.2 of Chapter 3. The Double Gyre flow is used as the test case in this example, where deformations are recorded over the interval $t \in [0, 15]$. Only forward FTLE is considered, therefore results are displayed for $t = 0$. A fine 200×100 particle sampling grid is defined in x and y, providing the test locations for each method. The results of the analysis are presented in Figure 4.8.

The baseline FTLE field is constructed by estimating the flow map Jacobian $D\mathbf{F}_0^{15}(\mathbf{x}_0)$ at each initial position using finite-differences with initial tracer spacing $|\Delta\mathbf{x}| = 10^{-6}$ and no tracer resampling at intermediate intervals. For this flow and duration, such spacing is sufficiently small to ensure that deformation over the 15 unit time interval remains linear regardless of where particles are initially placed. Thus, the results are trustworthy as a baseline. They are displayed in Figure 4.8a. Reviewing the data, all of the expected features of the FTLE are present, including a dominant codimension-1 ridge that traverses the domain. Because the tracer spacing is smaller than the spacing of sampling locations, the ridge appears to be aliased. This will be discussed in greater detail in future sections.

To provide contrast with LGR, a finite-differencing case without intermediate tracer resampling is presented in Figure 4.8b. This analysis is essentially identical to that shown Figure 4.8a, only, the initial spacing of the tracers is made to be $|\Delta x| = 0.1$, which is exceedingly large for this flow. As a result, significant features of the flow are lost, including the FTLE ridges which form the LCS features.

The first example of FTLE computed using LGR is provided in Figure 4.8c. As before, computations represent the duration from $t \in [0, 15]$, only now, intermediate Jacobians are computed every $\Delta t = 0.25$ units on tracers that are resampled at each interval. For this case, four regression tracers are resampled at each time step on the axes of the flow with spacing of $|\Delta \mathbf{x}| = 0.1$. Intermediate flow map Jacobians (i.e., those computed every 0.1 time units) are computed by finite differences and synthesized by equation 3.16 to estimate the Jacobian over the full $t \in [0, 15]$ interval. To be clear, the initial position of all tracers in this example are exactly the same as the initial positions presented in Figure 4.8b.

Finally, FTLE computed using the regression variant of LGR is presented in Figure 4.8d. This experiment only varies from the results of Figure 4.8c in the means of resampling and regression. At each time step, ten resampled particles are placed uniformly at random inside a radius $r = 0.1$ of the center tracer. Regression is then used to compute the Jacobian with kernel weighting defined by a Gaussian function on the radial distance from the central tracer location (equation 3.12) using standard deviation $s = 0.125$.

The results of this experiment demonstrate the effectiveness of LGR for accurately reconstructing the flow map Jacobian from sparsely distributed particles. Where the traditional computational strategy of Figure 4.8b does not capture the fine details or the expected ridges seen in the baseline, both approaches using LGR closely match the true values. The FTLE ridges that are typically used for identifying LCS are clearly present and identifiable, and all of the values in the field are commensurate with the baseline. When traditional methods are applied to sparse tracer patterns, as in Figure 4.8b, it is not evident that any FTLE ridges exist and the higher values in the field are significantly diminished.

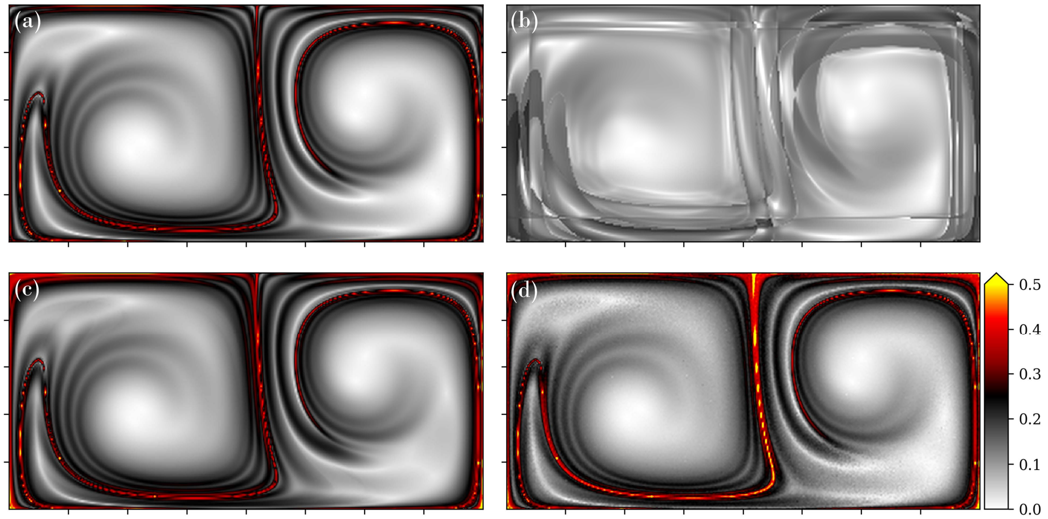


Figure 4.8: Performance comparison of FTLE computation schemes on the unsteady double gyre flow over the interval $t \in [0,\, 15]$: (a) Baseline: FTLE computed using finite-differences with no particle replacement for $|\Delta\mathbf{x}| = 10^{-6}$ by the planet-satellite method (Figure 3.2(c.i)). (b) Same as (a), but with large $|\Delta\mathbf{x}| = 0.1$. (c) FTLE computed using LGR with neighboring particle replacement every $\Delta t = 0.25$ with spacing $|\Delta\mathbf{x}| = 0.1$. Sampled tracers are placed on the axes of the flow, and intermediate Jacobian regressions are achieved by finite-differences. The full-time Jacobian is computed using equation 3.16 over all intermediate computations. (d) Same as (c), only estimating gradients are estimated using regression over 10 tracers randomly sampled within a radius of $r = 0.1$ from the trajectory (Figure 3.2(c.ii)). Regression uses Gaussian weighting based on the radial distance from the trajectory.

Evaluating Spatial and Temporal Sampling Parameters

Now, the analysis from the previous section is extended to consider the impact of varying resampling radius (sparsity) and frequency on Jacobian estimation and FTLE results. A numerical experiment was devised to sweep through many resampling radii and resampling rates and compare their results to a baseline test. Once again, all data is collected on the Double Gyre flow over the interval $t \in [0,\, T]$, where $T = 15$. Baseline FTLE results are computed using the planet-satellite method with $r_0 = \|\Delta\mathbf{x}\| = 1 \times 10^{-6}$, which was the same approach taken in Figure 4.8a. Initial tracer positions are situated on a 50×25 element uniform grid so that all regions of the flow may be sampled.

Spatial sampling is designed to vary such that $r = r_0 10^p$ where p varies at increments of $p = 0.5$ over $p \in [1,\, 5.5]$. This equates to varying over $r \subset$

$[10^{-6},\ 0.316]$ in the units of the flow. Temporal resampling satisfies $\Delta t = T/2^q$ for $q \in [0,\ 10]$ integers. In this way, the number of times tracers are resampled doubles for each increment in q. In flow units, this equates to $\Delta t \subset [0.014,\ 15]$. Using these resampling conditions, flow map Jacobians are computed using LGR for the same duration and initial conditions as the baseline data.

To compare results with the baseline, the mean absolute error (MAE) between LGR computed FTLE and the baseline is computed

$$MAE(\sigma_{t_0}^{T}(\mathbf{x}_0)) = \left|\sigma_{t_0,\,LGR}^{T}(\mathbf{x}_0) - \sigma_{t_0,\,true}^{T}(\mathbf{x}_0)\right|. \tag{4.8}$$

Absolute error is chosen over relative error since the numerical error is not likely proportional to the true value of the FTLE. Moreover, FTLE is chosen as the comparison metric since it is based on the maximum singular value of the flow map Jacobian $\left\|D\mathbf{F}_{t_0}^{T}\right\|_2$, and is therefore representative of the accuracy of the Jacobian as a whole. Results showing the MAE for all resampling conditions are provided in Figure 4.9a. To determine for which conditions resampling is more accurate than not resampling, the MAE of FTLE values computed with resampling is subtracted from the MAE computed without resampling for the same sparsity level (i.e., values of r). These results are presented in Figure 4.9b.

Some general observations are made from the results presented in Figure 4.9. First, when sampled sufficiently quickly, FTLE results produced by resampling yield comparable results to the baseline. The conditions highlighted inside the black rectangle exhibit relatively low absolute error, even when the sampling radius for calculations is large relative to the baseline. As sampling radius decreases and the number of intervals increases, the accuracy of the results improves.

Second, it is evident that computing Jacobians through LGR with resampling yields increased accuracy over the conventional method. This is indicated in Figure 4.9b by the cases colored blue in the upper right-hand corner of the frame. These display FTLE values nearer to the true values than when computations are performed on the same initial tracers without resampling (the far left column in Figure 4.9a). As context, the results observed in Figure 4.8c,d were recorded for $p = 5$ and $q = 5.91$.

Another observation is that resampling does not improve accuracy if the rate of resampling is not sufficiently faster than the unsteady time-scales of the flow. For LGR to be effective, the deformations observed between time steps

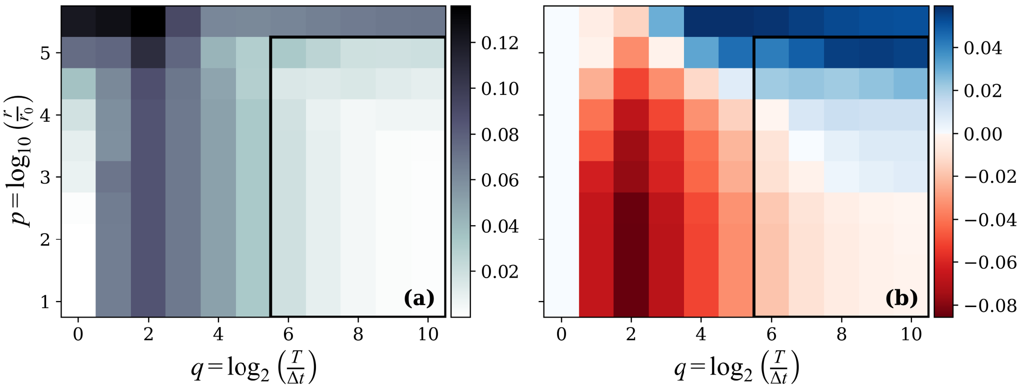


Figure 4.9: Evaluation of spatial and temporal sampling frequency on FTLE accuracy. All results are computed on the Double Gyre flow over the total interval $t \in [0,\, T]$ where $T = 15$. Baseline results (represented in the lower left corner of (a)) are equivalent to those presented in Figure 4.8a where sampling radius $r_0 = 10^{-6}$. Sampling radius varies in increments of $\Delta p = 0.5$ for $p \in [1, 5.5]$ such that $r = r_0 10^p$. Resampling frequency is varied such that the time between resampling for regression varies with intervals of $\Delta n = 1$ for $q \in [0,\, 10]$ such that $\Delta t = T/2^q$. (a) Mean absolute error (MAE) of FTLE computations with the baseline. (b) MAE of FTLE computed by resampling subtracted from MAE of FTLE computed without resampling (first column of (a)) for the same value of r. The region inside black rectangles represents conditions where relatively accurate results are obtained from sparse data.

must be approximately linear. In the examined case, for $q < 5$, the deformations over the intermediate intervals were dominated by nonlinearity, and thus yielded low FTLE accuracy even for small sampling radii. Thus, one should not consider using LGR to compute Jacobians if data cannot be recorded at sufficiently high rates. However, the nature of data collection in many practical flows examined in experimental fluids and physical oceanography suggests that this limitation will not often be of concern.

There are, of course, some caveats to this analysis. First, the tests in this study use a very controlled form of resampling. When operations are performed using nearest-neighbor detection rather than numerical sampling, the changing shape of regression polygons will likely impact the outcome. Second, the Double Gyre is a periodic flow that recycles many tracers back to proximity with their initial positions regularly. This, too, will likely shape the nature of errors observed in the FTLE field. Finally, this statistical analysis does not condition results based on the true value of FTLE. That is, errors computed near to FTLE ridges are weighted equally to those in FTLE basins.

It is therefore important also to consider the fields themselves as in Figure 4.8 to determine if ridges can, indeed be identified from sparse data.

FTLE Evaluation on Sparse Data

In the next demonstration, FTLE fields are computed directly from randomly distributed tracers using Algorithm 2, which is described in Appendix C. The results are compared with a baseline and with the approach of Mowlavi et al. [60] which was designed specifically for sparse and noisy data. The baseline approach computes Jacobians by regressing directly between initial and final states with no resampling and using only differences between the examined particle and its neighbors in the regression. The work of Mowlavi et al. [60] expands on the baseline by augmenting the regression matrices with additional inter-particle connections. Rather than just incorporating the differences to the central tracer, the differences between all particles within a neighborhood are included in the regression. Aside from that change, the algorithm remains the same: regression still seeks to fit a linear deformation between initial and final times only. All three implementations identify the same $k = 15$ nearest neighbors at time $t = 0$ and perform analyses on the deformation up to $t = 15$. For the conventional approach and that of Mowlavi et al. [60], the Jacobian is computed using unweighted ordinary least-squares regression ($\mathbf{K} = \mathbf{I}_d$, $\gamma = 0$ in equation 3.11).

Results of the comparative FTLE analysis are presented in Figure 4.10. The first column of frames displays results obtained using LGR, the second column provides the baseline results acquired by standard FTLE methods, and the third column contains FTLE fields resulting from the method of Mowlavi et al.. Particle distributions are introduced from top to bottom of Figure 4.10 by decreasing tracer density, where the exact number of tracers in the frame is indicated in the lower right-hand corner. The densest data contains 2500 complete trajectories, followed by 1000, 500, 250, and finally, 100 trajectories. As a point of context, FTLE computations on this flow often use over 10^6 tracers [41]. Thus, these representations can be considered sparse relative to typical computations. The coloring of the FTLE fields is consistent between frames and was chosen to reflect the true values computed in Figure 4.8a.

The performance improvement of LGR over other gradient approximation methods on sparse data is evident in Figure 4.8. In all degrees of sparsity considered, FTLE via LGR displays evidence of the principal FTLE ridge

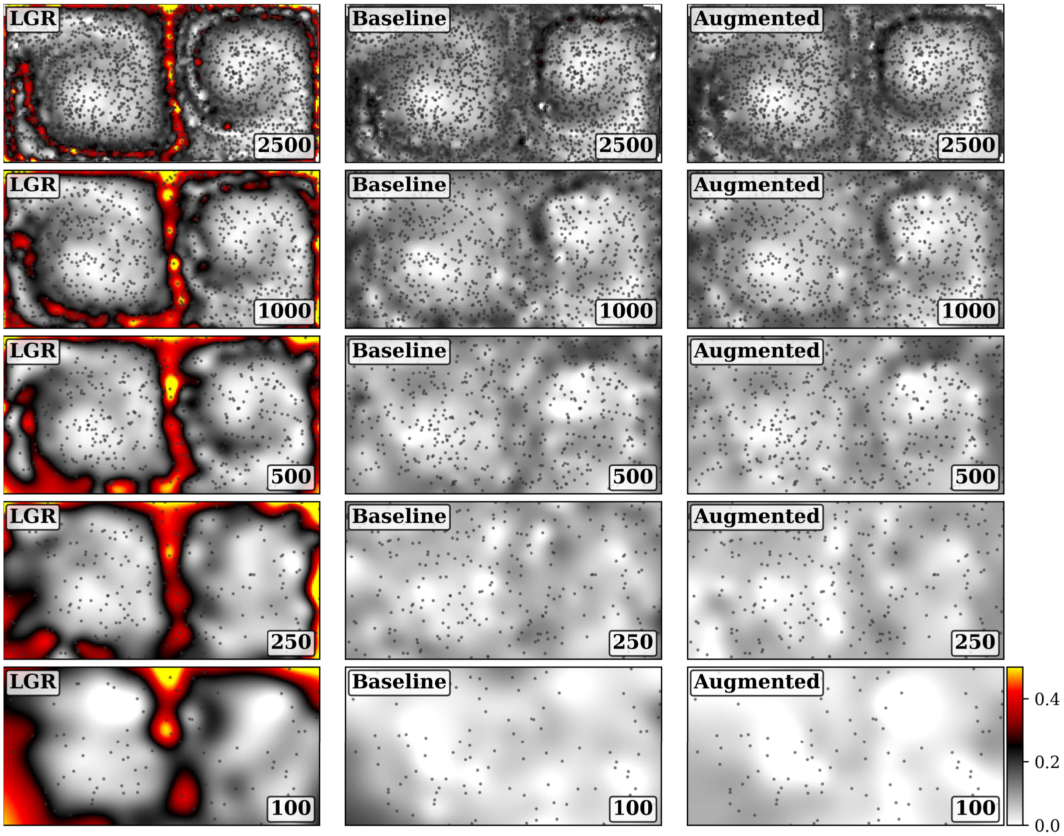


Figure 4.10: FTLE computed sparsely and randomly placed particles using LGR (left column), using only the initial and final positions of particles in the conventional approach (center column), and augmenting the data matrices according to Mowlavi et al. [60] (right column). Particles are advected over $t \in [0, 15]$ with resampling at $\Delta t = 0.1$. Jacobian computations use radial Gaussian weighting on the $k = 15$ nearest neighbor particles with standard deviation of 0.125. The number in the lower right-hand corner of the frames indicates the number of tracers used in computations, and the gray markers indicate their evaluation position at $t = 0$.

down the center of the domain. Moreover, the FTLE values that it identifies are commensurate with the values of the true field shown in Figure 4.8a. In contrast, only shadows of the FTLE ridges appear when other approaches are used, and those only when the field is more densely seeded. In the sparsest cases considered (500, 250, and 100 tracers), there is no evidence of dominant ridges anywhere in the field. It is thus concluded that LGR outperforms other available tools for sparse FTLE estimation, and can be effectively used as a tool for identifying hyperbolic LCS in sparsely observed flows.

Finite-Time Strain Metrics

The FTLE value of a trajectory represents the exponential growth rate of a linear deformation over the observation interval of the trajectory. Essentially, it discounts all deformations that occur during the time-history of the trajectory and only relates its initial and final states. In Section 3.5 of Chapter 3, the trajectory-integrated strain magnitude (TISM, see equation 3.18) was introduced as a quantity which measures the total strain experienced by a fluid element along a trajectory. In this section, FTLE and TISM fields are computed using LGR from the same set of trajectories and compared.

Figure 4.11 contains visualizations of FTLE and TISM results computed from trajectory data generated using the Double Gyre flow. The left column contains FTLE data, and the right column contains TISM data. Computations are performed over the interval $t \in [0, 15]$ with the common flow parameters used throughout this document (See equation 2.30 and the associated discussion.).

Based on the results of Figure 4.11, some observations on the nature and practicality of FTLE and TISM can be made. First, one must recognize the difference in the scaling of the two metrics. FTLE represents an exponential rate of growth, and is relatively small compared to the TISM values, which represent integrated strain magnitude. As a result the nature of the fields—while both directly related to strain—are slightly different. For instance, FTLE can accommodate negative values, whereas TISM cannot. If a trajectory is embedded in a heavily contracting flow, its exponential growth rate, and therefore its FTLE, will be negative. TISM, on the other hand, will represent both contraction and dilation as positive values. Thus, in flows that demonstrate compressibility effects (or, as shall be seen in the next chapter, those which are observed by imperfect tracers), FTLE and TISM values may not share the same sign.

Compressibility effects like dilation and contraction are not a concern with the Double Gyre, however. In this particular example, the flow is incompressible, which means that FTLE values must remain positive over any finite-time interval. Therefore, if a ridge exists in the FTLE field of an incompressible flow, then it will correspond to a ridge in the associated TISM field. However, the relationship does not go the other direction. Large values of TISM do not necessarily indicate large values of FTLE. A simple example would be any flow that stretches in one direction along a ridge, and then reverses to its

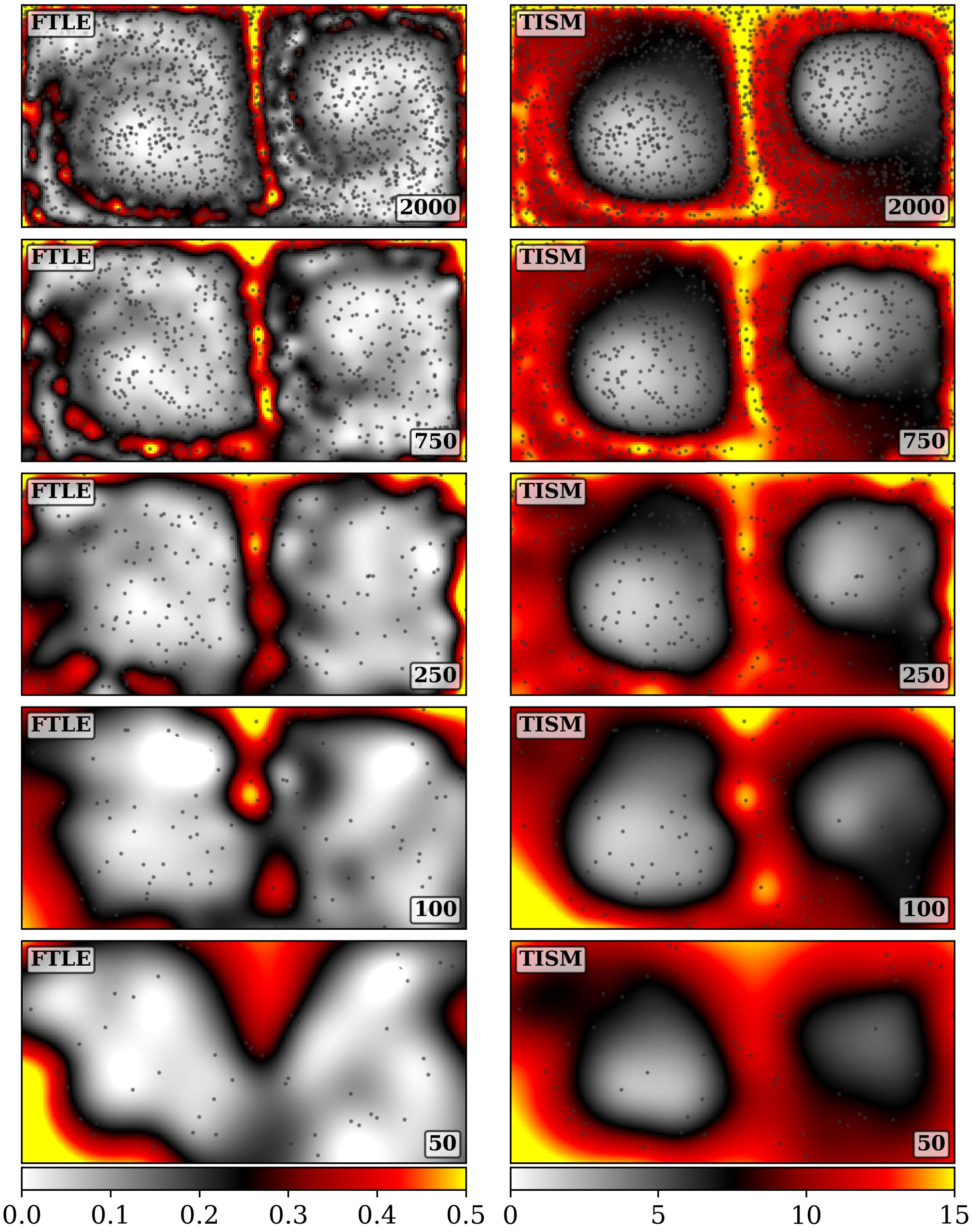


Figure 4.11: Comparison of flow stretching metrics FTLE and TISM as computed by LGR. Computations are performed over the interval $t = [0., 15]$ with increment $\Delta t = 0.1$ on the Double Gyre flow. Regressions of deformations include the 15 nearest neighbors of each tracer weighted by radius using a Gaussian function with standard deviation of $s = 0.125$. The number of simulated particles is indicated in the lower right hand of each frame, and the location of those particles at $t = 0$ are indicated by gray dots. Sparse values are interpolated to a field using RBF interpolation and colored for visualization purposes.

original state. TISM would accumulate strain over the entire history of the flow, whereas FTLE would mark as zero since no net deformation occurred. Therefore, while FTLE ridges will be represented in the TISM field, TISM ridges may not exist in the FTLE field.

Using sparse data, TISM may be a useful indicator of hyperbolic regions in a flow. As evinced by Figure 4.11, the high-strain regions of the flow are more diffusely represented by TISM than by the sharp FTLE ridges. With sparse data, sharp FTLE ridges are difficult to identify. With 100 and 50 tracers, for example, the ridges of the FTLE field are difficult, if not impossible, to discern. However, the TISM results for those densities indicate regions of high strain where the FTLE ridges ought to exist. Therefore, for more sparse data, TISM may prove usable where FTLE is not.

Finite-Time Rotation Metrics

While metrics quantifying particle dispersion such as FTLE are commonly computed in LCS analyses, they may be less practical for the purposes of AMFM measurements. One of the most prominent limitations of all Lagrangian methods is the ability to observe tracers over finite durations. Since FTLE ridges—the LCS feature pursued by FTLE and other like studies—are defined by rapid separation of tracers, they naturally exist in regions of the flow that are difficult to measure over long periods of time. In other words, they represent the most chaotic parts of the flow. For the purposes of autonomous feature tracking, seeking elliptic LCS (i.e., those defined by coherent material rotation) may be more practical.

Lagrangian vortices occur in between the ridges defined by hyperbolic LCS. They exist in the most stable portions of the flow. Tracers caught inside a vortex are likely to stay with that vortex over long intervals, making it a convenient tracking objective. Of course, the specific definition of a vortex is contested in the fluid mechanics community. For the purposes of this work, any material volume that exhibits coherent rotation over a finite interval is loosely referred to as a vortex. Such features can be efficiently identified by the metrics which have been introduced in Chapters 2 and 3. The purpose of the following analysis is to assess the merits of computing Lagrangian-averaged vorticity deviation (LAVD), dynamic rotation angle (DRA), and kinematic action (KA) from sparsely distributed random trajectories using LGR. LAVD and DRA were both developed by Haller and collaborators precisely as a means for identifying elliptic LCS in complex flows [51], [88]. KA was introduced in Chapter 3 as an extension to LAVD which simplifies the vortex identification criteria. LGR provides a convenient mechanism for computing all three quantities, which is codified in Algorithm 3 in Appendix C.

Figure 4.12 displays LAVD, DRA, and KA as computed by LGR for various tracer densities. As indicated, the left column presents LAVD, the center column DRA, and the right column KA. Particle densities ranging from 2000 tracers down to 25 tracers are presented in the figure. The exact number of tracers used in a simulation is indicated in the lower right-hand corner of each frame. As with prior examples, LGR is computed using 15 nearest neighbors along each trajectory with radial-Gaussian weighting using a standard deviation of 0.125. Fields are interpolated using RBF interpolation from scattered values recorded at particle positions. Coloring is based off of the ground truth values presented in Chapters 2 and 3. In frames depicting DRA, a solid black curve is plotted at the DRA = 0 contour. White curves are plotted in KA frames at the KA = 0 contour.

The results presented in Figure 4.12 demonstrate that it is possible to accurately evaluate elliptic Lagrangian metrics from tracer data with no velocity or velocity gradient information explicitly known *a-priori*. Moreover, elliptic LCS features are seen to be more robust to sparsity over hyperbolic ones. For the higher densities of tracers, LAVD, DRA, and KA closely resemble their dense counterparts from Chapters 2 and 3. As the tracer density decreases, the form of the structures in the field remains consistent so that, even with as few as 50 tracers in the frame the peaks and boundaries of the two vortices are still approximately observed.

In general, the LAVD and DRA have a convenient physical interpretation—they quantify the amount of rotation that a particle experiences over the integration interval. In Figure 4.12, the units of LAVD and DRA are presented in radians. As can be seen in the high particle density data, the absolute values of the two fields are nearly identical.

Though they both convey similar information, LAVD and DRA each have unique benefits relative to the other. As discussed in their introduction in Chapter 2, LAVD is always objective, whereas DRA is only objective under certain circumstances. Therefore, LAVD may be preferred when the relationship of the observer to the flow is not known. However, because DRA is signed, it enables convenient classification of distinct vortices in a flow. The

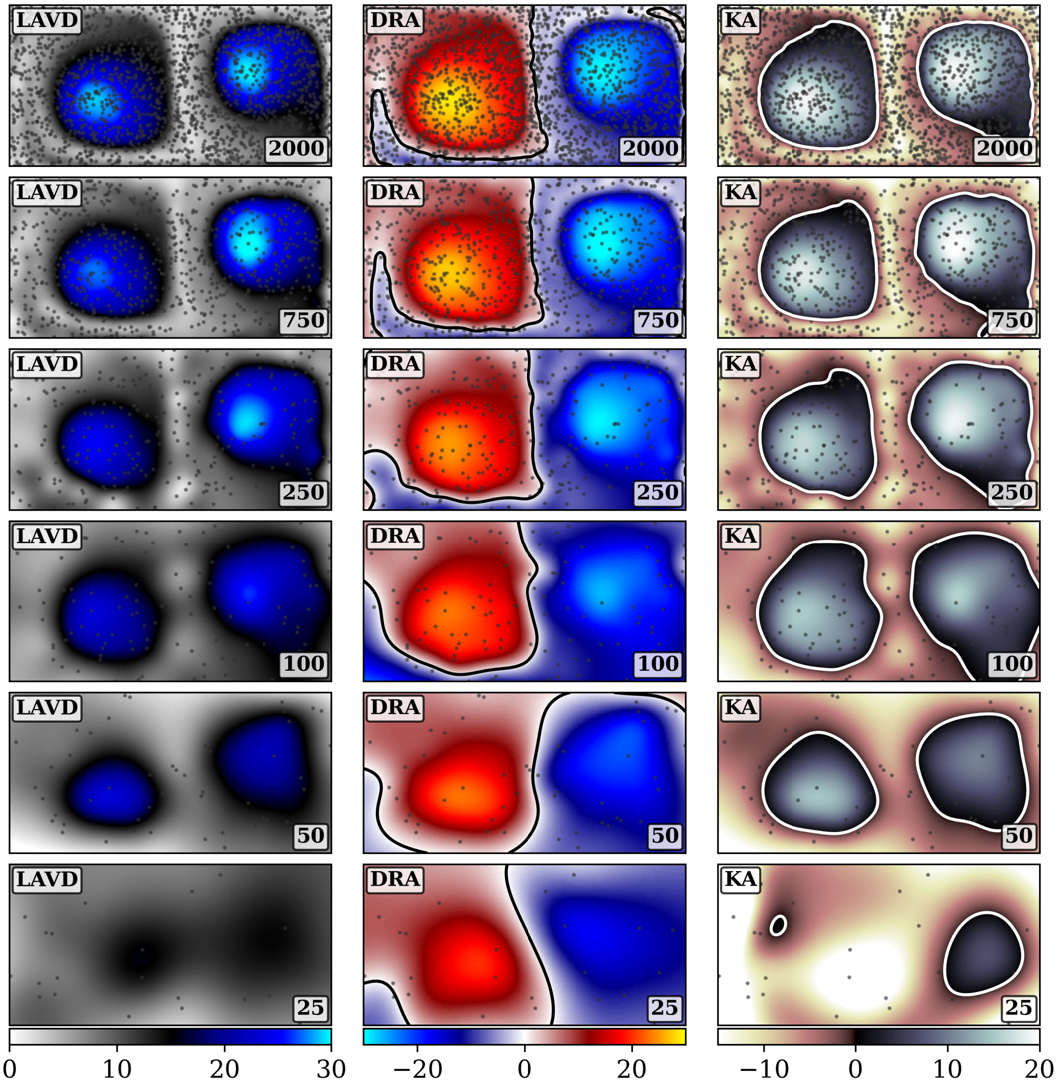


Figure 4.12: Comparison of integrated flow rotation metrics LAVD, DRA, and kinematic action (KA) as computed by LGR. Computations are performed over the interval $t = [0., 15]$ with increment $\Delta t = 0.1$ on the Double Gyre flow. Regressions of deformations include the 15 nearest neighbors of each tracer weighted by radius using a Gaussian function with standard deviation of $s = 0.125$. The number of simulated particles is indicated in the lower right hand of each frame, and the location of those particles at $t = 0$ are indicated by gray dots. Sparse values are interpolated to a field using RBF interpolation and colored for visualization purposes. For DRA fields, the zero-contour is indicated by a black line, and for KA fields, the zero-contour is indicated by a white line.

zero-contour of the interpolated DRA field acts as an approximate and easily identified hyperbolic LCS, continuously dividing any observed vortices. Thus, since DRA records the direction of rotation while LAVD and KA do not, it is best able to capture the true nature of the underlying field from the sparsest data (25 tracers) presented in Figure 4.12.

While DRA and LAVD are both useful for identifying the vortical regions in a flow, neither provides a convenient approach for approximating vortex boundaries from sparse data. The typical approach for vortex boundary detection in LAVD fields is to identify closed level-set contours which satisfy certain convexity constraints. This process requires data to be interpolated to a field and that the results preserve the contours of the true field. From the results displayed in Figure 4.12, it is clear that, the sparser data ($n < 750$ particles) does not satisfy these criteria. Thus, identifying vortex boundaries from sparse, Lagrangian LAVD and DRA data is challenging and subjective.

However, by juxtaposing LAVD with TISM (see equation 3.23), KA provides a cheap and convenient heuristic for determining tracer membership in a vortex. If, for a tracer, KA is computed to be KA > 0, then it its motion is dominated by rotation and it can be considered to be inside some vortex boundary. Otherwise, it is in a region of the flow that is dominated by strain. The cutoff threshold can be easily adjusted to accommodate stronger or weaker. Because it does not require additional constraints or interpolation, KA may be favorable to LAVD and DRA for identifying certain Lagrangian flow features.

4.4 Discussion of Results

Throughout this chapter, LGR has been applied in various scenarios to evaluate its capacity for estimating flow gradients. In this section, highlighted points from the results are discussed.

LGR Yields Reliable Flow Gradients from Sparse Trajectories

The first discussion point is the most straightforward: LGR is a viable tool for estimating both instantaneous velocity gradients and finite-time flow map Jacobians. This has been demonstrated through a variety of examples where common flow metrics are accurately represented using the equations and algorithms of Chapter 3. Section 4.1 demonstrates that LGR can reliably approximate velocity gradients by showing that estimated results match the analytical ground truth closely, even for relatively few observed trajectories. In Section 4.3, FTLE, LAVD, and other integrated metrics computed by LGR are seen to closely match with their baseline estimates. Importantly, when trajectories are sparse, LGR provides a significant improvement to flow map Jacobians and FTLE results over other methods used in the field.

LGR is a Capable Tool For Particle Image Experiments

The comparison of LGR with PIV and LPT, which featured prominently in Section 4.1, indicates that LGR is a viable tool which could be applied directly to experiments in fluids. Given the particular flow and the specific processing parameters used, LGR provides results that are as accurate or better than the corresponding results from PIV, LPT, or both for all tracer densities tested.

A significant implication of these results is that LGR could be used to dramatically decrease the storage burden of modern LPT studies while still retaining the ability to compute gradients. Modern LPT studies may track tracers over large domains that need to be interpolated onto a fine Eulerian grid in order to compute gradients. If the study is performed on a large, 3D domain, then all components of velocity, position, and any computed quantities like gradients, must be stored at all $N_x \times N_y \times N_z \times N_t$ grid locations in space and time, which can easily become unwieldy. LGR allows for gradients to be computed in a purely Lagrangian fashion, meaning that data need only be stored at only $N_p \times N_t$ positions. For context, in a 3-D domain that is 100 cells cubed, an Eulerian grid must save at one million locations, even if far fewer tracers are observed in that space.

LGR may also exhibit reduced computation times over methods like LPT. Assuming that trajectories are already known, the computational complexity of LGR is limited by the K-nearest neighbors operation, which is $\mathcal{O}(n \log n)$ using efficient data structures, where n is the number of particles. On the other hand, LPT is limited by the approach used for scattered interpolation. Efficiently implemented linear scattered interpolation is $\mathcal{O}(m \log n)$, where m is the number of query points (grid locations). More sophisticated interpolation schemes like the RBF interpolation approach used in the examples of Section 4.1 are more computationally expensive. For example, the RBF interpolation is $\mathcal{O}(n^3)$. Therefore, in circumstances where the number of interpolation grid points is larger than the number of particles, LGR has the potential to outpace LPT gradient approximations even for simple interpolation schemes.

A final advantage of LGR in the context of particle image experiments is that it naturally provides an extension to Lagrangian analyses where PIV and LPT do not. For both PIV and LPT, additional interpolations are required at each time step to compute relevant values for integration along each trajectory. Because LGR is fully Lagrangian and stores all computed quantities with the particle, these additional operations are unnecessary. It provides a much more efficient means of moving between instantaneous and finite-time measurements than other paradigms in experimental fluids.

Despite the promise of the study performed in Figures 4.4 and 4.5, the results cannot be viewed as entirely conclusive. Many untested factors—including the flow being tested, specific PIV and LPT algorithms implemented, and parameters used in PIV, LPT, or LGR—may influence the outcome. For instance, the particle tracking scheme applied is over a decade old, and will not perform as well as modern implementations like Shake-the-Box [25]. More work should be done to continue evaluating the potential of LGR for particle image experiments.

LGR Provides an Interpretable Approach to Scale Sensitivity

When developing methods to operate on practical data sets, estimating the sparsity of the trajectory data is typically a concern. Many authors approach this by counting the number of tracers used in an experiment on a benchmark flow like the Double Gyre (for example, [41], [55], [60]). Some authors will quantify sparsity according to some heuristic. For instance, Aksamit et al. [110] quantify sparsity by counting the number of tracers inside a square volume defined by a pre-selected length scale. While insightful and necessary, both of these approaches are subjective—the former to the number of tracers which are "typically used," and the latter to the selected length scale.

Codifying scale sensitivity to an objective relationship between tracer distribution and resolvable spatial and temporal scales is challenging, but the results of this chapter suggest qualitative guidelines. Generally, with a uniformly random distribution of tracers, spatial gradient fluctuations 5–10 times larger than the average radial distance to neighboring tracers will be captured (in the double gyre, this corresponds to 50–200 simulated tracers). Similarly, the results of Figure 4.9 suggest that resampling of regression tracers should occur with a frequency at least 10 times faster than the fastest time scale in the flow $\mathcal{T}$. In modern flow measurement applications, this is a reasonable constraint, as data is typically collected at a frequency much faster than the rate of evolution of the flow.

The surest guideline to evaluate whether a given regression will produce reliable results is to check the linearity of the regressed operators over short intervals. Given matrices of tracer separation at the initial time $\mathbf{X}_t$ and the deformed time $\mathbf{X}_\tau$ and the regressed deformation gradient $D\mathbf{F}_t^\tau$, the relative error in the approximated deformation is defined as

$$e = \frac{\|D\mathbf{F}_t^\tau \mathbf{X_t} - \mathbf{X}_\tau\|_F}{\|\mathbf{X}_\tau\|_F}. \tag{4.9}$$

If $e \ll 1$, then the operator $D\mathbf{F}_t^\tau$ is approximately linear and the estimated gradients are trustworthy.

LGR Improves Jacobian Approximation from Sparse Data

Section 4.3 began by applying LGR to compute flow map Jacobians and FTLE fields from sparse data. In these analyses it was found that, by frequently resampling local tracers performing regression over the intermediate intervals, LGR is able to achieve significantly improved results on sparse data over other methods used in the literature.

As discussed in Chapter 3, resampling improves Jacobian results because it ensures that the nonlinear influence of the flow remains minimal. The flow map Jacobian is the first order approximation of the deformation experienced by the tracers. For large clouds of tracers, that means that only small time intervals can serve as a linear approximation of the deformation. The analysis presented in Figure 4.9 illustrates this by showing the error incurred in FTLE calculations using varied sparsity and resampling rate relative to a trusted baseline. When the sampling rate was not sufficiently fast, error became large due to compounding nonlinearities.

Real, observable tracer data is usually relatively sparse in space but highly resolved in time. Since conventional approaches to Jacobian estimation only use the initial and final states of the tracers in their estimation, they ignore a significant quantity of information available for analysis. LGR incorporates all of the data at the intermediate time steps and therefore can achieve improved performance.

Finally, in Lagrangian analyses, a severe limitation is the persistance of tracers over the observation interval. By the conventional approach, all tracers involved in the initial cloud of particles must remain visible from t_0 to t. Therefore, when tracers disappear from view for any amount of time (for example, missed GPS readings on a drifter, or LPT tracers move out of illumination), they become entirely unusable over the whole duration of observation.

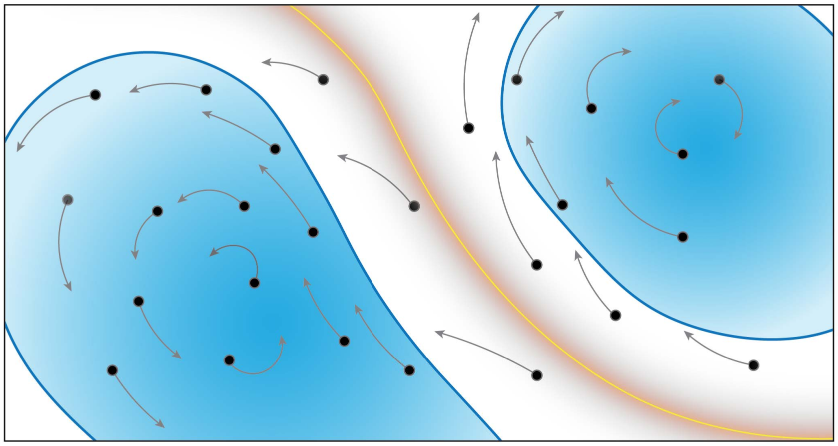


Figure 4.13: Illustration of particles interacting with elliptic and hyperbolic LCS features. The elliptic features enclose volumes of the flow whereas the hyperbolic ridge is an infinitesimally thin material surface. As a result, more tracers engage with the elliptic features and render them more detectable from sparse data.

LGR relaxes this limitation by resampling local tracers at each time step. If a tracer disappears for a moment from the flow, it cannot be used in the sets of regressions where it is missing. However it can be used in all other regressions performed on neighboring particles over the observation interval. Therefore, LGR allows for more trajectory data to be incorporated to approximations, not only temporally, but also spatially.

Elliptic LCS Metrics Appear to be More Robust to Sparsity

When comparing the LCS results from sparse trajectories to the those in previous chapters where computations were performed on dense, structured data, it is apparent that the elliptic metrics and velocity gradients are more robust to sparsity than the hyperbolic/parabolic LCS as revealed by the FTLE fields. In Figure 4.10, for example, drawing precise ridges would be a challenging task to implement algorithmically for even the relatively dense field with 2000 random particles. On the other hand, the salient features of the elliptic LCS (Figure 4.12) and velocity gradients (Figure 4.1) are clearly visible with only 100 particles in the domain. This discrepancy results from the topology of the computed structures and their sensitivity to interpolation errors; where elliptic LCS and velocity gradients are essentially measuring volumetric quantities, hyperbolic LCS identify codimension-1 manifolds (material surface) of infinitesimal thickness that are more difficult to sense.

To illustrate this further, consider the schematic in figure 4.13. Tracers and their trajectories are indicated alongside a repulsive hyperbolic LCS ridge (in orange) and two elliptic LCS (in blue) which hold tracers in Lagrangian vortices. Because the hyperbolic ridge is infinitesimally thin, it is exceedingly unlikely that any tracers exist immediately on top of the feature. Moreover, because the ridge is repulsive, the trajectories that begin near it diverge over time, allowing the nonlinear influence of the flow to further skew the results. The elliptic LCS, however, entrap particles within their boundaries for the observed duration. Since these particles exhibit similar rotational behavior and are spread over finite volume, the feature is more likely to be sensed by identification algorithms. An implication of this is that elliptic LCS are more amenable to interpolation than hyperbolic LCS. Because a larger region of tracers is affected by an elliptic LCS than a hyperbolic LCS, it is much simpler to define a field from it over sparse data. These factors account for the improved robustness of elliptic LCS over hyperbolic and parabolic ones seen throughout this work, and supports the findings of other studies on LCS robustness such as that by [124].

Kinematic Action Provides a Useful AMFM Tracking Objective

Autonomous mobile flow field measurements require that flow structures be easily and quickly identified from sparse and noisy data. Based on the results presented with Figure 4.12, kinematic action (KA) seems to offer a promising step in that direction. As has been discussed throughout this chapter, elliptic metrics indicating rotation appear to be more identifiable and robust when computed from sparse data. However, typical elliptic metrics like LAVD and DRA do not provide a simple heuristic for determining whether or not a particle is part of a vortical motion. Kinematic action, on the other hand, accomplishes this as a simple threshold, distinguishing tracer behavior as either rotational or dilatational. Tracers with KA values larger than the threshold can be reasonably considered as part of a coherent rotational movement in the flow.

However, there are still challenges associated with using KA as an objective for AMFM. The largest of these is that KA is currently defined as a forward-time metric, meaning that to know the KA values at an instant in time requires integration of behavior at unknown future times. Theoretical progress should be made towards understanding and connecting forward- and backward-time

KA and the vortices that they define. If elliptic structures could be reasonably detected using backward-time KA, it may provide a convenient tracking objective.

4.5 Related Directions for Future Work

The results and discussion provided in the chapter suggest that LGR may be a viable tool for analyzing numerical and experimental fluids, and that it may factor into future implementations of AMFM-type measurements. However, there remain many avenues of further inquiry into the technique. Here, a few prominent future directions are listed:

1. LGR should be applied to a diverse array of flows across applications. In this chapter, nearly all of the results were provided on an exceedingly simple flow—the Double Gyre—in order to maintain clarity and avoid superfluous details. However, future studies should apply LGR to more practical flows such as assimilated ocean velocities, Lagrangian drifters, 3-D LPT studies around aerodynamic bodies and more.

2. LGR should continue to be validated as a tool for computing velocity gradients and flow map Jacobians from sequences particle images. the studies performed here present a baseline justification for the use of LGR in fluids research, but much more is required before LGR can be confidently added to the arsenal of experimental fluids analyses.

3. By performing computations along a trajectory, LGR lowers the barrier to calculating Lagrangian metrics such as LAVD, TISM, and KA, which have been shown to produce useful, finite-time information. There is potential to develop additional integrated metrics based on different differential quantities. One example might be to use elements of the Schur decomposition of the velocity gradient as the differential quantities of integration. Such methods should be developed and tested.

4. With AMFM applications, it is necessary to distinguish between distinct flow features simultaneously identified in scene. Since the data provided by LGR analyses is Lagrangian, it will be necessary to develop an approach for bounding features without first converting to Eulerian fields. Graph-based algorithms could be an effective approach to consider.

5. If LGR is ever to be used for *in-situ* AMFM measurements, algorithms will need to be optimized to operate in real time. The results of this chapter demonstrate a proof-of-concept, but the computations are too slow to be implemented for most real flows. However, there are many opportunities to optimize software and hardware for real-time performance.

Chapter 5

DETECTING FLOW STRUCTURES WITH NATURAL TRACERS

The examples provided thus far have utilized nicely defined analytical flows or pre-computed velocity fields. Autonomous mobile flow field measurements (AMFM), however, must be able to operate on sub-optimal data sets. In that paradigm, it will not be possible to quickly compute velocity fields, seed and integrate trajectories, and compute gradients due to computational constraints and the lack of clean image data. Rather, modern technology must be leveraged to identify and track meaningful trajectories of tracers so that flow structures may be detected. This chapter attempts to accomplish that goal by addressing the second challenge posed in the introduction: that of interpreting fluid motion from feature-rich images.

The remainder of this chapter proceeds as follows: First, additional background is provided regarding motion analysis techniques in computer vision and their applications to dynamical systems. Following this, a modular detection and tracking pipeline is proposed as a solution to the problem of feature-rich images. Deep detection models are trained to identify debris and track their trajectories on the surface of a recorded flow. These trajectories are provided as inputs to Lagrangian gradient regression, which is employed to estimate flow gradients and identify flow structures. Once defined, the method is applied to a laboratory experiment where unconventional tracers are used to visualize flow structures. The chapter concludes with a discussion of the results and the implications of the proposed method.

5.1 Motion Estimation Techniques from Computer Vision

As has been discussed at length in previous chapters, estimating flow fields and analyzing dynamical behavior from sequences of images is commonly performed within the field of fluid mechanics, where carefully controlled laboratory experiments provide the appropriate conditions for accurate measure-

---

The contents of this chapter have been adapted from a manuscript by Harms, Brunton, and McKeon recently submitted under the title *Estimating Dynamic Flow Features in Groups of Tracked Objects*.

ments to be made. However, technological barriers have inhibited the generalization of such analyses to arbitrary systems of dynamical objects. The algorithms that are effective with carefully structured fluids experiments are not, for instance, effective when applied to videos of naturally occurring tracers or to the motion of more semantic objects such as pedestrians, vehicles, or swarming creatures. Now, however, it is no longer difficult to identify arbitrary or semantic objects in images. Technologies such as deep detection and tracking enable broader classes of dynamical systems to be studied in the same manner as fluidic ones.

In the vision community, estimation of dynamical behavior is often pursued through one of two primary approaches. The first of these is optical flow (OF), which identifies the apparent motion from one image to the next in a sequence. As was briefly discussed in the introduction, PIV may be considered a sub-category of OF tailored for specific application to fluids. A variety of surveys have been published to categorize the optical flow algorithms that have proliferated in recent years (for instance, [125], [126]). Traditional approaches such as Lucas-Kanade [127] and Horn-Schunk [28] are knowledge-driven in the sense that they do not need to be trained on data to be implemented. In recent years, deep, or data-driven, OF models such as FlowNet [128], [129], PWC-net [130], and RAFT [131] have seen significant improvement in accuracy and computation time over their knowledge-driven counterparts [126]. Scene flow—the extension of optical flow to three dimensions using either disparity (depth) estimation in frame or 3D point clouds obtained via LiDAR—has also been heavily researched in recent years due to breadth of applications [126]. Though OF methods have become adept at estimating motion in scene, they are not well equipped to estimate the dynamic behavior of specific groups. OF does not discriminate objects in a scene, so any analysis of the motion is done on all motion within the scene, regardless of its relation to system dynamics. Moreover, OF analyses are not intrinsically objective and are limited to only finite-time measurements.

Rather than broadly estimating the motion of an entire scene, multiple object tracking (MOT), examines the movement of particular objects over time. In a sense, it is the generalization of LPT. MOT can be used to estimate the velocity of specific subjects along their trajectories rather than the velocity field over the entire scene. Many reviews have been written on the subject (for instance, [132], [133]). In its most basic form, MOT consists of an object detection step and a target association step. Object detection may be accomplished through the use of large vision models [134], and association achieved by state estimation techniques like Kalman filters, particle filters, the Hungarian algorithm, or one of many other approaches [133]. Recent years have seen tracking frameworks built around vision transformers [135], [136], graph models [137], [138], and attention [139], [140], to name a few alternative approaches.

Traditional MOT methods are also limited as a tool for identifying dynamical patterns in crowd behavior. MOT analyses are designed to give information about the behavior of individual entities, rather than interpret the underlying systems driving their motion. A field of work related to this is crowd analysis [141], [142], which seeks to understand people's behavior in crowds. MOT is sometimes used for such studies [143], which has seen recent interest as a result of social distancing protocols during the COVID-19 pandemic [144], [145]. Crowd analyses, however, are typically not directed towards understanding the dynamic flow patterns espoused by the studied objects, but rather towards identifying anomalies or perceiving the intent of the crowd [142]. Understanding flow patterns in crowds is still largely unaddressed.

While PIV and LPT provide robust and highly capable methods for analyzing flow fields in highly controlled experiments, there remains a need for measurement technologies that leverage naturally occurring and imperfect (in the sense that they may be sampled from a broad distribution or that they do not ideally follow the true flow) tracers and for methods which can be successful using non-specialist imaging equipment. The current state-of-the-art methods in flow measurement require expensive instrumentation and the addition of artificial tracers in order to be viable.

The methods in this chapter leverage modern capabilities of computer vision in tandem with LGR to provide a framework for extracting dynamical information beyond velocity from groups of tracked objects. Specifically, large vision models are used to identify objects of interest and record their motion as a series of trajectories. These are provided as inputs to the Lagrangian gradient regression (LGR) algorithm which enables reliable gradient estimation from sparse trajectories such as are characteristic of MOT studies. Using the flow gradients, quantitative information of the flow including rotation and deformation rates for instantaneous and finite times can be computed. While such gradient-based analyses have been useful in fluidic applications [39], [41], this work aims to extend their use to real-world scenarios where dynamical ob-

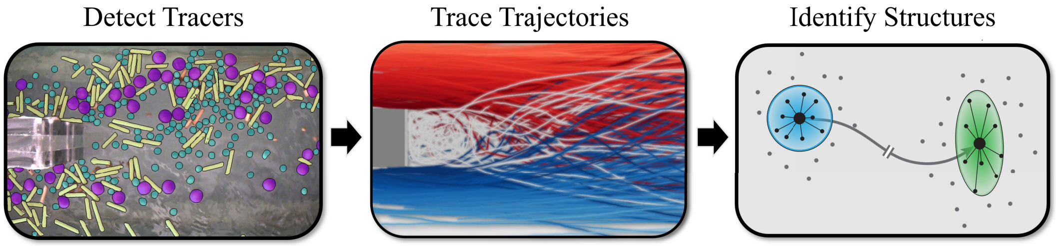


Figure 5.1: Illustration of the modular flow structure identification process developed in this chapter. The first step detects tracers through deep vision models, the second maps out their trajectories through algorithms from computer vision or LPT, and the final step estimates gradient-based flow structures using LGR.

ject classes and the scenes they inhabit can be visually complex and difficult to analyze by standard approaches.

5.2 Identifying Flow Structures with Natural Tracers

Here, a 3-step modular approach (illustrated in Figure 5.1) is proposed for combining MOT with dynamical systems analyses. The first stage of analysis is object detection, the second is object tracking given the detections, and the final stage uses LGR to estimate flow gradients from tracked trajectories and identify underlying features. The stages of the analysis are defined in a modular way to allow for various implementations of detection and tracking to be easily interchanged. This enables the rapidly evolving detection and tracking technologies to seamlessly integrate into the flow structure identification process. Moreover, it may be that some models perform better on certain data types than others, so it is desirable to be able to quickly switch one for another. More information on each stage is here provided.

Tracer Detection

Acquiring trajectories for flow structure computation is achieved through modern object detection and tracking techniques. Using image sequences, trajectories may be observed in either $d = 2$ or $d = 3$ dimensions. In the first scenario, a single camera is assumed to record motion on a plane that is approximately parallel to the sensor plane or where a homography can reasonably orient the camera as perpendicular to the plane of motion (See Appendix B, Section B.2). For 3-dimensional motion, multiple cameras are required and an additional triangulation step is necessary between detection and tracking. In this chapter, only planar flows observed with a single camera are considered.

Deep vision models provide the framework for robust and general object detection. Many models, including RCNN architectures [146], [147], YOLO architectures [148], [149], transformer architectures [150], and more are sufficient. It is necessary to represent each detection as a single point in the flow space. In this work, the centroid of an identified mask is used as the tracer location, although other approaches can be considered based on the problem parameters. Large vision models such as those previously mentioned also allow for flexible application. If unconventional objects must be detected, then transfer learning techniques [151] can be applied to fine-tune the network while preserving the features represented in the pre-trained weights. Many cases may also involve dense crowds of objects or relatively small objects. In such instances, it may be helpful to incorporate a sliding window approach to detections, which will be discussed in more depth with the case study in section 5.3.

Tracer Tracking

Using whatever detection model is preferred, object tracking algorithms are used to stitch independent detections into trajectories. Many such algorithms exist in the computer vision literature, including filters and deep implementations [152], [153] as well as in the fluids literature from the Lagrangian particle tracking community [19], [25]. Whatever tracking algorithm is used, the quality of the trajectories in frame should be emphasized. Trajectory lengths should be made as long as possible while in the frame. As will be seen later, truncated trajectories are the most significant failure point of the LGR algorithm. However, while many computer vision applications are concerned with re-identification of objects that leave the frame, that is not an issue here. Any objects that exit and re-enter the scene can be viewed as entirely new entities without loss of performance.

Structure Identification

The quality of LGR analysis depends the characteristics of the observer and the properties of the flow. In planar flows, optimal measurements will be made when the camera sensor is aligned parallel to the flow and lens distortion is minimized. As has been discussed in previous chapters, LGR will identify

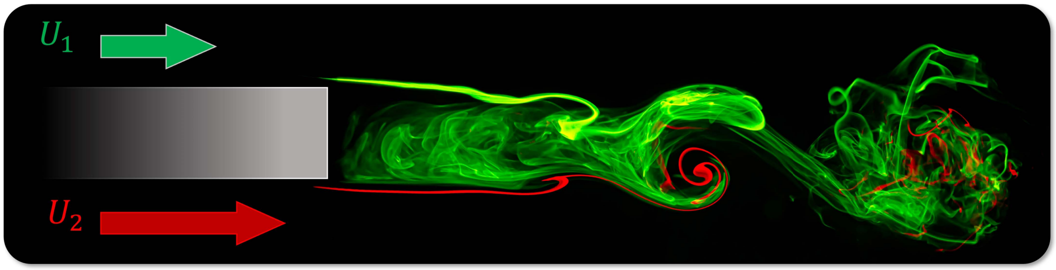


Figure 5.2: Dye Flow Visualization of the asymetric wake flow past a blunt trailing edge. Immediately beyond the trailing edge, the flow is characterized by bluff body vortex shedding. Further downstream, the flow transitions to a shear layer. The flow is visualized using fluorescent dye added just upstream of the trailing edge of the plate.

a greater range of spatial flow scales if the spacing between them is relatively small. Small and crowded objects, however, are known to be more difficult to detect. Therefore, there is a tradeoff between flow structure resolution (which improves with higher density and smaller objects) and detection quality (which improves with larger and distinct objects).

5.3 The Asymetric Wake Past a Blunt Trailing Edge Experiment

In this section, an experiment is performed to demonstrate features of the proposed method and its ability to perform with complex images. The emphasis of the designed experiment is to simulate debris on the surface of a flow using unconventional tracers and use their motion to identify structures. Once again, the asymmetric wake flow is used as the testbed of methods being developed. Fluid flows around the blunt trailing edge of a long plate such that the flow velocity differs across the plate. The result is characterized by bluff-body vortex shedding immediately downstream of the plate, which transitions to a shear layer further downstream. To better conceptualize the flow, dye visualization taken during the experiment is provided in Figure 5.2. Because it is used multiple time throughout this dissertation, extensive details and schematics of the experiment are provided in Appendix B Section B.4. A brief description is provided here.

Experimental Setup

The experiment was performed in the NOAH water channel at the Graduate Aerospace Laboratories at California Institute of Technology (GALCIT). A 0.057 meter thick plate was placed into the channel so that the trailing edge was situated near the streamwise midpoint of the $0.45 \times 0.45 \times 1.22$ meter test section. A velocity ratio of $U_2/U_1 < 1$ was enforced by adding various blockage implements upstream.

This experiment was performed with generally accessible optics. A machine vision camera was implemented in this case, but the study could have easily been conducted with cell phone cameras. Specifically, video recording was performed using a FLIR Blackfly S BFS-U3-23S3C-C USB3 color camera with an Edmund Optics 4mm C-mount lens. The optical setup was mounted directly above the water surface just downstream of the plate trailing edge and 1920×1200 pixel images were recorded at 60Hz during the tests. Camera calibration was accomplished using the ChArUco approach described in Appendix B Section B.2.

Flow motion was visualized using atypical tracers which were added on either side of the splitting plate upstream of the trailing edge. Three categories of tracers were considered: 1. 9.53mm diameter birch spheres, 2. 19.05mm diameter birch spheres, and 3. 6.35mm diameter birch rods cut roughly between 12.7 and 50.8mm in length. The following results will discuss two test cases: In the first, only small birch spheres are used as flow indicators. The second uses all of the particle types simultaneously.

5.4 Gradient Estimation via Existing Methods

To emphasize the need for a new approach, flow gradients were estimated on the experimental images using conventional velocimetry approaches from fluid mechanics and computer vision. The two algorithms considered are multipass particle image velocimetry (PIV) [18] computed using the open-source package OpenPIV [121], and RAFT for optical flow [131] using pre-trained weights available at the associated software repository. The results are displayed in Figure 5.3, where out-of-plane vorticity $\omega_z = \partial v/\partial x - \partial u/\partial y$ is displayed as an indicator of the gradient.

This analysis shows that conventional motion estimation techniques are not well suited to approximate velocity gradients in feature-rich images. For both tested techniques, false gradients are apparent at the top and bottom edges of the debris cluster (a red line near the top of the image and a blue line near the bottom). These appear since PIV and optical flow are agnostic to its semantic information. They attempt to identify motion in all parts of the image based on correlation or pixel intensity conservation and thus perceive a

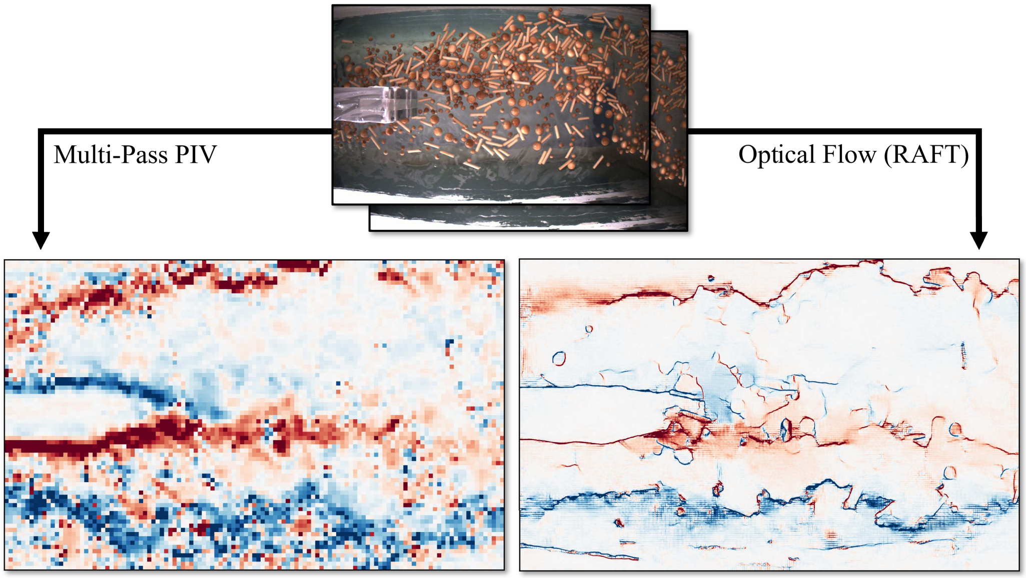


Figure 5.3: Computing flow gradients from an image sequence using standard procedures from experimental fluid mechanics (bottom left) and from computer vision (bottom right). The algorithm used on the left is multi-pass PIV [18], and the algorithm used on the right is RAFT [131]. The results display vorticity ($\partial v/\partial x - \partial u/\partial y$). From them it is clear that existing methods are not suited for spatial flow gradient estimation. In both cases, there is significant noise and the existence of many spurious features.

strong false gradient between the cluster of debris (where motion is evident) to the surrounding flow without tracers (where motion exists, but is invisible to the camera). There is also a significant amount of noise present in both analyses, which stems from the complexity of the images being analyzed and the presence of non-flow features such as reflections. Furthermore, due to the poor quality of estimated velocity gradients, computing finite-time metrics like FTLE or LAVD using PIV or optical flow is challenging and would likely perform poorly.

5.5 LGR Processing Appraoch

The methods discussed in the previous sections allow for flow gradients to be reliably estimated. In this case study, the Mask-RCNN architecture [147] is employed for object detection and an in-house developed template matching algorithm is used for object tracking. More details on the implementation of detection and tracking are provided below.

Stage 1: Detection The Mask-RCNN model architecture was selected for the case study because it allows for full masks, and therefore accurate centroids, to be computed. The model was implemented using PyTorch, where a pre-trained ResNet50 [154] backbone was incorporated. While three classes of particles existed in the flow, a custom head was built to identify all three as a single debris class. Identifying a single class, however, is not necessary and may not be desirable in some cases. The generality of large detection models enables studies to be performed on multiple classes of tracers at once.

Training data was collected using a custom app built around the Segment Anything Model (SAM) [155], which allowed for many precise masks to be identified quickly in training images. In total, 165 training images were used with > 3000 identified masks. The custom model was trained for 100 epochs using a 120-to-45 training-to-testing split. Stochastic gradient descent was used as the optimizer with initial learning rate of 0.0005, momentum of 0.9, and L_2 regularization of 0.0005. A scheduler reduced the learning rate by a factor of 10 every 10 epochs. For more details on the training data collection procedure, see Appendix B Section B.3.

Because the tracer particles were localized in crowds and were relatively small compared to the frame size, a windowing scheme was used to achieve improved detection results in each snapshot. For the results discussed below, a 400×400 pixel window with 10% overlap was used for detections. Any overlapping masks were consolidated into single particle masks, which were then used to identify the tracer centroids. Figure 5.4 shows an example of the windowing procedure applied to a sample image from the debris flow.

Stage 2: Tracking Object tracking was executed using a straightforward template matching scheme involving a forward search for the tracers identified in the first image of each image pair in the sequence. A window around the identified tracer was specified as a template whose greatest correlation was found within a larger window in the subsequent frame. If the peak of the correlation was found to be near a detection in the second frame, the detection was appended to the trajectory. Velocity and acceleration constraints were employed to ensure that non-physical trajectories could not be created. Finally, to improve trajectory statistics, identified trajectories were stitched together by fitting a polynomial to an existing trajectory and searching future time steps for possible matching trajectories. Once trajectories were determined, the camera calibration and fitted homography were applied to the trajectory data. The details of the tracking algorithm and its implementation can be found in Appendix C.

Stage 3: Structure Identification Gradient estimation and structure identification was performed using LGR according to the procedure outlined Chapter 3. Equation 3.11 was employed between each time step using the $k = 15$ nearest neighbors of each tracer to populate the $\mathbf{X}_{t_0}$ and $\mathbf{X}_t$ matrices. The kernel matrix $\mathbf{K}$ was populated on the diagonal by

$$\mathbf{K}_{ii} = e^{-\frac{\Delta \mathbf{x}_i(t_0)^2}{2s^2}}, \tag{5.1}$$

with $s = 0.03$ meters and set to zero otherwise. A small regularization constant $\gamma = 10^{-6}$ was applied to ensure numerical stability.

Visualization Approach

To visualize the data, each tracer centroid was provided as a query point to SAM [155], which identified its semantic mask. Each mask was overlaid on the original image with partial transparency and colored according to the value of the chosen metric. For example, in Figure 5.7.

5.6 Detection and Tracking Results

Results from the asymmetric wake demonstration of the proposed method are presented in two stages. First, results from detection and tracking are addressed independently. The algorithmic performance in these tasks is critical to the success of gradient estimation. Therefore, this section is devoted to guaranteeing that the detection and tracking algorithms perform satisfactorily. Section 5.7 examines the results of the gradient-based metrics computed on the flow using LGR.

Detection Results

Once the detection model had been trained, it was applied to each image in a recorded video. It was necessary to separate each frame into overlapping tiles upon which the detector was applied independently to achieve satisfactory results. The need for this is highlighted in Figure 5.4, where 5.4a displays

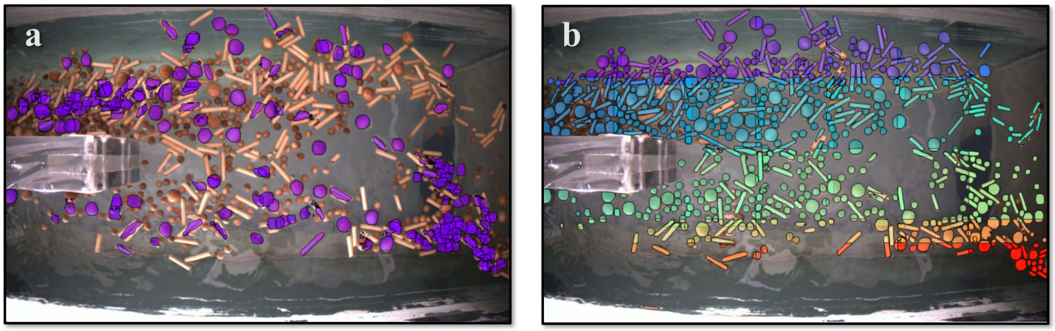


Figure 5.4: Tracer detections with and without windowing. (a) All detected debris when the detection model is applied directly to the entire image (1200 × 1920 pixels). (b) Detected debris when the detection model is applied to smaller, overlapping windows (300 × 300 pixels with 10% overlap) tessellated over the original image.

all masks identified in the image when the detector is applied to the entire 1200 × 1920 pixel frame at once and 5.4b displays all masks identified when the image is first tiled into 300 × 300 pixel windows with 10% overlap. Clearly, windowing is necessary given this particular model.

Poor detector performance is often expected when seeking to detect objects that are small relative to the size of the frame and are in crowds. This may be due to the functionality of detector models that are readily available for fine-tuning. The Mask-RCNN model [147] is a convolutional neural network (CNN) which must infer from uniformly sized images. Thus, when an oversized image is provided to the network, it must first be downsampled and scaled to match the acceptable dimensions of the network. An undersized image will be upsampled and scaled appropriately. Upsampling, however, does not eliminate features from the image, whereas downsampling will. Therefore, when a full-sized image of tracers is provided to the detector, the fine features of the small and crowded tracers may be lost to image pre-processing. Other backbone architectures such as visual transformers (ViT) [150] may provide an avenue to overcoming this obstacle. The state-of-the-art in detection is a rapidly advancing frontier, where breakthroughs frequently occur. For this reason, the proposed methods were designed to be modular.

When windowing was applied, the detection architecture was able to consistently identify a majority of the particles in each frame. Evidently, the spheres were better detected than the rods. This may be due to a variety of factors, including simpler geometry, rotational symmetry, and proportional representation in training data. Additionally, the rods demonstrated a tendency to align lengthwise into groups, which were commonly undetected or misidentified.

Tracking Results

The ability to detect flow structures from gradients largely hinges upon the quality of identified trajectories in a flow. While LGR significantly relaxes trajectory requirements by resampling tracers at each time step, the flow map Jacobian $D\mathbf{F}_{t_0}^{t}$ along a given trajectory can only be computed if the analyzed tracer is observable at all time steps in the interval $[t_0,\, t]$. (Note that the relaxation afforded by LGR is a result of the non-necessity to observe *neighboring* tracers over the entire observation interval. These tracers may only exist for a single time step, but can still be used in the incremental gradient calculations that feed into $D\mathbf{F}_{t_0}^{t}$ via equation 3.16.). Thus, to the degree that trajectories are accurately and fully mapped, flow structures can be reliably identified.

A significant hindrance to complete trajectory mapping is the occurrence of missed detections in the previous stage. Even a sufficiently trained detector applied to tessellated windows across the images occasionally misses detections, which splits a given trajectory into two pieces reducing the ability to perform finite-time analyses. These missed detections, however, often only occur once or twice in a row at random along a trajectory. Therefore, to overcome this challenge, the polynomial-based trajectory merging algorithm detailed in Appendix C was designed to restore trajectories broken by random and sparse missed detections.

The improvement to trajectory statistics as a result of merging is presented in Figure 5.5, where empirical complementary cumulative density functions (ECCDF) are presented for merged and unmerged processing of the all debris and the small spheres videos. Large markers indicate the median trajectory length and dashed vertical lines indicate the mean trajectory length.

A few observations can be made from the data. First, all of the cases exhibit near-subexponential distributions of trajectory length—they are all essentially heavy-tailed. As a result, instantaneous metrics, which only require trajectory length of 2, will have many more sample points at each snapshot in time than their finite-time counterparts. As the length of integration extends, the number of available trajectories decreases.

As a second observation, the small spheres data seems to slightly outper-

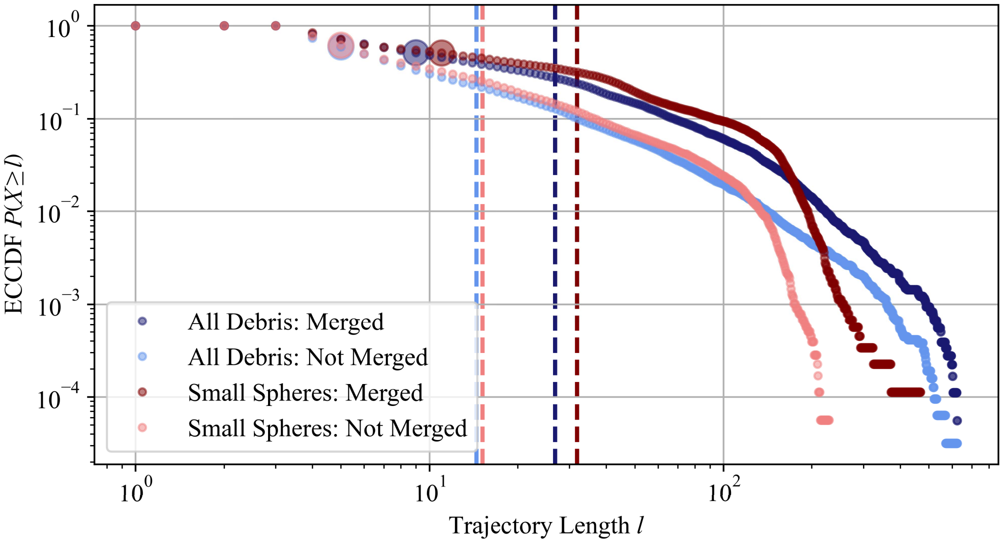


Figure 5.5: Trajectory length statistics for the split plate LGR experiment. Empirical complementary cumulative density functions are provided for the trajectory merged and unmerged processing of both the small spheres data and the all debris data. Large circles indicate the median trajectory length of each data set, while dashed vertical lines indicate the mean trajectory length. From the data, it is observed that trajectory merging successfully improves trajectory quality prior to LGR implementation.

form the all debris data given the same processing parameters. This may be due to a number of factors, including the added complexity of additional types of debris, the ability of the detector to generalize from spheres to rods, and clustering and crowding when more debris is present. However, the differences are relatively small, and do not significantly impact the results on LGR.

Finally, the trajectory merging algorithm is observed to yield a dramatic improvement to tracking capability. This is most readily perceived by the median and mean trajectory lengths, which both increase significantly after application of merging. This allows for many more viable samples of finite-time metrics at each snapshot, thereby improving the interpretability of the outcome. Nevertheless, it is noted that, even with trajectory merging, the tracking implementation is relatively naive, and can be improved by applying more advanced algorithms. Strategies for such improvements are discussed in Appendix C.

Though the presented approach to object tracking is simple, it was able to construct a sufficient number of tracks for trustworthy gradient estimation.

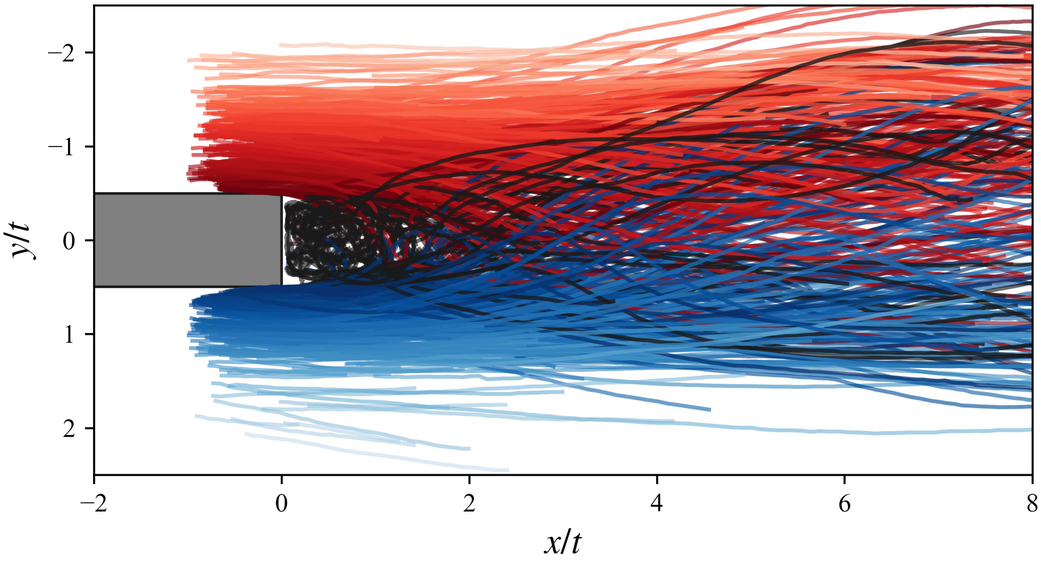


Figure 5.6: Sample trajectories from the split flow experimental case study. A subset of trajectories length 50 or longer from the small birch spheres dataset. All trajectories start in $x/t \in [-1, 1.25]$. Color is determined by y/t. If the first instance in a trajectory is in $y/t \in [-0.5, 0.5]$ it is colored black. In this display, t represents the thickness of the splitting plate.

Figure 5.6 displays a selection of the recorded tracks from the small spheres data with 50 or more associated detections in world coordinates normalized by the thickness of the plate t. Trajectories are colored based on their starting location in the flow. Trajectories whose first detection is above the splitting plate are colored in red, while those that are first detected beneath are colored in blue. Those trajectories which began behind the plate are colored in black. By observing the color of the trajectories based on their initial position, the mixing effect of the shear layer begins to become evident.

5.7 Results for Gradient-Based Metrics

Having thus demonstrated the quality of estimated detections and trajectories, LGR is now applied on the experimental data. Standard metrics based on the velocity gradient tensor and the flow map Jacobian are displayed in Figure 5.7 and discussed.

Flow Feature Identification

The performance of LGR is first demonstrated through the computation of vorticity. Figure 5.7a presents the vorticity for data containing only small birch spheres as tracers. Here, the vortex shedding that occurs at the trailing

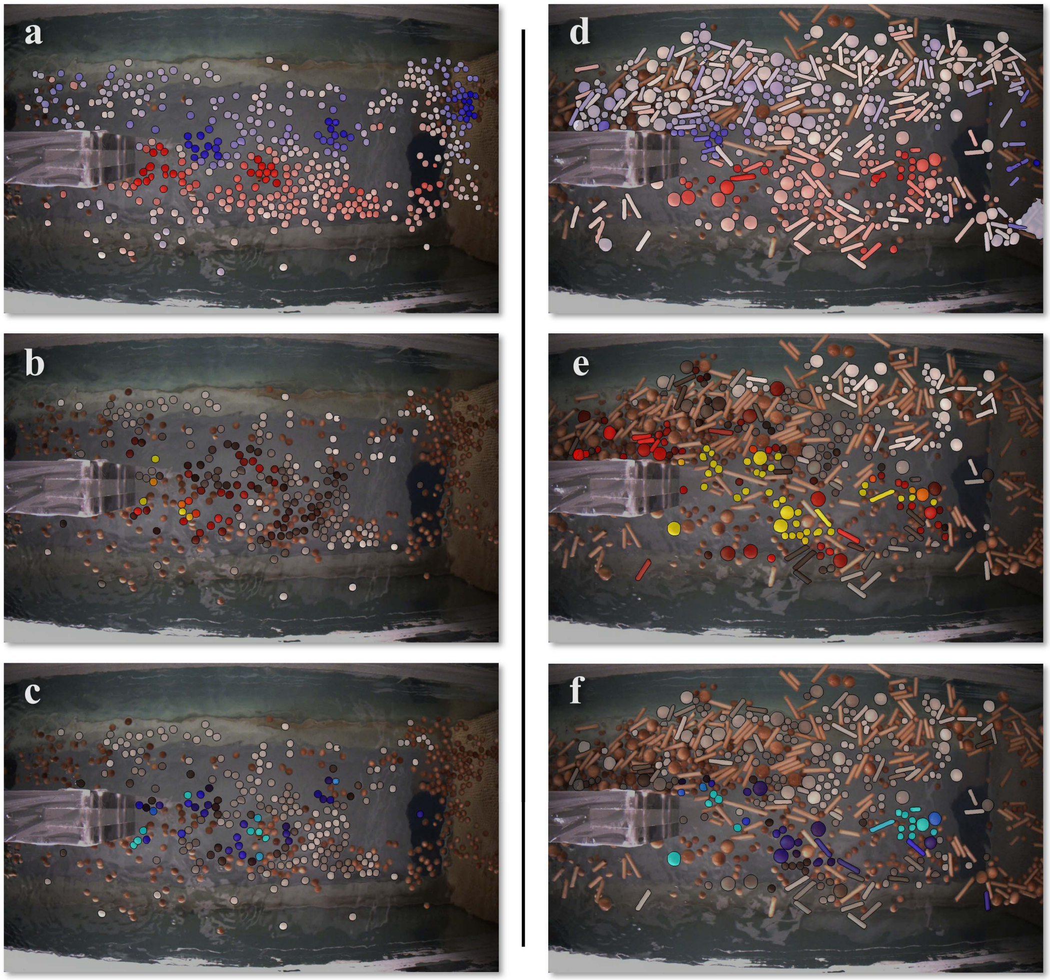


Figure 5.7: Results of the proposed analysis on the debris flow data. Top: Small birch spheres as tracers. Bottom: All birch debris as tracers. Vorticity is computed in (a) and (d), where red indicates positive value (counterclockwise rotation) and blue indicates negative value. Finite-time stretching (FTLE) is presented in (b) and (e), where yellow indicates large value and white is zero, and finite-time rotation (LAVD) is given in (c) and (f), where bright blue indicates large values.

edge of the plate is clearly visible in the computed vorticity values. In the given frame, there are three clockwise vortices (blue clusters) which alternate with counterclockwise vortices (red clusters). The vorticity in the data set using all debris is presented in Figure 5.7d, where the inclusion of large spheres and rods increases the density of tracers on the water surface and thereby impedes some tracer motion through particle interactions. This is clearly seen in the vorticity data, where there are fewer vortices (clusters of either red or blue) visible.

Finite time metrics can also be used to identify patterns in the flow of detected tracers. Measurements of maximal stretching (FTLE) are presented in Figures 5.7b and 5.7e for the spherical tracers and for all tracer classes, respectively. A large value of FTLE (red to yellow masks on the image) indicates that surrounding tracers will separate over the duration for which the computation is performed. The theory of LCS suggests that ridges in the FTLE field are codimension-1 material surfaces that act as barriers of flow transport [48]. Because they are infinitesimally thin, FTLE ridges are difficult to identify from sparse data such as what is represented in Figure 5.7. Additionally, inertial tracers, such as those used in the experiment, will diverge in vortices as a result of centrifugal force due to rotation. This effect is evident in frames e and f of Figure 5.7, where tracers are seen to have large FTLE and large LAVD simultaneously. Despite these confounding effects, identifying regions of trajectory divergence remains useful even when sharp ridges may be difficult to discern.

LGR also enables measurements of cumulative rotation through computation of LAVD. The LAVD results for the case study are displayed in Figures 5.7c and 5.7f for the spherical tracers and for all tracer classes, respectively. If a tracer exhibits a large value of LAVD (bright blue masks), then surrounding tracers will rotate around the examined trajectory. Unlike FTLE fields, LAVD reveals volumetric regions of the material that experience significant rotation. Therefore, clusters of tracers with large LAVD values can be instructive for identifying coherent structures in the dynamical system. Such clusters are clearly visible in both Figures 5.7c and 5.7f.

5.8 Discussion of Results

In this chapter, a modular procedure has been developed to enable kinematic measurements in flows that are characterized by the motion of naturally existing observable tracers. The results for the proposed method have been presented on a laboratory flow emulating a debris flows which may be observed elsewhere. Here the method and the results are discussed with implications. The method shows promise for expanding flow measurement capabilities, but also exhibits limitations that encourage further development.

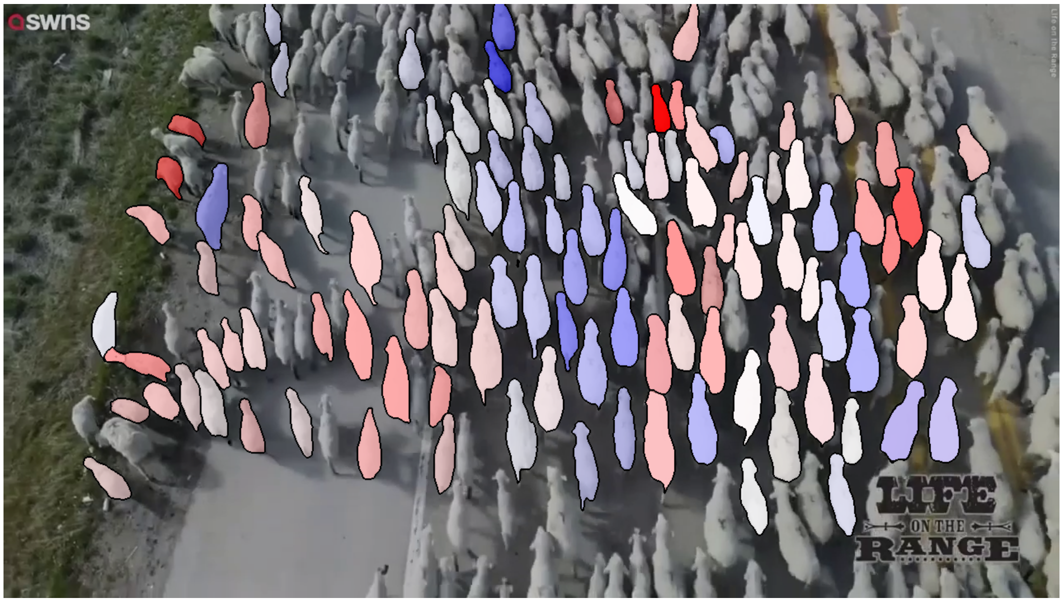


Figure 5.8: Application of the methods proposed by this chapter to aerial footage of herding sheep. Sheep are colored by their "vorticity." The displayed image is taken from a YouTube video (https://www.youtube.com/watch?v=MSjbOlQVN58) originally produced by the Idaho Rangeland Resources Commission and posted by SWNS [156]. Reproduced with permission.

A Semantic Approach to Measurement of Flow Kinematics

Typical approaches to flow measurement are *syntactic*, in the sense that they focus on the structure and form of the data rather than its meaning. Optical flow and PIV are both examples of syntactic analyses of motion, which quantify observed movement based on the structural change of pixel intensities between frames. The effect that this has on natural data was evident in Figure 5.3, where large spurious gradients were observed at the interface of the cloud of tracers where tracer motion was visible but flow motion was not.

The approach developed in this section is, to the author's knowledge, the first *semantic* approach to measuring flow kinematics. It measures the flow based on the meaning and context of the underlying images rather than on their structure. By selectively identifying the immersed observable tracers, the motion of the flow is factored into the analysis without including other details from the scene. This was shown to effectively detect flow features in the analysis centered on Figure 5.7 by a variety of metrics.

That the proposed method is semantic implies that it is extensible to use cases not often considered for kinematic analysis. Two examples come to mind: First, it could be directly applied to assessing the kinematic behavior of groups of tracers of other categories than passive flow tracers. For instance, the method could be applied to the motion of pedestrians, traffic, or herds and swarms if the correct training data is supplied. In fact, out of interest, these approaches have already been applied to aerial footage of sheep being herded; an example image displaying vorticity (if such a thing exists in sheep flows) is presented in Figure 5.8.

The second example of a use case enabled by the proposed method is that of analysis of multiple object categories simultaneously. For example, while not pursued in this work, it would have been possible to consider the motion of rods separately from the motion of spheres in the simulated debris flow. Other useful examples may include the simultaneous analysis of bubbles, sediment, and neutral tracers simultaneously in a flow.

Reliable Gradients from Difficult Data

The methods introduced in this work were designed with the intent of enabling the computation of flow gradients and related metrics in dynamical systems which are typically intractable to gradient-based analyses. As illustrated in Figure 5.3, the experimental case study of section 5.3 represents data where flow gradient estimation is extremely difficult when applying traditional analyses. By implementing the methods developed in this chapter, reliable gradients and associated metrics are directly computed with little additional processing. The principal requirement for success is the availability of training data on the classes of tracers to be tracked.

Incorporating Single-Particle Kinematics

Though it is not explored in examples contained in this dissertation, the semantic tracking approach to flow kinematics may naturally be extended to consider the rotation and deformation (if a deformable body) of tracked tracers. One application of this might be to incorporate the observed rotation of a tracer into estimates of underlying flow gradients. Because the mask of the object is computed through the detection step, all of the information required to calculate particle rotation is readily available. Alternatively, one could identify features on each object and track their motion between frames. Similar approaches could be employed to measure deformation of an identified body.

Modular Framework for Continuous Improvement

The framework developed in this work has been modularly constructed to accommodate the rapid advancement of object detection and tracking capabilities. The detection and tracking schemes used in the examples above are relatively naive given the current state of the field, and therefore serve as an apt demonstration of the framework as a whole; if gradients can be successfully computed using simple detection and tracking schemes, then more advanced methods will be able to improve upon the results. As computational analyses become increasingly general, they will serve to enhance this approach to kinematic analysis rather than supersede it.

Lenient Hardware Requirements

The methods introduced in this work can be easily implemented using affordable, readily available hardware. All of the data presented in the experiments were collected using measurement systems costing less than 1000 USD. The cost of laboratory experiments, by contrast, can easily exceed orders of magnitude greater than this. Moreover, due to the availability of large pretrained computer vision models, detector training costs are modest and do not require more computational power than a desktop computer. Therefore, the methods developed here are relatively democratic in the sense that they allow hobbyists and other enthusiasts to begin making measurements without access to specialized equipment.

Present Limitations

Missed detections resulting in truncated trajectories represent the primary barrier to estimated gradient accuracy. This is especially true when computing finite-time metrics like the FTLE and LAVD, which require the existence of trajectories over extended intervals. With the simulated debris flow, while there were missed detections for both the small spheres and all debris cases, they were more frequent when using all of the debris. Rods, and especially groups of aligned rods, were less likely to be detected than the spheres. This suggests that improvements can be made by training on more data. Additionally, as the algorithm is currently implemented, the detections are independent in each frame. However, significant improvement could be made to the LGR outcome if tracking techniques that incorporate detections from previous frames such as SORT [152] or deepSORT [153] are implemented for trajectory construction. Thus, it seems that there are many algorithmic modifications which can quickly improve performance.

Moreover, while real-time implementation of the LGR pipeline is a long-term goal of this work, it is not yet achievable. With the current implementation and modest hardware, processing one of the debris flow videos requires on the order of hours of computation. The bottleneck in computation time is the detection step, which uses a large model to fully mask tracers and store the mask information. However, for the gradient analyses, the masks are not needed—only the centroids. Other models might be able to efficiently identify tracer centroids without all of the overhead incurred by large detector models like Mask-RCNN.

Aside from changing models, computation time could be greatly reduced if one were to optimize the code for better performance on a GPU. This work represents a proof-of-concept and a demonstration of a new technology, and is yet far from practical, consumer implementation. If engineering effort were devoted to the task, it seems likely that many more cases could be analyzed online in real-time.

5.9 Related Directions for Future Work

The semantic kinematic measurement approach developed in this chapter offers many avenues of potential development and expanded research.

1. The proposed method should be applied to a variety of different flows, especially those outside of the laboratory. This extension is explored later in the thesis with Chapter 7, but should be pursued in other contexts as well. For instance, the method could be extended 3D and the analysis of deformable bodies like jellyfish. Such studies would progress the state of the method and would benefit the research community.

2. This dissertation is satisfied to implement only the Mask-RCNN for detection and template-matching for tracking. However, other algorithms should be tested in their place and considered for their benefits. As real-time analysis is desired, for instance, it may be useful to test lighter-weight centroid detection algorithms and apply target association algorithms that do not require image content. Such advances would be a step closer to AMFM systems.

3. In the same vein as the previous point, the existing algorithms should

be optimized to perform quickly and to perform on GPUs. The current implementation is designed for research purposes, and is not yet suitable for deployment.

4. Finally, from a fluid mechanics perspective, analysis of surface tracer motion relative to sub-surface motion should be examined. This is the subject of Chapter 6, which performs a preliminary investigation of the matter.

Chapter 6

SYNCHRONOUS EXAMINATION OF SURFACE TRACER AND SUB-SURFACE FLOW DYNAMICS

Common approaches to experimental flow field measurement require that very specific flow tracers be added to the studied fluid and that they are carefully illuminated in the region of interest. In the last chapter, however, a measurement approach was proposed which could estimate kinematic behavior from arbitrary classes of tracers in arbitrary scenes. A principal advantage of this approach is that it can be used to perform experiments in locations and circumstances where there is limited control over the flow. Rather than seeding and illuminating tracers, those that are naturally available can be used for the measurements.

However, measuring a flow by the motion of arbitrary tracers raises questions as to the authenticity of the approximated kinematic features that they elicit. The tracers used for common flow field measurement techniques such as particle image velocimetry (PIV) are assumed to be neutrally buoyant and essentially massless. In other words, they are assumed to perfectly follow the flow without deviation from the pathlines of fluid elements. These assumptions, however, do not hold in general. Surface tracers have mass, non-trivial geometry, buoyancy, roughness, and other features which cause their trajectories to deviate from the trajectories of fluid parcels.

This chapter is written in response to the problem posed by imperfect tracers. Can kinematic measurements approximated from the motion of arbitrary tracers—and particularly observable surface tracers—be considered trustworthy?

To address this question, an experiment was performed which pits traditional flow field measurements against the methods developed in Chapter 5. PIV is recorded simultaneously with semantic tracer tracking, and the results are compared directly. A preliminary analysis is performed on the data, and the results are considered and discussed. It is found that, while surface tracers do not perfectly represent the motion of sub-surface flow (to a degree dependent on tracer properties), their motion can still generally be used to identify features like vortices with reasonable accuracy.

The remainder of this chapter is structured as follows. First, a brief discussion of relevant background work is provided in Section 6.1. This is followed by details regarding the performed experiment in Section 6.2 and processing parameters in Section 6.3. Various aspects of results are discussed in Sections 6.4, 6.5, and 6.6. Finally, discussion and implications are provided in Section 6.7 and avenues for future work are provided in Section 6.8.

6.1 Additional Background

There is a great breadth of literature examining the effectiveness of imperfect tracers for measuring fluid flows. It is not possible, in the scope of this chapter, to adequately probe its depths. Nevertheless, this section provides a cursory discussion of some important topics related to imperfect tracer dynamics in complex flows.

Due to the prevalence of optical flow field measurements involving tracers such as PIV and LPT, much research has been done to determine measurement accuracy from very small immersed tracers. Typically, the Stokes number is used to quantify a particle's response to the flow. The Stokes number represents the ratio of the characteristic response time of a tracer τ_p to the characteristic timescale of the flow τ_f [18], [21]

$$Stk = \frac{\tau_p}{\tau_f}. \tag{6.1}$$

In PIV experiments, the force of drag on the tracers is typically approximated by Stokes' drag law

$$F_d = 6\pi\mu R v, \tag{6.2}$$

where F_d is the force of drag on a spherical body, μ is dynamic viscosity, R is the sphere radius, and v is the flow velocity relative to the object. For Stokes' drag law to be valid, it must be assumed that the tracers are smooth, spherical, of homogeneous material, surrounded by locally laminar flow on the scale of the particle, with no inertial effects or interactions with other particles. In other words, the tracers must be sufficiently small that they mimic the accelerations in the flow. If these assumptions hold, then

$$\tau_p = d_p^2 \frac{\rho_p}{18\mu}, \tag{6.3}$$

where d_p is the tracer diameter and ρ_p is the the density of the tracer [18]. The characteristic time scale of the flow τ_f is dependent on the specific flow. It is typically represented as the ratio between a length scale and a characteristic velocity. It is often recommended that tracers should have Stokes number $Stk < 10^{-}1$ for them to be considered accurate flow tracers. Practically, this means that PIV tracers seeded in water are often in the range of 10-100 microns.

While appropriate for PIV tracers, the Stokes' drag assumptions are stringent. When such conditions are not satisfied, mathematical models incorporating inertia and flow unsteadiness are often used to study tracer behavior. The Maxey-Riley equation [157]–[160] is a commonly used model which describes the motion of a small rigid sphere in non-uniform flow. The form of these equations is complex, involving spatial and temporal derivatives along with a time-integrated memory term. Since they are not being solved here, the equation is not reproduced.

The Maxey-Riley equation has been widely applied in studies examining tracer motion. For instance, it has been applied to the formation of rain droplets [161], the dispersion of aerosolized particles with applications to COVID 19 [162], and to the motion of ocean drifters [163], among others. It has also been applied to understanding the influence of tracer inertia on coherent structure formation [82], [164]. For instance, the study by Sudharsan, Brunton, and Riley [164] apply the Maxey-Riley equation to tracers embedded in the Double Gyre flow. This study showed that heavy tracers (relative to the fluid) accumulate near the FTLE ridges, whereas lighter tracers accumulate near the vortex centers. This result is relevant to the present study, since surface tracers will, by nature, be less dense than the fluid they are in.

The Maxey-Riley equation, however, is only suitable for modeling sufficiently small, spherical particles. The dynamics of larger tracers or those with aspherical geometries do not have a general, accepted model. That is not to suggest that such tracers are not significant in modern applications. Two prominent examples include the motion of ice floes in at Earth's poles (for example, see the work by Wilhelmus and collaborators [165], [166]), and the transport and settling of plastic particles and garbage in global waterways (e.g., the work of van Sebille et al. [167], [168] and of DiBenedetto et al. [169], [170]). From a slightly different vantage, the visual anemometry approach of Goldschmid et al. [171] utilizes imperfect tracers for flow measurement by inferring wind velocities from the recorded motion of vegetation. Due to the complexity imperfect tracer dynamics, these applications tend to be analyzed experimentally, as constructing a general numerical model in such cases is not tractable.

Aside from evaluating the degree to which surface tracers can be useful for flow measurement, this work provides a novel experimental approach to studying the motion of arbitrary tracers. Currently, a popular approach employed to study aspherical particles is to examine tracer shadows cast by a collimated light source [169], [172]. Part of the reason why shadowgraphy is required in these instances is the inherent difficulty of identifying tracers in natural images. The semantic flow measurement approach of Chapter 5, however, may enable such measurements without the need for shadowgraphy.

6.2 Impulsively Stopped Airfoil Experiment

In order to compare surface tracer motion with sub-surface fluid motion, it was necessary to test a flow which was highly repeatable and approximately 2-dimensional. Given the available experimental equipment at GALCIT in the NOAH laboratory, the flow of an impulsively stopped backwards-facing airfoil at high angle of attack was selected. The details surrounding the design and implementation of that experiment are provided in this section.

Experimental Setup

The impulsively stopped airfoil experiment was conducted in the NOAH free surface water channel facility described in Appendix B. Though typically run as a flowing water channel, the fluid in the test section was kept still for the present experiments. Instead, fluid motion was generated using NOAH's captive trajectory system (CTS), which is described in Section B.1 of Appendix B. A 2-dimensional NACA 0018 airfoil section with 10 cm chord was affixed to the head of the CTS manipulator and traversed through a pre-programmed trajectory. Image recording was synchronized to the CTS movement, guaranteeing repeatability between experimental runs. A schematic of the experimental setup in the NOAH facility is provided in Figure 6.1.

While preparing and testing the experiment, it was found that the most effective way to study surface and sub-surface flow simultaneously was to calculate surface tracer motion and PIV from the same image sequence. Therefore, PIV images were recorded from beneath the water channel with various tracers present on the water surface during records. Although the scattered light of the laser was sufficient to illuminate the underside of the surface tracers,

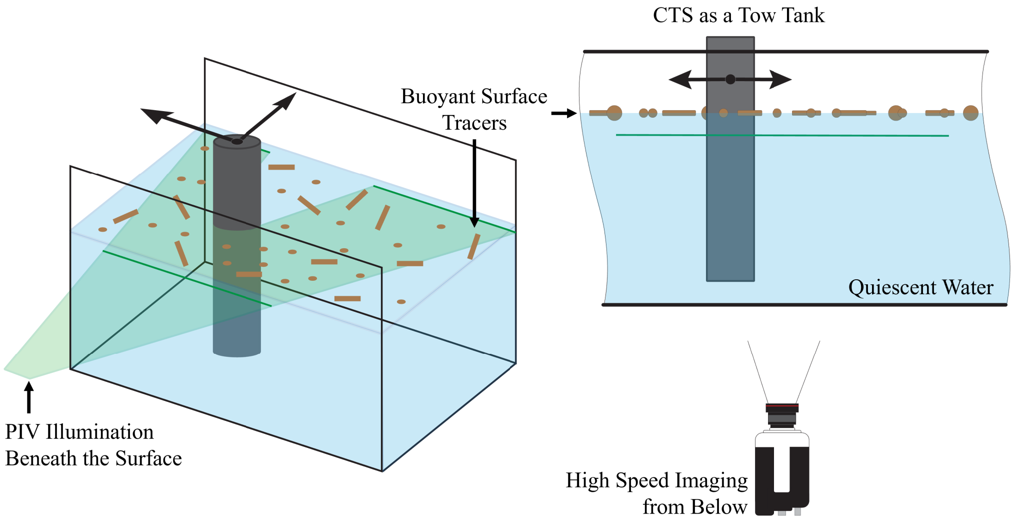


Figure 6.1: Illustrations of the experimental setup used for the surface tracer characterization experiment. Left indicates a 3D view of the test section and right indicates a 2D side view including the imaging equipment.

additional illumination was provided by an LED lamp to improve tracking performance. The plane of the laser used for PIV measurements was situated approximately 3.8 cm (1.5 inches) beneath the surface of the water so that surface movement could be better correlated with sub-surface measurements. If the distance between the PIV plane and the water surface were closer, surface deformations due to the induced flow would have struck the laser sheet and caused aberrations in PIV measurements.

Flow imaging was performed using a single Phantom Miro Lab 320 high-speed camera through a Nikon Nikkor 35mm f/1.8 lens. All data was collected with a 200 Hz sampling rate to ensure the quality of PIV image processing. Because the deep object tracking methods are sensitive to missed detections, the finite-time measurements benefit from a slower sampling rate. Therefore all videos were down sampled to 100Hz prior to tracer tracking. It is noted that the PIV plane was observed directly through the floor of the water channel and was not observed through the free surface. Therefore, the sub-surface flow was not obscured by surface tracers, nor was it distorted by surface deformations.

Flow Configuration

The flow selected for analysis is that of an impulsively stopped backwards-facing airfoil at high angle of attack. The chosen airfoil was a 2-dimensional wing section of NACA 0018 geometry with 0.1 m chord length and 0.45 m

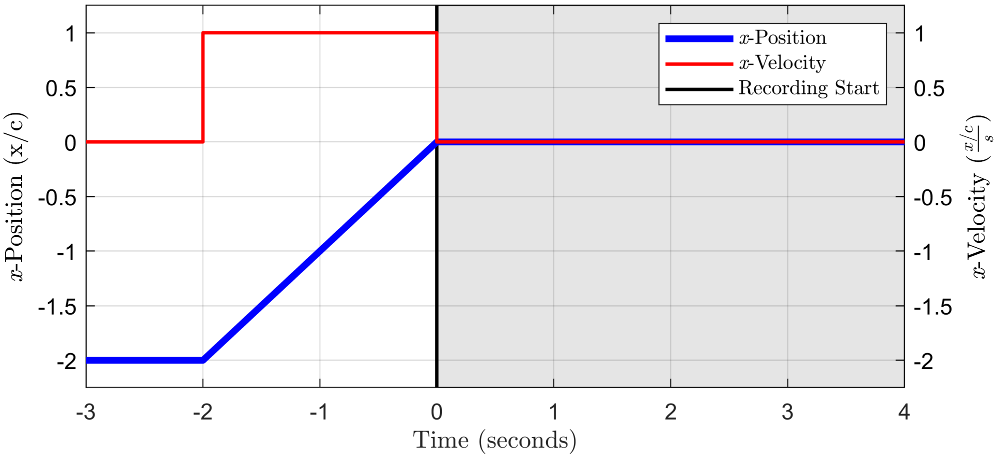


Figure 6.2: Graphical representation of the prescribed airfoil trajectory and subsequent image collection.

submerged span. The airfoil extended vertically from the CTS mount into the water to a position near the floor of the water channel. Other experiments with a similar configuration in the NOAH channel include those performed by Hooper et al. [173], [174].

Fluid motion was generated by traversing the submerged airfoil through a prescribed trajectory with the CTS. After performing a variety of tests, an impulsively started and stopped motion of the backwards-facing airfoil at high angle of attack was chosen for the experimental flow. The airfoil was initially placed outside of the camera's field of view and given a 135° angle of attack to the direction of prescribed motion. Thus, the airfoil was faced backwards at 45° towards the incoming flow. This was designed to encourage repeatable flow separation at the trailing edge of the airfoil.

At the start of each run, the airfoil was impulsively started to a prescribed velocity $\mathbf{v}_a$ and translated in a straight line over a fixed distance d_a. The airfoil was then impulsively stopped at a position $\mathbf{x}_{stop}$ such that the trailing edge of the airfoil protruded into the bottom of the experimental field of view. For the cases considered in this chapter, the airfoil velocity was set to $\mathbf{v}_a = 0.1$ m/s and the translation distance was fixed to $d_a = 0.2$ m $= 2c$, where $c = 0.1$ is the airfoil chord length. Figure 6.2 illustrates the prescribed trajectory.

The command to stop the airfoil translation was synchronized with the command to begin recording images. Therefore, the experiment examined the vortex dynamics immediately following the airfoil motion. Records of 2500 frames were recorded at 200 Hz, equating to $t = 12.5$ seconds of recorded motion or $t^* = \frac{t\mathbf{v}_a}{c} = 12.5$ in units nondimensionalized by the airfoil velocity and chord length. To ensure repeatability between runs, the airfoil was returned to its original position and left still for approximately 10 minutes between records.

Surface Tracer Conditions

Since the intent of this experiment was to evaluate the legitimacy of surface tracers as flow indicators, a variety of tracer configurations were tested. These included birch spheres (9.53 mm and 19.05 mm diameters), cylindrical birch rods (6.53 mm diameter cut between 12.7 and 50.8 mm lengths), and beeswax pellets (approximately spherical with 3.17 mm diameter). Tests were performed using various configurations of the tracers discussed above. The particular configurations along with the number of runs collected for each configuration are recorded in Table 6.1.

Given the studied flow and the tracers selected, the assumptions of Stokes' drag law almost certainly do not hold. Nevertheless, an estimate of the Stokes number is provided for all four of the tracer types implemented. To approximate the characteristic timescale of the flow, the chord length of the airfoil $c = 0.1$ m and its translation velocity $\mathbf{v}_a = 0.1$ m/s are used. Thus, the characteristic timescale of the flow $\tau_f = \frac{c}{\mathbf{v}_a} = 1$. Even though the tracers are not fully submerged in the water, the dynamic viscosity $\mu = 89 \times 10^-5$ kg/(m $\cdot$ s) is assumed for water at 25° C. The density of the birch spheres and rods is assumed to be $\rho_p = 640$ kg/m^3, and the density of beeswax is assumed to be $\rho_p = 964$ kg/m^3. Thus, by applying equation 6.1, the estimated Stokes number of the small spheres is $Stk_{SS} \approx 3.6$, of the large spheres is $Stk_{LS} \approx 14.5$, of the rods is $Stk_R \approx 6.4 - 103.1$ (Using the length of the rod. It is recognized that Stokes' drag is designed to apply only to spherical particles. This value is only computed for consideration.), and of the beeswax $Stk_{BW} = 0.6$. All of these values are far above the recommended Stokes number threshold of $Stk = 0.1$.

Aside from the factors accounted for by Stokes number, the particles examined in this experiment also differ from idealized tracers in other regards. Factors such as surface roughness, hydrophobicity, aspherical geometry, and surface tension will also cause the tracers' trajectories do deviate from ideal trajectories with similar initial conditions.

Surface tension and hyrdrophobicity influenced the experimental seeding procedure for the surface tracers by generating capillary forces which caused tracers form clusters [175]. Before each experimental run was conducted, the tracers were manually positioned within the frame of view and dispersed from clusters which had formed. This reduced the number of clusters formed at the start of each run. However, it is worth noting that the clusters themselves may be useful tracers in some regards.

Surface tension has also been seen to influence the interactions of vortices with a free surface. For instance, the work by Gharib, Willert, and Weigand [176], [177] explores the modes in which vortex rings and vortex tubes parallel to the free surface disconnect from themselves and connect to the free surface. It was found that modifying the water surface tension through the application of a surfactant altered vortex connection behavior. Since the vortices examined in this experiment are normal to the free surface and the wood and beeswax tracers do not alter the surface tension of the water, surface tension measurements were not collected during experimentation.

6.3 Data Processing Approach

Defining the appropriate data-processing pipeline was critical to the success of this experiment. On one hand, obtaining accurate PIV measurements was contingent upon removing the visible surface tracers from the images prior to velocimetry. On the other hand, semantic kinematic measurements required the ability to clearly distinguish the same tracers that were removed from PIV

Table 6.1: Surface tracer configurations with the number of datasets recorded per configuration.

Configuration	3.17 mm Pellets	9.53 mm Spheres	19.05 mm Spheres	6.53 mm Rods	Runs
Approx. *Stk*	0.6	3.6	14.5	6.4–103.1	
Control					10
Beeswax	✓				5
Small Spheres		✓			3
Large Spheres			✓		3
All Spheres		✓	✓		3
Rods				✓	3
Spheres and Rods		✓	✓	✓	3
All Debris	✓	✓	✓	✓	3

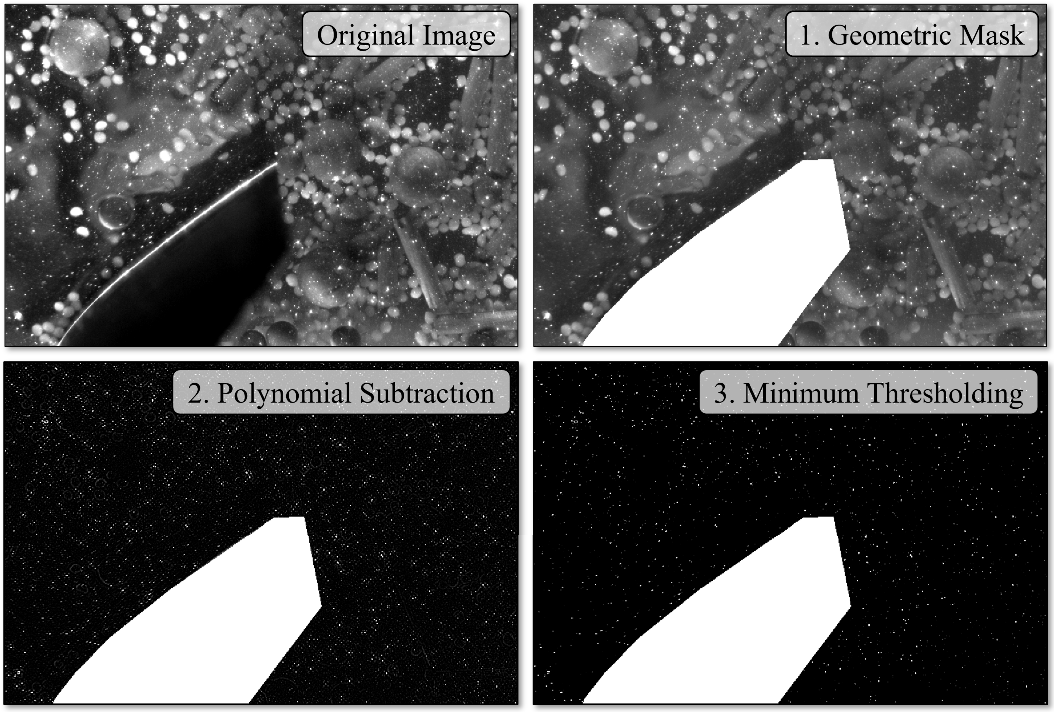


Figure 6.3: Pre-processing steps applied to raw images prior to PIV velocity estimation. First, a mask is applied to remove physical obstructions from calculations. Then, background content including visible surface tracers are removed by the subtraction of a polynomial locally fitted to image intensities. Finally, additional evidence of surface tracers are removed by subtracting values under a threshold. The result contains only PIV particles to be used in the computations.

processing. This section outlines the process used to satisfy these contradictory objectives.

PIV Processing

Surface tracer removal from PIV images was accomplished through a multi-step process illustrated in Figure 6.3. First, the permanent obstruction of the airfoil protruding into the frame was removed by geometric masking. This step involved identifying within an image which pixels were not relevant to the flow and removing them by hand. Since the airfoil did not move during the recordings, only one mask was needed for all frames. The odd mask geometry observed in Figure 6.3 results from the fact that the 2D airfoil protrudes out of the measurement plane towards the camera. Moreover, there is a small elastic oscillation of the blade that occured as a result of the impulsive stop which was also covered by the mask.

Once the mask had been applied, visible surface tracers were eliminated from the image via subtracting a local polynomial fit of image intensities from within a surrounding 7 × 7 pixel window at each pixel location. This is effectively an advanced form of median subtraction, as PIV particles—which are characterized by spikes in pixel intensity—are not captured by the smooth polynomial fit of the underlying scene. A variety of window sizes were tested while establishing a processing pipeline, but it was found that larger windows did not effectively remove surface tracer edges, and smaller windows removed too many particles. Once the polynomial fit was removed, all pixels with values under a threshold were set to zero to eliminate any remaining evidence of surface tracers.

Once the raw images had been pre-processed to remove evidence of surface tracers, 2-dimension, 2-component (2D2C) PIV was computed on the image sequence. Four passes of the PIV algorithm were applied to the images. The first pass used a 64 × 64 circular pixel window with 50% overlap, and the following three passes used a 24 × 24 circular pixel window with 75% overlap. Vector validation was performed on the resulting vector fields. Any vectors with velocity larger than 0.4 m/s were removed outright, and a universal outlier detection scheme based on median filtering was applied to eliminate spurious velocities. All removed vectors were filled via interpolation.

All of the runs listed in Table 6.1 were processed using the described approach. A control set of data containing ten runs without any surface tracers and processed without image pre-processing were collected to ensure that deviations in PIV velocity fields due to surface tracers were minimal. Two sets of ensemble averaged velocity fields were computed from the recorded data. The first set averaged only the clean data, and the second averaged over all runs. Comparing these results yielded little difference, so the average of all data was henceforth used as a baseline for flow behavior.

Surface Tracer Analysis

The motion of surface tracers was examined using the semantic detection and tracking approach developed in Chapter 5. Tracer detection was achieved using the Mask-RCNN detection model [147] with a pre-trained ResNet50 Backbone [154]. The Mask-RCNN model was fine-tuned to identify surface tracers using training data collected from the videos recorded by the process outlined in Section B.3 of Appendix B. These training images were added to the set collected and utilized in Chapter 5 in order to develop an improved semantic understanding of debris particles. During inference, 256 × 256 pixel windows with 25% overlap were supplied to the detection model for all configurations except for the Large Spheres case, which used 1024×1024 pixel windows with 25% and the Rods case, which used 512 × 512 pixel windows with 25%. The centroids of tracer masks were recorded and considered as the tracer locations on the flow surface.

Tracer trajectories were constructed using the template-matching scheme applied in the asymmetric wake experiment of Chapter 5 and thoroughly detailed in Appendix C. Trajectories with fewer than three detections were automatically discarded. Initial trajectories were merged by the merging protocol described in Appendix C. Results were filtered and smoothed by the application of a median filter with a kernel length of 5 and a Gaussian filter with a kernel length of 10.

Calibration Alignment

Although both sub-surface and surface flow measurements were recorded from the same set of images, a separate calibration was required for each measurement plane. Calibration in the plane of PIV was performed using a LaVision Type 11 calibration target. The target was translated to various positions in the field of view, and a pinhole calibration model with radial and tangential distortion correction was fit to the collection of calibration images. Calibration on the surface of the water channel was achieved using the ChArUco method outlined in Section B.2 of Appendix B. A large ChArUco calibration target was floated on the water surface such that it filled the entire field of view of the camera.

Both of these calibration procedures successfully mapped sensor values to the world coordinates with the appropriate scale, but they were not naturally aligned. The datum in each frame was not aligned, and there was slight difference in rotation angles in x, y, and z axes. To align the calibration planes so that one-to-one measurements could be made, masks of the airfoil were used as landmarks to be aligned. Comparing the transformed positions of the masked airfoil between calibrations, a 3 × 3 rotation matrix and a 2-D translation in x and y were found by iterated inspection and applied to the homography matrix used in the LGR calibration. The results are displayed in Figure 6.7, which is discussed in a later section.

Procedure for Interpolation to Fields

In this chapter, the results between the surface tracking and the sub-surface PIV are compared, in part, by interpolating the scattered data observed by the surface tracers to the field positions where PIV is estimated. Converting the data from Lagrangian velocities to Eulerian velocity fields can be sensitive to the particular scheme used for interpolation, especially when there is some measure of uncertainty in the scattered data. Performing a direct linear or even polynomial interpolation typically yields poor results, as it ensures the interpolated field passes through all of the scattered data points.

Other interpolation schemes handle noise and uncertainty gracefully by providing the option for interpolated fields to deviate from the scattered data values to some degree. One such interpolation scheme is radial basis function (RBF) interpolation, which is employed in this chapter for approximating velocity fields from Lagrangian surface tracer measurements. RBF interpolation is a method which constructs a function that is a linear combination of radial basis functions, which depend only on the distance between the point of evaluation and the scattered data points. The function is constructed such that it interpolates the scattered data points and minimizes the difference between the interpolated field and the scattered data. A smoothing parameter can be adjusted to control the degree to which the interpolated field deviates from the scattered data. The implementation of RBF interpolation used in this chapter is the `scipy.interpolate.RBFInterpolator` function provided by the SciPy Python package [178]. A linear kernel with a smoothing parameter of $\lambda = 0.01$ was applied in most cases presented. More details about the mechanics of the method can be found in the SciPy documentation and the references therein.

6.4 Character of the Flow

In this section, the qualitative behavior of the flow is briefly examined purely through the lens of PIV. Considering the dynamics of the flow in isolation helps to provide context for the comparisons to be made in later sections.

Figure 6.4 displays sample frames from the progression of the flow immediately following the impulsive stop of the airfoil. Vorticity computed from PIV (heavily smoothed with a 15×15 Gaussian smoothing kernel) is displayed atop the image from which it is computed. The flow is shown at different instances throughout its progression from top to bottom, left to right, indicated by the nondimensional timestamp in the lower right hand corner of each

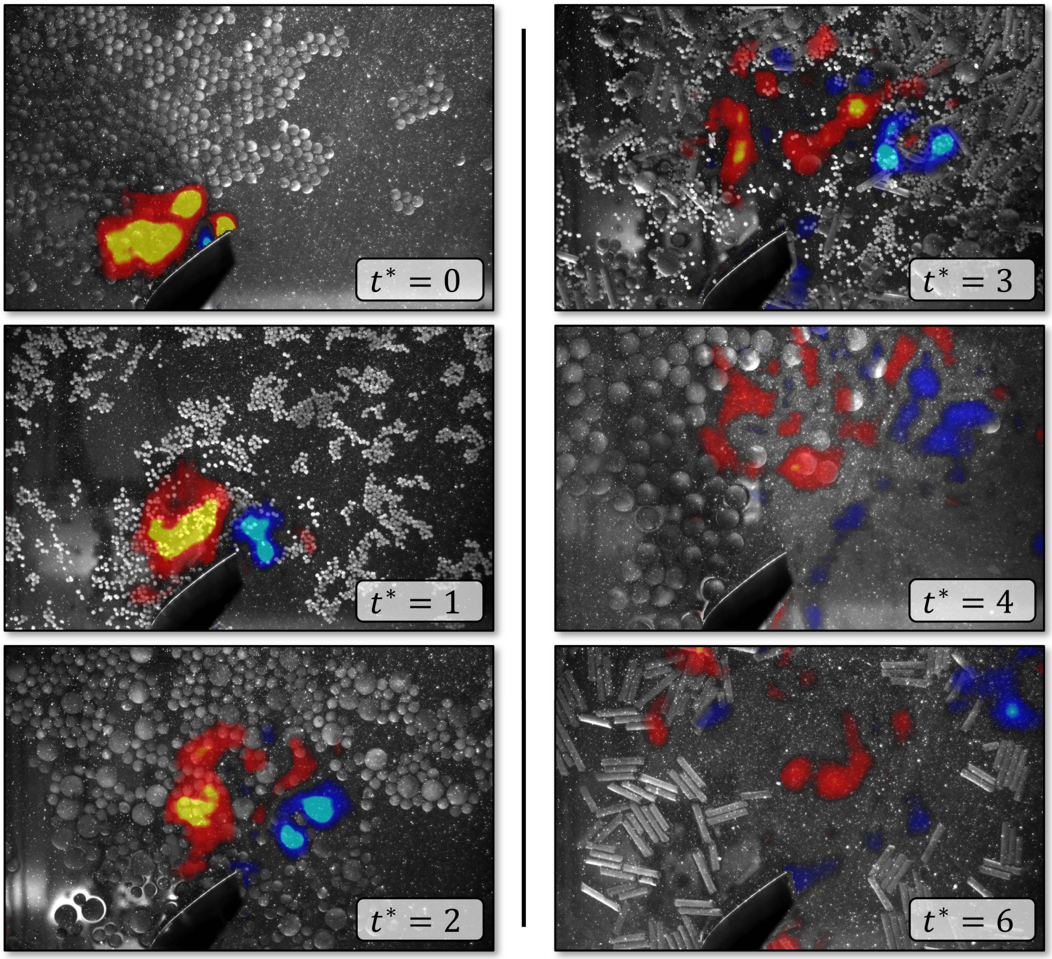


Figure 6.4: Smoothed vorticity computed by PIV for the impulsively stopped airfoil flow. Positive vorticity is indicated by hot colors, and negative vorticity by cold colors. Underlying images represent the data from which PIV is taken. These exhibit different surface tracer configurations which, from top to bottom, left to right are: small spheres, beeswax, all spheres, all debris, large spheres, and rods.

frame. Positive vorticity is colored in warm colors, while negative vorticity is given cool coloring. The colormap is kept constant throughout the progression to illustrate how vorticity dissipates over time.

Frames are taken from computations performed on different surface tracer configurations, which are displayed underneath the vorticity. The images of tracers are included in the display in order to give context for the appearance of various tracers in the images and to visualize qualitatively how tracers respond to the underlying flow over time. In the same order as the timestamps, the tracer configurations displayed are: small spheres, beeswax, all spheres, all debris, large spheres, and rods.

Flow measurement begins as the airfoil ceases movement. Already, the sharp point of the trailing edge and the large angle of attack have generated a strong counterclockwise vortex as the airfoil was moved to its final position. This vortex is visualized in frame $t^* = 0$ as the large yellow cloud of vorticity. As the flow develops, the momentum induced by the motion forces fluid around the trailing edge of the airfoil creating a clockwise rotating vortex indicated in blue at time $t^* = 1$. These vortices are directed upwards by the fixed geometry of the airfoil and the velocities induced by the vortices. They are observed moving together towards the top of the frame by $t^* = 2$. At this point, the vortices begin to break down and dissipate into broader, less concentrated regions of vorticity. As time continues, energy dissipates further, and vorticity begins to neutralize throughout the domain. The last fluid motions observed over the interval are vortices shed off of the leading edge of the airfoil just off screen of the bottom edge of the frame.

An important feature to notice in the surface tracer motion is that a void develops in their spatial distribution in the region of the observed vortices as they begin to break down. Multiple contributors may be responsible for this phenomenon which was observed in most of the experimental cases. First, the momentum of the tracers themselves may lead to the opening of surface tracer voids. As tracers are entrained into the vortices they begin to rotate with them. Near the center of these vortices, the rotation can be relatively high—peak vorticity in this flow is over 10 radians per second. Because the surface tracers have mass, this rotational motion will generate a centripetal force pulling them away from the center of the vortex. If this force becomes large enough, it may pull the tracers away from the core of the vortex, opening up the gaps which are observed.

Three-dimensionality of the flow may also be a contributor to surface tracer dispersion. The gap seems to grow as the two coherent vortices break down into a more chaotic tangle of smaller vortices. While only planar velocities are observed in this experiment, the underlying flow field at times $t^* \geq 3$ surely exhibits three-dimensional motion. These three-dimensional motions likely impact the flow at the surface, but buoyant surface tracers will move primarily in-plane due to their buoyancy. Thus, the increasing three-dimensionality of the flow may contribute to the opening of voids in the surface tracer distribution.

Regardless of the mechanism exposing the gaps in the surface tracers, their existence affects what is measurable on the surface of the water. Where the gaps are opened, interpolated field quantities will be less reliable.

Tracer clustering due to capillary forces [175] is visible in the frames of Figure 6.4. The clusters are most prominently observed in the small spheres ($t^* = 0$), the beeswax ($t^* = 1$), and in the rods ($t^* = 6$) which cluster in parallel, but the phenomenon was prevalent across all surface tracer conditions. When the underlying gradients were small, the clusters tended to move as a single unit, but when the flow gradients beneath the clusters were strong, the dynamic forces on the tracers overcame the capillary adhesion, causing them to disperse from the clusters.

In an attempt to characterize the reliability of the PIV data by measurement time, an analysis of the PIV error from the mean was conducted. An ensemble average of all of the PIV flow fields listed in Table 6.1 was calculated and the root-mean-square error (RMSE) from the mean was computed for each individual run according to

$$\mathrm{RMSE}(|\mathbf{v}|) = \sqrt{\frac{1}{n}\sum_{i=0}^{n}\left(|\mathbf{v}_i| - \overline{|\mathbf{v}|}\right)^2}, \tag{6.4}$$

where $|\mathbf{v}_i|$ represents velocity magnitude of a PIV vector and $\overline{|\mathbf{v}|}$ represents the ensemble average of the velocity magnitude of that vector from all samples. These errors were then averaged over all runs to obtain the average RMSE at each vector location and time in the recordings. The change in RMSE over time is considered in absolute terms by the mean RMSE per frame of PIV data, and in relative terms by the norm of the RMSE in a frame divided by the norm of velocity magnitude in a frame. The results of these computations are presented in Figure 6.5.

The spatial mean of the RMSE of velocity indicates that the total measurement error in the flow is relatively low at all times, but is lowest at the start of the recording. The error magnitude increases to a peak at approximately $t^* = 4$ and then decreases to a steady state by $t^* = 0.8$. These results are consistent with what might be expected from Figure 6.4, where the flow field at $t^* = 4$ appears disordered, but still exhibits higher energy levels. Prior to this, the flow motions are high energy, but more structured. Beyond it, the energy

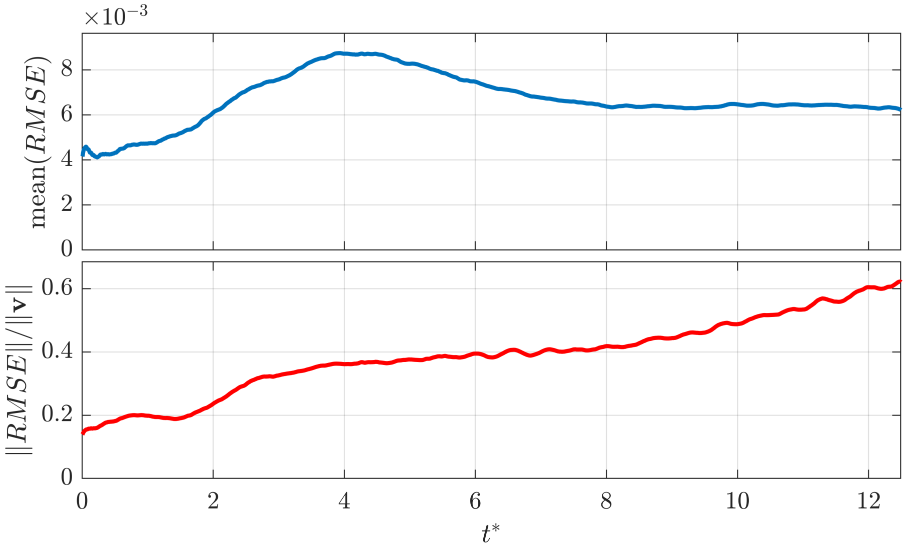


Figure 6.5: Examination of PIV uncertainty over time. Top: Spatial mean root-mean-square deviation from the ensemble average of all velocity fields. Bottom: Relative deviation of velocity magnitude as measured by the norm of each RMSE field over the norm of the corresponding averaged velocity magnitude field at each instant in time.

decreases, but the disorder increases. The spatial mean of RMSE is tied to the amount of energy in the flow, so even though the structure of the flow is chaotic beyond $t^* = 4$, values still decrease since the energy is dissipating.

The relative deviation of velocity magnitude shows a different trend. This metric accounts for the amount of energy in the flow by normalizing the error by the norm of the ensemble averaged velocity magnitudes. Thus, this is more strictly a measure of the repeatability of the experiment at a given instant in time. By this analysis, it is evident that flow variability continues to grow in throughout the interval. The most reliable times for performing comparisons are when $t^* \leq 2$. This is also consistent with what was observed in Figure 6.4, as the strong vortices which characterize the initial fluid motion have not yet begun to break down into smaller vortices and dissipate their energy.

This analysis suggests that, when judging between cases, the fairest comparisons will made early in the recording where the absolute and relative error are the lowest. This is taken into consideration in the following sections as results are discussed.

6.5 Quality of Surface Tracer Detection and Tracking

In the previous section, the character of the flow and the trustworthiness of PIV measurements were considered. The quality of tracer detections and associated trajectories must also be examined. A sufficient number of reliable trajectories must be found for any conclusions relating surface tracer motion to sub-surface dynamics to be considered reliable. Here, the effectiveness of semantic object detection and tracking is examined for the various categories of surface tracers tested.

Detection Results

Figure 6.6 displays sample frames from each surface tracer configuration with detected tracer segmentation masks overlaid with a unique color. Each detection was assigned a single color, therefore, adjacent tracers with the same color represent a single detection. By the results, the detection procedure appears to be effective at identifying a majority of the tracers in all configurations to varying degrees of success. The model performs best on for the configurations containing small or large spheres only. In these cases, almost every tracer is identified in the sample frame and the masks generally cover all of the tracer they represent. There are some aberrations in the “all spheres” configuration where the masks of many large spheres appear to be incomplete (i.e., the mask exists on the tracer, but not over the entire tracer). This is likely due to the fact that 256×256 windows were used for detection, which was occasionally too small to fully detect the large spheres.

Somewhat surprisingly, beeswax pellets were also detected with relatively high accuracy. This is especially true in regions of the flow where the pellets are broken apart from the clusters that they naturally form. This is useful in flow measurement applications, as fluid movements like vortices tend to break apart the groups of pellets, making them more effective flow tracers. The detections of beeswax pellets are the least accurate when they are gathered in groups. In this case, the detector seems to often identify a group of tracers as a single tracer. This is not a critical failure mode, as the group of tracers can act as a single tracer, but it complicates trajectory formation as the clusters can morph or break up, causing ambiguity.

Rods were the least accurately detected of all tracers considered in this experiment. This is consistent with what was found in Chapter 5. Rods tend to gather together lengthwise, presumably due to surface tension effects. When

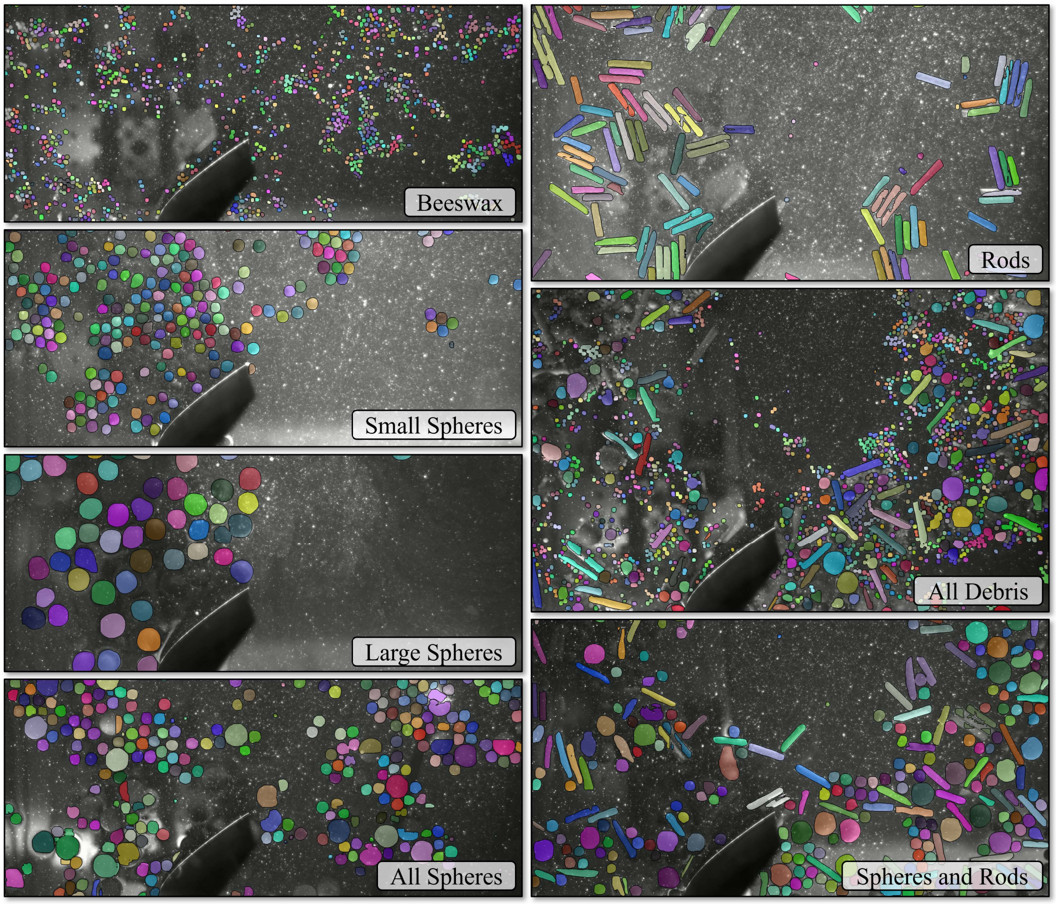


Figure 6.6: Sample frames (some cropped) from each surface tracer configuration demonstrating the quality of surface tracer detections. Detected tracers are indicated by randomly colored segmentation masks. Each detected tracer is marked with a single color. Thus, any adjacent tracers that are indicated by a single color were detected as a single tracer (as was the case with many groups of rods).

such grouping occurs, the detector often identifies neighboring rods which have aligned in parallel as a single tracer. In the example image from Figure 6.6, this phenomenon is frequently observed. Interestingly, the rods tend to be better detected in the "spheres and rods" and "all debris" cases than when they are alone, possibly as a result other tracers separating them from one another.

The "all debris" condition performed fairly well, but demonstrated the same issues that were apparent with the other data sets. Namely, the larger spheres and rods were occasionally missed due to the small window size required for the beeswax, and tracer clumps were often identified as single tracers. However, despite missed detections and mis-identifications, the "all debris" case contained many tracers, which was beneficial for flow field estimation.

Tracking Results

Tracking effectiveness was briefly considered through the computation of basic statistics on trajectory lengths. Table 6.2 contains selected quantities. For each configuration, the number of total identified trajectories N over the duration of the recording is recorded along with the mean length l_{mean}, the standard deviation of length l_{std}, the median length l_{med}, and the percentages of tracers over 10 detections, 50 detections, and 100 detections long. In general, larger numbers represent better tracking performance since they indicate tracers which follow the flow better for longer.

Considering the statistical quantities presented in Table 6.2 suggests that the cases tracking beeswax and small spheres outperform the the other tracer configurations. These exhibit relatively long mean and median trajectories, and contain a greater percentage of long trajectories. The worst performing tracers are the rods, which display the smallest values across all categories. The large spheres perform well, but are limited in the number of trajectories available. Additionally, they seem to exhibit weaker performance over longer intervals.

The trajectory length data also illuminates some inefficiencies of the chosen tracking approach. As in the experiment from Chapter 5, the distribution of trajectory length for all conditions appears to be heavy-tailed. In all exam-

Table 6.2: Surface tracer configurations with the number of datasets recorded per configuration. Number of recorded trajectories is reported along with mean, median, and standard deviation of trajectory lengths and the percentage of trajectories longer than 10 ($t^* = 0.1$), 50 ($t^* = 0.5$), and 100 ($t^* = 1.0$) associated detections.

Configuration	N	l_{mean}	l_{std}	l_{med}	$\%(10^+)$	$\%(50^+)$	$\%(100^+)$
Beeswax	140636	15.53	30.23	6.0	34.2	6.2	2.1
Small Spheres	30225	15.77	26.52	7.0	37.9	5.9	1.8
Large Spheres	9303	14.25	19.74	7.0	40.3	4.6	1.0
All Spheres	44797	14.28	25.04	6.0	32.9	5.3	1.6
Rods	17725	10.95	18.49	5.0	27.6	2.8	0.7
Spheres & Rods	38084	12.37	21.67	6.0	31.1	3.5	1.0
All Debris	169711	11.95	21.97	5.0	28.5	3.7	1.1

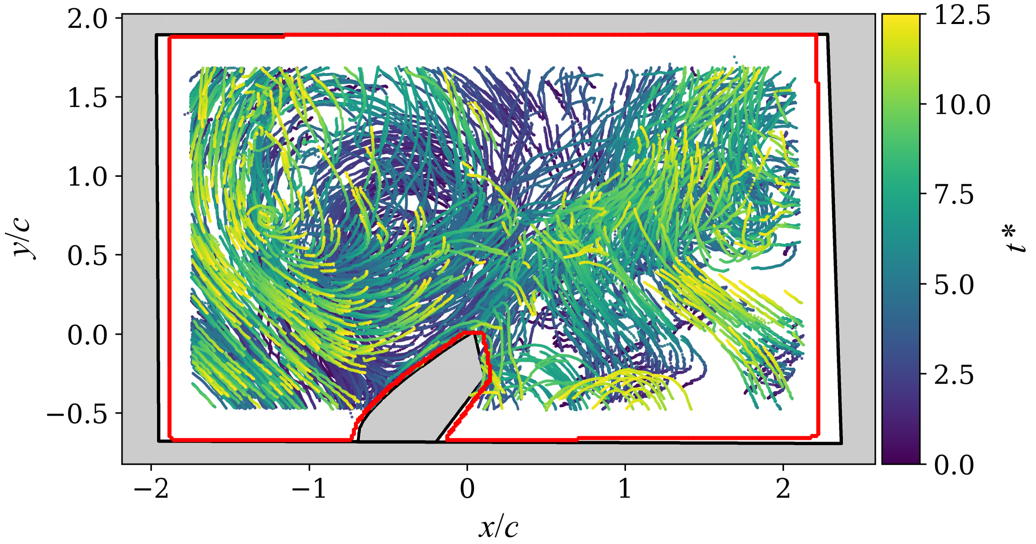


Figure 6.7: Sample trajectories identified from a recording containing beeswax pellets as the surface tracers of interest. Tracers displayed are all those with at least 100 associated detections ($t^* = 1.0$). The red outline indicates the calibrated frame within which PIV data was collected, and the black outline indicates the calibrated frame where object detection and tracking occurred. Trajectories do not extend to the edge of the frame due to the present implementation of object tracking.

ples, more than 50 percent of the recorded trajectories contain fewer than 10 detections. Additionally, the number of trajectories identified for each configuration seems to be bloated. This is certainly due to the heavy-tailed nature of the data, but may also indicate that the implemented algorithm assigns detections to multiple trajectories. These observations encourage improvement to the tracking algorithms chosen for these analyses.

Finally, Figure 6.7 displays trajectories of a subset of tracers from the beeswax configuration. The trajectories shown represent all those with more than 100 associated detections. Coloring indicates the time at which a sample was recorded. The black frame and outline indicate the field of view at the surface of the water and the mask of the airfoil in that frame. The red frame represents the field of view of PIV images underneath the surface of the water. Trajectories do not extend to the boundary of the outline due to the present implementation template-matching tracking algorithm.

6.6 Results

The results of the impulsively stopped airfoil experiment are provided in two parts. In both instances, velocity fields are interpolated from the measured velocities of surface tracers and compared with the results computed by PIV. Figures 6.8 and 6.9 examine estimated velocity magnitude fields over time via the beeswax tracer configuration. Then, Figures 6.11 and 6.10 consider the behavior of the different tracer conditions at a single instant in the time through field estimates of velocity magnitude and vorticity.

To estimate velocity fields from the motion of surface tracers, individual tracer velocities are computed from trajectories and interpolated to the same grid points utilized in the PIV data. At each time step, some of the tracks exhibit erroneous velocity vectors. Thus, a proximity-based velocity filtering algorithm was developed to ensure fields were cleanly computed. The velocities of nearest-neighbors to a given tracer were recorded, and if the Mahalonobis distance of the tracer's velocity to neighboring velocities—defined by

$$d_M = \sqrt{\left(\mathbf{v} - \overline{\mathbf{v}}\right)^\top \mathbf{S}_\mathbf{v} \left(\mathbf{v} - \overline{\mathbf{v}}\right)}, \tag{6.5}$$

where $\mathbf{S}_\mathbf{v}$ is the covariance of the local velocity data—was outside some threshold, it was removed. For the data presented in this section and the next, filtering was performed with $k = 15$ nearest-neighbors and a d_M threshold of 2. Once tracer velocities had been filtered, Eulerian velocity fields were approximated by the RBF interpolation procedure described in Section 6.3.

Surface and Sub-Surface Motion Over Time

Surface tracer motion is compared to sub-surface motion by examining velocity fields interpolated from beeswax pellet trajectories. Two figures are presented to illustrate key features.

First, Figure 6.8 assesses the beeswax pellet motion over the duration of the flow for single test run by overlaying tracer position atop interpolated fields of velocity magnitude. Fields are computed at nondimensionalized times $t^* = [0.0, 1.0, 2.0, 3.0, 4.0, 6.0, 8.0, 10.0]$. If pellets are not approximately stationary, then an arrow indicating the velocity vector is plotted with its root at the tracer position. Color mapping is kept constant for all times.

Figure 6.9 then compares fields interpolated from surface tracer motion to those approximated by the ensemble average of PIV recordings. Once again, fields are presented by the color mapping displayed within each frame. The

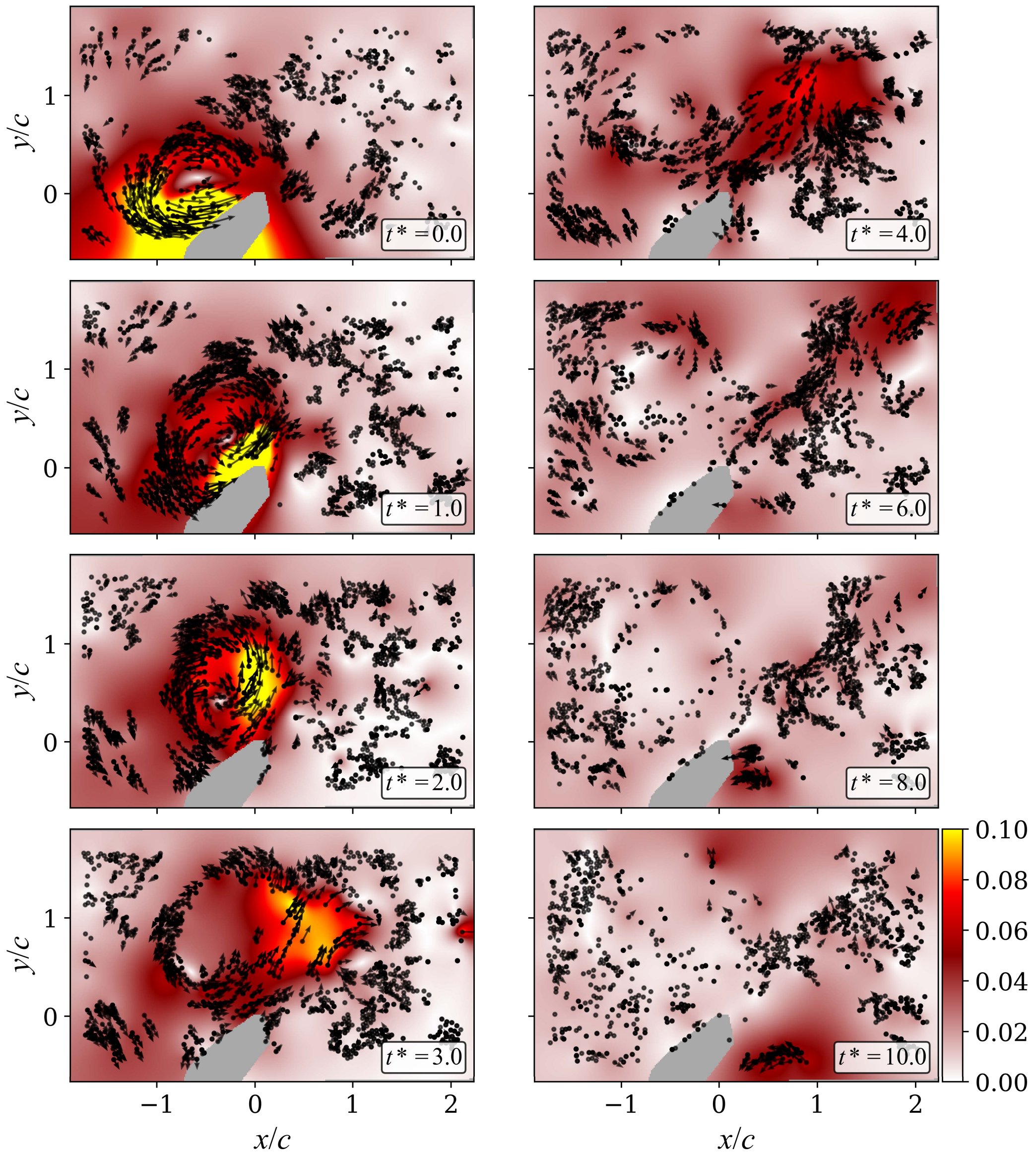


Figure 6.8: Fields of velocity magnitude interpolated from the Lagrangian velocities of beeswax surface tracers observed over time. Fields are interpolated at the time indicated in each frame. Tracers are represented by gray dots. Velocity vectors are indicated if the tracers are not approximately stationary.

left column colors the fields using the same dataset presented in Figure 6.8, while the right column colors the field by the all-run ensemble average of PIV measurements. The color mapping values are consistent within all frames. Black contour lines overlaid on the pseudo-color plots in both columns represent isocontours of velocity magnitude from the ensemble averaged PIV data at the indicated time. These are provided for ease of reference. The flow is measured at $t^* = [1.0, 2.0, 4.0, 6.0]$. As discussed in section 6.4, the correlation of structures between the PIV ensemble mean and the individual runs is expected to decrease as time increases.

Before considering the method comparisons made in Figure 6.9, a few observations from the progression of frames displayed in Figure 6.8 are discussed. First, interpolation effects play a relatively large role in what is observed. This is especially obvious at $t^* = 0.0$, where the large velocities from the leading edge vortex extend to the entire area around the airfoil in a way that is not expected for the flow. This is because there are no other surface tracer measurements in those regions near the bottom edge of the frame. These errors are also evident in the data presented at $t^* \geq 3.0$ as space opens between surface tracers. That regions in the flow are often unrepresented by surface tracers represents a notable limitation of their use for flow measurement purposes.

Figure 6.8 also highlights the opening of surface tracer voids over the course of the flow. At times $t^* \leq 2.0$, tracers are observed to be entrained in the strong leading edge vortex. Beyond this point, it appears that the momentum of the tracers carries them outward away from where the vortex core may be. This void stays open in the surface of the flow for the remainder of the interval, even as tracers begin to cease movement by $t^* = 10.0$. Thus, the results observed in Figure 6.8 align with the similar discussion on Figure 6.4.

Having considered the motion of beeswax tracers in isolation, a comparison with PIV data is now made. The results presented in Figure 6.9 indicate that the surface tracers, while not perfectly representing the dynamics of the flow immediately beneath the surface, largely capture the principal features observed by PIV. This will be examined chronologically in flow time.

The first frame, recorded at $t^* = 1.0$, best resembles the ensemble PIV average. The highest velocities occur near the trailing edge of the airfoil, as the fluid entrained by the leading edge vortex generated by the airfoil motion is ejected around the obstacle. The high velocities found in this region of the flow closely correspond to the overlaid contours from the PIV data.

The frame recorded at $t^* = 2.0$ begins to show greater deviations from the PIV ensemble average. The largest deviations appear in the location of the fluid ejection from the trailing edge, where the region of large interpolated velocity magnitudes appear to curve upward from the largest values observed in the PIV. This is observed across tracer classes, and will be discussed further with the next analysis. Additionally, the velocity magnitudes recorded from surface tracers are smaller than those observed in the PIV data.

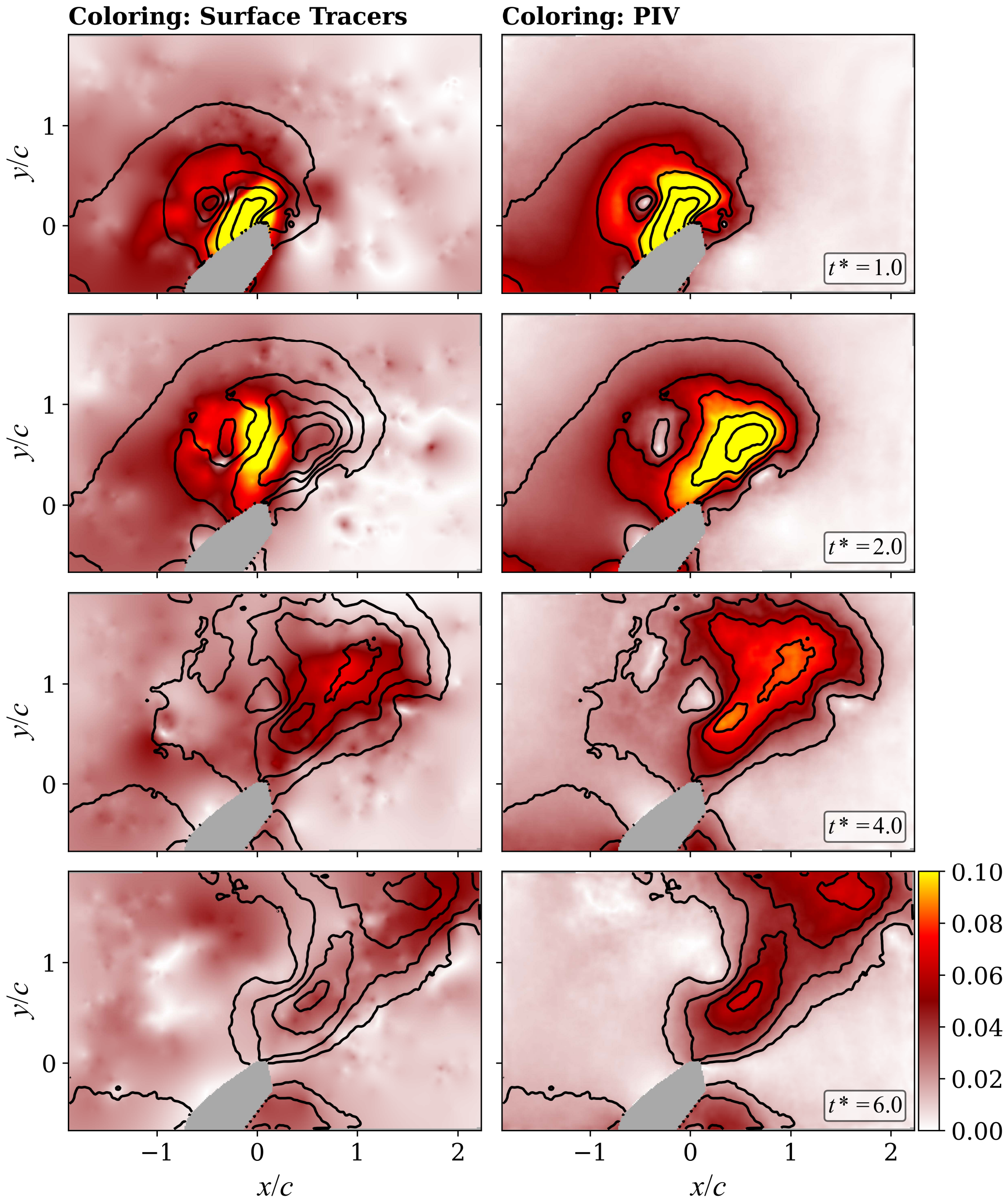


Figure 6.9: Comparison of interpolated fields of velocity magnitude computed from beeswax surface tracers compared with the ensemble average of velocity magnitude fields computed by PIV. Left column displays velocity magnitude fields interpolated from the surface tracer velocities. Right column displays the PIV fields. In both columns, isocontours of the PIV ensemble average are overlaid.

In the next frame, at $t^* = 4.0$, the position of greatest surface velocities seem to have re-aligned with the PIV data. The dominant motion observed in the frame is a diagonal stream between the two counter-rotating vortices that formed by the airfoil motion. The surface tracers and the PIV both seem to capture this effect, although, the tracers represent it with smaller magnitudes and reduced spatial extent. In both frames, the measurements exhibit small values of velocity magnitude.

In the final frame displayed, at $t^* = 6.0$, there appears to be some correlation between observed surface and sub-surface flows in the upper right-hand corner of the velocity field and just behind the airfoil. These regions appear to exist where the strongest vortices in the flow were identified at this time in Figure 6.4. The large values in the upper right corner are the result of the clockwise vortex which was shed from the trailing edge of the airfoil and the one behind the airfoil is induced by vortex shedding around the blade. However, the PIV field indicates a diagonal swath of relatively large velocity magnitude, which seems to be less well represented by the surface tracers. Moreover, the surface tracers seem to exhibit larger velocities towards the left-hand side of the frame. Comparing this with the velocity vectors shown in Figure 6.8 seems to indicate that these velocities are the result of a large-scale counterclockwise rotation which is not as apparent in the PIV data.

Briefly, it is noted that analyzing the flow in an instantaneous sense (i.e., using velocity fields, vorticity, or some other metric defined at a moment in time) provides flexibility when working with imperfect tracers. The velocity magnitude was distorted by the surface tracer observations at time $t^* = 2.0$, but resembled the flow more closely before at $t^* = 1.0$ and after at $t^* = 4.0$. In that sense, the analysis is able to “recover” from deviations if instantaneous metrics are being considered. Alternatively, trajectory-based metrics like FTLE or LAVD would compound errors over time due to imperfections in tracers.

Surface Motion Observed Varying Tracer Type

The next analysis considers the structure detection capability of the various types of tested tracer configurations at the fixed point in time. The nondimensional time $t^* = 2.0$ was selected as the case under consideration since it contains strong vortical features that have not yet begun to dissipate. The results of this study are presented in Figures 6.10 and 6.11, which should be considered together.

The left-hand column of Figures 6.10 and 6.11 displays velocity magnitude fields as in the previous analysis. The right-hand column displays out-of-plane

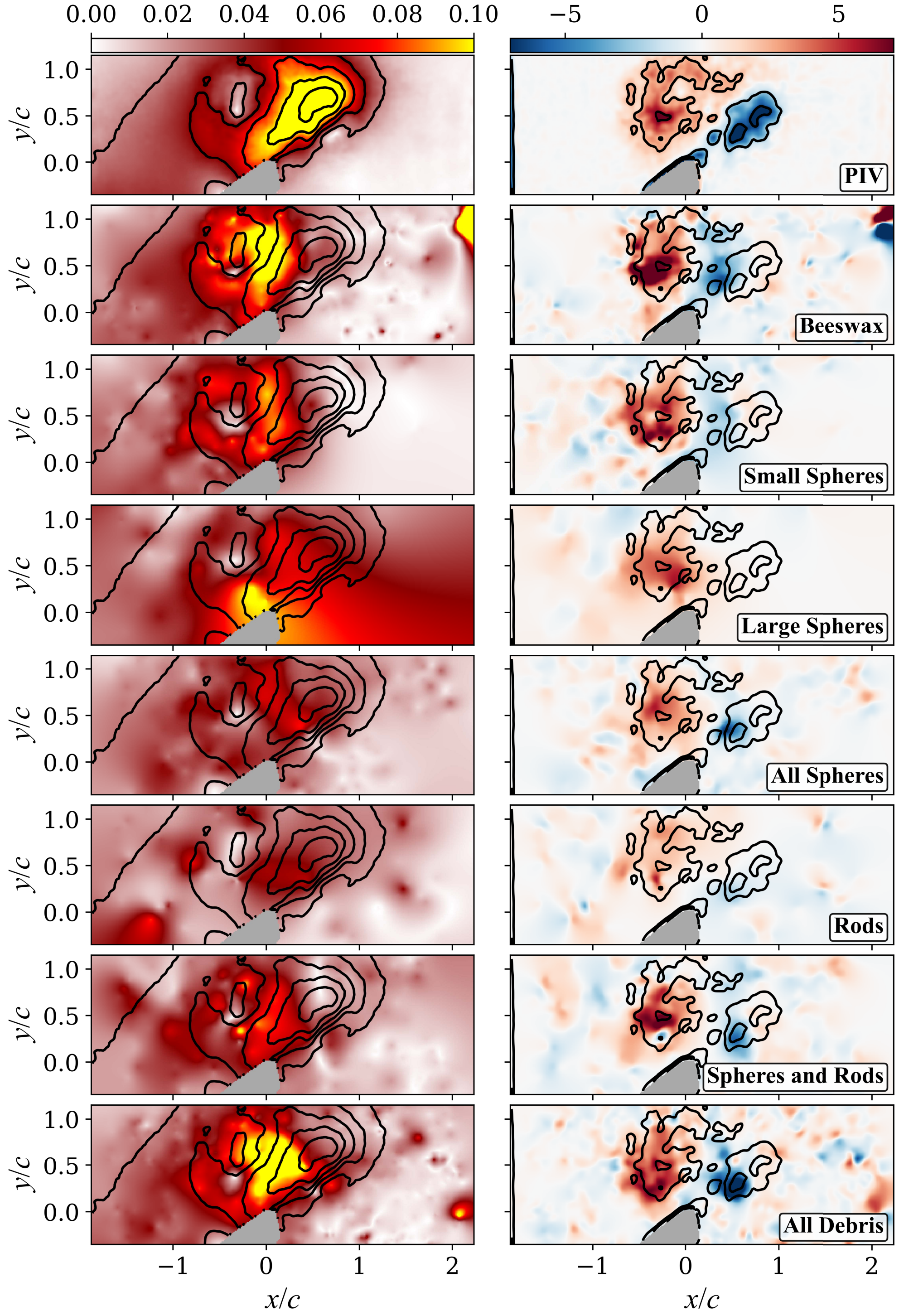


Figure 6.10: Comparison of flow fields estimated from various surface tracer cases with ensemble averaged PIV data at time $t^* = 2.0$. Left column displays velocity magnitude. Right column displays out-of-plane vorticity. Isocontours in all frames are taken from PIV fields in the top row.

vorticity computed by finite-differences over the given velocity fields. Indicated in each vorticity plot are the measurement conditions used to generate that data. The top row is generated from the ensemble average of PIV data. The PIV vorticity was lightly smoothed using a 5 × 5 Gaussian filter. Figure 6.10 facilitates comparison between PIV and surface flow approximations by overlaying isocontours of the PIV ensemble data on all frames. Because the location of tracers determines the structure of interpolated fields, Figure 6.11 indicates the centroid location of each tracer as a gray dot. The same data is used between figures.

It is clear in Figures 6.10 and 6.11 that the type of surface tracer used to analyze a flow has significant bearing on the quality of resolved flow structures. From the configurations considered, those containing smaller particles (i.e., “beeswax” and “small spheres”) tended to provide better representations of immediately sub-surface flow than the larger tracers. This is, of course, unsurprising, as the smaller tracers have lower Stokes number (if considered ideal particles) and will therefore respond more quickly to fluid motion.

Interestingly, the presence of small surface tracers in a flow can produce better results even if large surface tracers are present. This is observed in the combined configurations (i.e., “all spheres”, “spheres and rods”, and “all debris”), which appear to represent the flow relatively well despite also tracking large tracers.

Those flows containing only large tracers, however, represent the sub-surface PIV data less well than the rest of the cases. The “large spheres” case and the “rods” case both correspond less well to the ensemble averages than the rest of the configurations. This, too, is expected, since the larger tracers have more inertia, integrate flow motion over larger areas, and are more susceptible to inter-particle interactions. These factors are especially prominent for the case with only rods, as a rod’s motion is affected by all of the flow along its length and because they have the tendency to align parallel to other rods. Moreover, the rods were less well identified by the detector, making their results less reliable in general.

Now, considering the vorticity fields together yields a useful observation: the vortical features observed on the surface appear to be aligned with sub-surface vorticity for all surface tracer configurations. As discussed above, the representation is best for the smallest tracers—the vorticity magnitude is more similar and the features are sharper. However, even for the cases with large

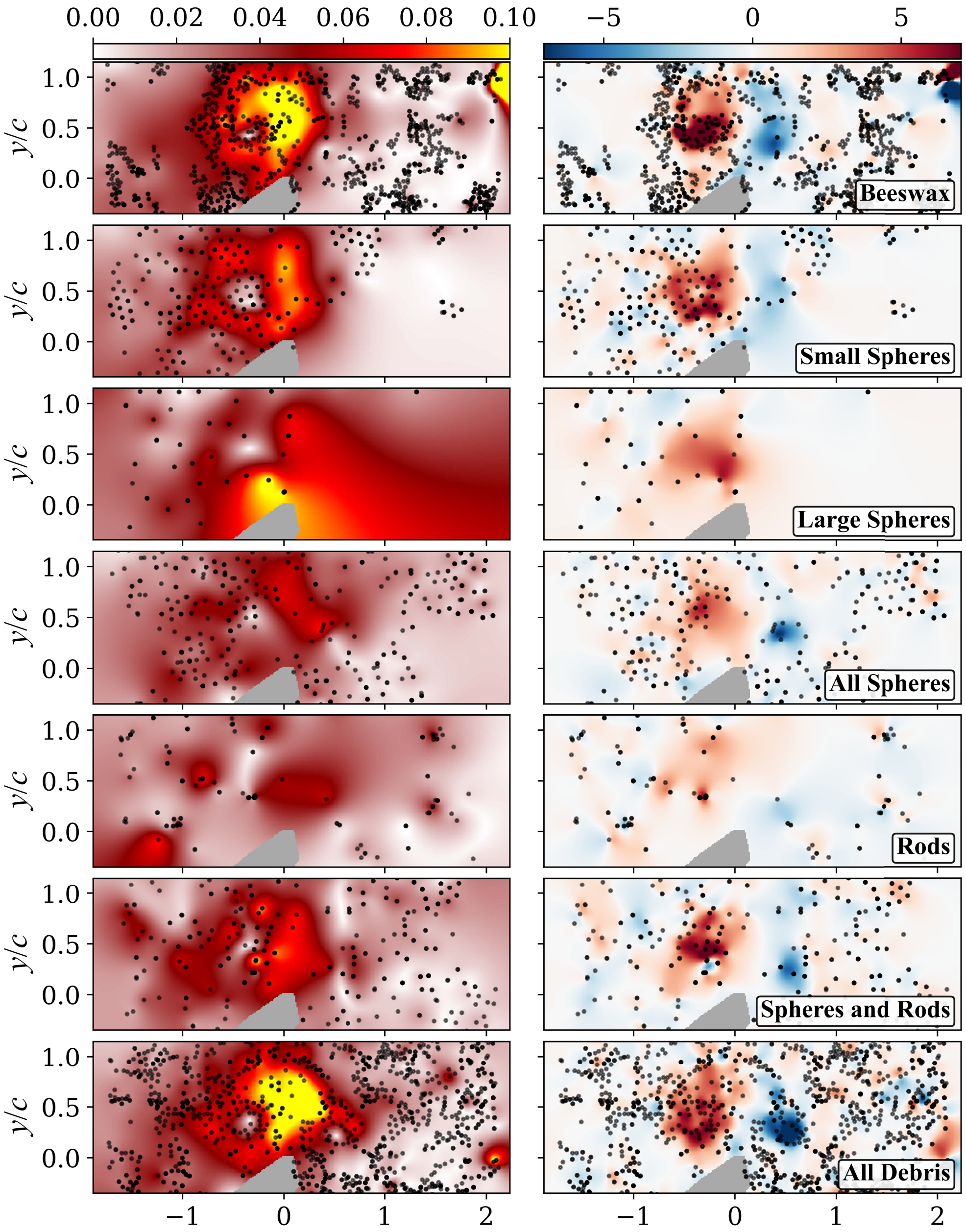


Figure 6.11: The same data as in Figure 6.10, only instead of overlaying isocontours from PIV data, tracer centroids used to interpolate the field are indicated as gray dots.

spheres and rods, the dominant vortical features are evident and appropriately situated. There are some caveats to this. For instance, in the "small spheres" and "large spheres" configurations, there are insufficient tracers in the field near the clockwise vortex shed off the trailing edge of the airfoil, and it is therefore not adequately sensed. Additionally, the magnitude of vorticity in the “rods” configuration is small across the field, and may be difficult to interpret without additional knowledge the flow.

Vorticity seemed to be robustly detectable by surface tracers, but the velocity magnitude was observed to be less reliable. Specifically, particle motion appears to be curved towards the leading edge vortex generated by the airfoil motion. While the largest velocity magnitude values in the PIV data appear aligned with the airfoil angle of attack, those in the tracer data seem to form a ring around the right-hand side of the vortex core. The “large spheres” and, especially, the “rods” configurations do not adequately approximate sub-surface velocity magnitudes.

A few factors may contribute to the deviations observed in the velocity magnitude fields. First, because the tracers are lighter than the fluid, they will be drawn towards the low pressure region at the center of the vortex (for example, see the results of [164], [179]). This will likely cause their path to deviate from the trajectories expected by PIV velocity data. Moreover, the reduced pressure causes an depression in the topology of the water surface, which may also deflect tracers. Additionally, surface tension effects will cause deviations in flow behavior. Tracers of all configurations were observed to gather into clusters if left in quiescent fluid. Both the existence of clusters and the tendency to gather will influence flow measurement ability. Moreover, the tracers are finite-sized with mass and momentum, and frequently collide with one another during the tests. Such collisions certainly affect the measurements recorded by their motion.

These considerations may also provide a rudimentary explanation for the surface tracer voids which develop later in the flow. At $t^* = 2.0$, the leading edge vortex remains coherent and strong, pulling tracers towards its center. Though the tracers have mass and momentum, the attractive force of the vortex is strong enough to overcome the centripetal force experienced by curving tracers. However, once the vortex begins to dissipate, the inertia of the particles overcomes the attraction of the vortex, forcing the surface tracers to disperse and leaving a void near the weakened core of the vortex.

It should be noted that these explanations are based on limited observations and a preliminary analysis.

6.7 Discussion of Results

This chapter set out to address whether or not kinematic behavior observed through surface tracer motion is representative of true flow patterns near the free surface. Can flow structures be reliably identified from imperfect tracers? The results displayed throughout this chapter indicate that the answer is yes. . . sometimes.

Detecting Flow Features from Imperfect Tracer Motion

It is well understood that mass, buoyancy, geometry, and other physical tracer attributes cause their motion to deviate from the pathlines traced by ideal tracers. The analysis performed in this study has little to add to the breadth of literature surrounding the subject. What is observed here affirms much of what is known about imperfect tracer motion.

In the broader context of AMFM and the identification of flow features *in situ*, the critical observation obtained from this experiment is that immediately sub-surface vorticity can be reasonably approximated by the motion of surface tracers. In a sense, this is surprising, since the velocities of the surface tracers seem to differ notably from the subsurface movement. For the example considered, it seems that the low pressure and surface depressions caused by the vortices draws particles near to their core. Thus, while the trajectory deviates from its ideal path, the vortices are still highlighted. Therefore, if one seeks to identify flow features from imperfect tracers (which will often be the case in AMFM field studies), then vortices, and metrics derived of vorticity may be more reliable than stretch-based alternatives like the FTLE.

Discerning Tracer Relevance to Flow Dynamics

While the results obtained in the analyses of this chapter indicate that imperfect tracers can be useful for structure identification, it was also apparent that certain tracer categories were more effective at elucidating structures than others. If these methods are applied to an unknown flow where little information is known about the tracer characteristics, some means of assessing tracer relevance to the measured flow must be obtained. This is briefly discussed here.

Perhaps the most direct approach to ensuring that only relevant tracers are included in analyses is to allow an informed practictioner to train the detector on curated training data which omits irrelevant tracers. This is the approach which was undertaken for all of the examples discussed in this dissertation, and can be relatively easily implemented. The training data may be collected directly from frames of the videos which are being analyzed, thereby guaranteeing that tracers relevant to the flow are identified. This approach, however, still requires some user input, and therefore makes it impractical for fully autonomous application.

If, instead, it is assumed that a general detector capable of identifying all tracers and distinguishing between their categories (e.g., from this experiment, between rods, small and large spheres, and beeswax) is implemented, then it may be possible to employ an iterative process to winnow out misleading tracers. One approach for such a procedure may be to compute gradients from the tracers varying the tracer class and processing parameters such as regression neighborhood radius. A cost metric would be defined to evaluate the legitimacy of each processing iteration. For example, if the flow is expected to be incompressible, one could seek to minimize the observed tracer divergence, although other cost functions could also be conceived. The results of such an iterative scheme would identify the most relevant tracers and optimal processing parameters given the flow.

Alternatively, if properties of the tracers are known, it may be possible to use their motion and the flow fields they espouse to regress the true flow at the free surface. An extension of this work might be to fit such operators given various tracer distributions.

Finally, when taking measurements in the field (as will be done in the next chapter), the size of surface tracers relative to the flow features of interest will likely be relatively small—one can imagine leaves or bubbles on the surface of a river, where the scale of eddies is generally much larger than the geometric scales of the tracer. In those instances, tracers may be considered closer to ideal particles, and the structures they expose more trustworthy. This assumption is supported by the results of this experiment, where the small beeswax pellets better represented the flow than other, larger tracers.

Enabling Complex Experiments by Semantic Tracking

This experiment was enabled by the semantic flow measurement approach developed in Chapter 5. Without the ability to track arbitrary tracers in PIV images, this study could not have been adequately performed. This type of analysis opens the door to a range of other experiments examining the interactions of physical bodies with complex flows.

Studying the motion of imperfect tracers in turbulent flows, for example, is an important research topic in modern times due to the prevalence of plastics and other anthropogenic debris in global waterways [168], [170], [180]. Typical studies of aspherical particle dynamics involve complex experimental setups due to the technological inability to ascertain tracer positions from common images [169], [172]. The methods employed here demonstrate that it is indeed possible to accurately track the motion of tracers without any special implementations.

Another extension of this experiment may be to comparatively examine the dynamics of tracers with identical geometry while varying other properties like density and surface roughness. One way to distinguish between particles would be to simply apply different colors to different tracers. Performing experiments such as this would build on the literature containing studies such as [82], [164].

The methods employed in this chapter are general to dynamical systems and therefore do not specifically apply to fluid flows. Thus, novel studies could be performed which examine the motion of active agents like fish, jellyfish, or brine shrimp as they navigate various flows. Moreover, the methods could be applied to complex deformable bodies such as bubbles or flags, among other possible studies of immersed objects.

Lastly, the detection and tracking methods employed in this study may also be useful for the removal of unwanted features from images of experimental flows. For example, in PIV studies, dynamic masking of complex bodies is a common challenge. Already, researchers have begun to employ segmentation models like SAM to this problem [181]. As models develop, opportunities for generalization abound.

6.8 Related Future Work

A large body of literature surrounds the motion of imperfect flow tracers in complex flows. The experiment performed in this chapter considers only one narrow aspect of the field, and even that incompletely. There is much work that can be done to expand upon this experiment and the insights gained from it. Potential avenues are presented below:

1. The results presented here principally compare the Eulerian fields es-

timated by PIV and by interpolation of Lagrangian velocities. An extension of this study would use the measured PIV fields to simulate Lagrangian tracers and compare them to what was recorded on the surface. Various tracer motion models could be incorporated. For instance, ideal tracers and Maxey-Riley tracers could both be compared with the motion observed on the surface. Information learned from such an experiment could potentially be useful in modeling efforts for surface tracer behavior.

2. This analysis consisted only in measuring instantaneous flow quantities. Extending the study to finite-time analyses including coherent structure determination may be of interest.

3. The ability to semantically measure tracers and estimate flow motion opens the door to a variety of experiments in the realm of imperfect tracer motion. An interesting follow up could be to study the motion of bubbles and sediment in complex flows and in 3 dimensions.

4. The methods in this experiment allow for localized tracer motion to be compared directly with the underlying flow. It may be interesting to measure the rotation of aspherical tracers directly in relation to the flow that they are in.

5. Finally, this experiment was designed to assess the viability of various surface tracers for estimating flow immediately sub-surface. One application where this may be advantageous is the study of polar ocean dynamics via ice floe trajectories. A valuable extension of this experiment would seek to solve the inverse problem from ice floe trajectories to sub-surface ocean dynamics.

Chapter 7

FIELD MEASUREMENTS IN CALTECH'S TURTLE PONDS

This dissertation has developed and tested an approach for calculating flow gradients from sparsely distributed tracers observed in complex images and representing the flow imperfectly. Chapters 2, 3, and 4 introduced and evaluated the Lagrangian gradient regression (LGR) algorithm, which enables trustworthy gradient approximation from sparse trajectories. To obtain these trajectories from complex images, Chapter 5 proposed a semantic object tracking scheme to identify and trace arbitrary objects in arbitrary scenes using the machinery of deep computer vision. These methods are joined together and tested experimentally on simulated debris flows in Chapters 5 and 6. Chapter 6 considered the relationship of surface tracers in the flow to the underlying currents and established that, although surface tracers do not perfectly represent sub-surface flows, the connection is strong enough to use surface tracers for flow analysis when other information is not available. This chapter synthesizes all of these results to explore fluid motion observed in the field based solely off of naturally occurring tracers and using affordable hardware.

In this particular instance, the field is not far from the lab. The experiment under consideration was performed at the Caltech turtle ponds fifty feet from the entrance to the GALCIT laboratory facility. The turtle ponds are a familiar and beloved landscaping feature where visitors and members of the caltech community alike can frequently be found enjoying the natural serenity. Indeed, many afternoons during the production of this work were spent reflecting in the surrounding garden. Small streams and waterfalls drive the motion of the fluid, which is visible by the trajectories of leaves, bubbles, and other debris that are often found upon its surface. The experiment performed in this chapter films these debris and uses the methods developed throughout this dissertation to compute flow quantities from the videos. No artificial seeding was supplied—only those tracers present on the water surface at the moment of recording were required.

The contents of this chapter have been adapted from a manuscript by Harms, Brunton, and McKeon recently submitted under the title *Estimating Dynamic Flow Features in Groups of Tracked Objects*.

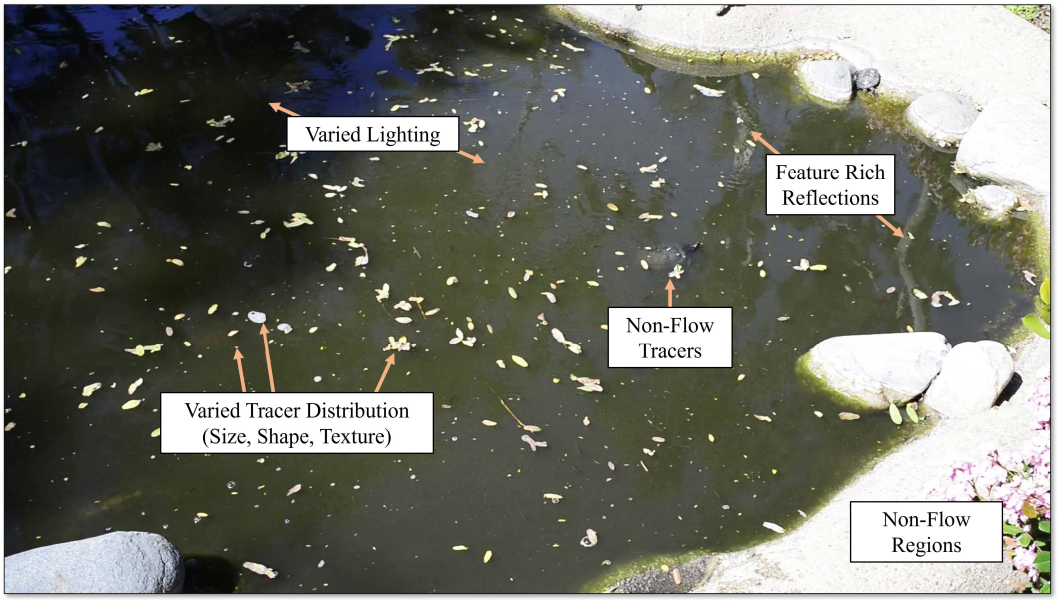


Figure 7.1: An example image from the Caltech turtle ponds. Image features which complicate normal motion and gradient estimates are highlighted.

Images recorded in the turtle ponds were exceedingly complex beyond what has been considered so far in this dissertation. They contain many features that are not relevant to the flow and garner the potential to confound kinematic measurements. As an example, consider the image taken from the Caltech turtle ponds which is presented in Figure 7.1 and annotated to emphasize the feature-rich complexity of the natural flow images. The scene being considered includes shadows and variations in lighting, multiple forms of legitimate tracers (bubbles, leaves, and other debris), illegitimate tracers (including turtles and fish), feature rich reflections from the surrounding flora which ripple in the motion of the water, and regions of the image which are not part of the flow. All of these attributes severely complicate the estimation of velocities and, especially, of spatial flow gradients. Therefore, the tools developed and tested throughout this work are necessary to compute useful approximations of the flow motion.

This chapter is organized as follows. Section 7.1 describes the experimental setup and data collection process. Section 7.2 discusses the processing pipeline used to analyze the data collected from the turtle ponds. Section 7.3 presents the results of the analysis, which includes both the detection and tracking of tracers and the computation of flow gradients from different locations in

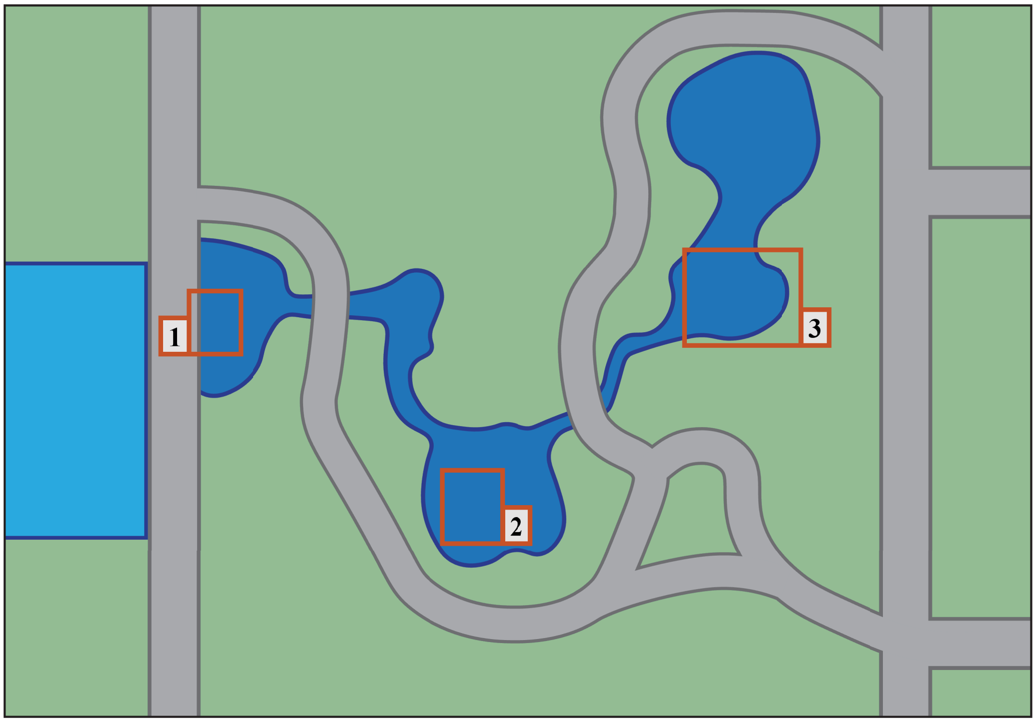


Figure 7.2: Overhead view of the Caltech turtle ponds with locations of flow recordings indicated.

the ponds. The chapter concludes with a discussion of the results and their implications for the future of flow field measurement.

7.1 Experimental Setup

Data was collected in the Caltech turtle ponds on two days and in three separate locations. The specific locations of the tests are indicated by the numbered orange boxes shown in Figure 7.2.

Tests at locations 1 and 2 were conducted together on the first day of data collection. In these tests, imaging was performed using the Teledyne/FLIR Blackfly machine vision camera described in Section B.1 of Appendix B. An Edumund Optics 4mm UC series wide angle lens was affixed to the Blackfly camera for data collected at location 1. At location 2, a 15mm Edumund Optics lens was implemented in order to better resolve the small particles in the flow. Multiple runs were collected at each experimental location, all of which were recorded at a frame rate of 100 Hz with 1920 × 1200 resolution. Because LGR tends to perform better with lower sampling rate, these videos were downsampled to 30 Hz prior to tracer tracking and LGR analysis.

Tests performed at location 3 from Figure 7.2 were unplanned, but have proven to be productive. This data was collected on a day when an abundance of natural debris had fallen into the turtle ponds, exposing the flow structures by their motion. Other responsibilities were temporarily abandoned, and the Nikon D800 and calibration boards were quickly gathered for the impromptu experiment. Data was collected through two lenses during this recording: a Nikon AF Nikkor 35mm f/2D lens and a Nikon AF Nikkor 50mm f/1.8D lens. The results displayed in this chapter were taken using the 35mm lens. In all recordings, images were sampled at the device maximum framerate of 30 FPS with 1920 × 1080 pixel resolution.

Once again, cameras were calibrated using the ChArUco method described in Appendix B Section B.2. Since the camera could not be placed directly above the flow, the homography served to orient the camera to a perpendicular position as well as to dimensionalize the frames.

Training data for the Mask-RCNN detector was collected from frames randomly sampled from all videos recorded during the experiments. Tracers identified in these images involved leaves, bubbles, bugs, and other passive debris, but omitted active tracers like turtles and fish, which were sometimes visible. Samples of these training images have been displayed in Figure B.4.

7.2 Implementation of Detection, Tracking, and LGR

The same algorithmic pipeline was applied to the turtle pond data as was implemented in the laboratory experiments of Chapters 5 and 6. For detection, the Mask-RCNN [147] detector architecture provided segmentations of legitimate tracers, from which centroids were identified. Given the training data, the detection model was trained for 150 epochs using an ADAM optimization scheme with a decreasing learning rate. Windowing was again used to improve the quality of detections. In the turtle pond images, detections were made inside 400 × 400 pixel tiles with 25% overlap.

Trajectories were tracked using the template-matching approach detailed in Appendix C. A few modifications were added to improve overall quality. First, to reduce the number of stored trajectories and to improve the average quality of tracks, only trajectories with length 5 or greater were kept. Temporal signal filtering was applied to the identified trajectories to mitigate noise that occurred as a result of the detection process. This filtering included a median filter with a 5 sample kernel length to remove outliers, followed by a Gaussian smoothing filter with a 10 sample kernel length. Following initial construction of trajectories, trajectory merging was implemented to increase the average length of trajectories and improve LGR results.

7.3 Results

Results from the turtle ponds experiments are presented in two parts: the data collected in the lower pool is first analyzed, followed by the data collected in the upper pool.

Results from the Lower Pool

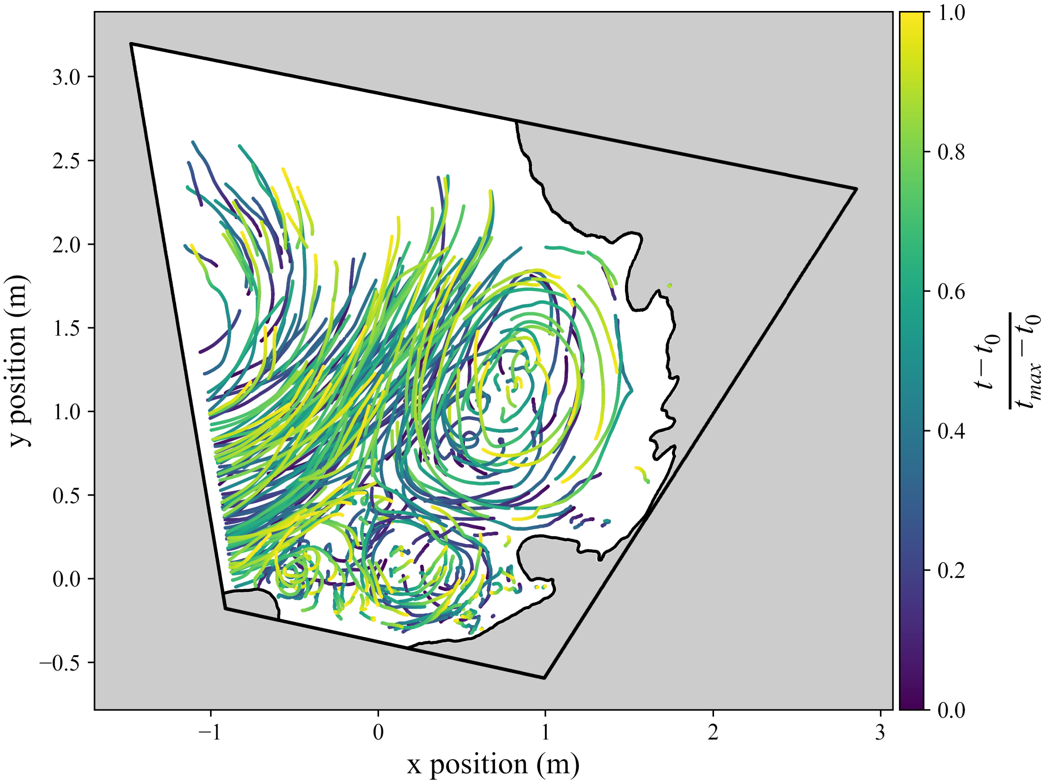
The first set of flow data collected in the Caltech turtle ponds was recorded in the lower pool as indicated by the box labeled with the number 3 in Figure 7.2 of Appendix B. The analyzed image sequence was captured using the Nikon D800 on a day when the leaves and other debris were plentiful on the surface of the water.

Figure 7.3 provides examples of the tracer detections and trajectories identified in the flow measurement process. Visualizations of all tracers detected in a sample frame of the analyzed video are displayed in Figure 7.3a, where detected tracers are indicated in color. Considering the visualized detections, it is evident that the trained model identifies few false negatives (very few of the masks shown in Figure 7.3a are not actually tracers), but more commonly makes false positives (a number of tracers are not detected). Since a particle may be captured in one frame and lost in the next, the problem of false positives significantly limits finite-time analyses (e.g., FTLE or LAVD) by causing premature truncation of trajectories.

Figure 7.3b displays the subset of identified trajectories with length of 200 or more snapshots (6 and 2/3 seconds or longer). The coloring scheme indicates the relative position in time of the trajectory instance within the recording using a nondimensional time defined as $t^* = \frac{t-t_0}{t_{max}-t_0}$, where t_0 is the time at the first frame and t_{max} is the time at the last frame. The shaded gray regions represent a mask applied to the images for plotting purposes and the flow domain outside the field of view of the camera.

By observing the plotted trajectories, the character of the flow begins to appear. Tracers enter the field of view at the bottom left hand side of the stream as driven by a small stream entering the lower pond off screen. The bulk motion of the flow gradually rotates counterclockwise with a large radius

(a) A sample image from the data set with all detected surface tracer segmentations provided with randomly generated colors.



(b) All identified trajectories with at least 200 samples (6 and 2/3 seconds) calibrated to world coordinates and colored by nondimensional time t^* within the recording. The gray shaded regions indicate parts of the flow that are either masked out for visualization purposes or external to the field of view.

Figure 7.3: Tracer detection and tracking results from data collected in the lower pool of the Caltech turtle ponds.

of curvature. However, the geometry of the pond causes eddies to be formed in the gulf between the bulk fluid rotation and the pond shore at right. The largest of these eddies occurs in the upper right hand portion of the image, with smaller ones closer to the bottom of the image. As the coloring indicates, these eddies remain largely stationary throughout the approximately 64 second duration of the flow. The trajectories alone, however, do not provide quantitative information regarding the flow behavior; to achieve this, LGR is now implemented and relevant metrics are computed.

Flow gradients were estimated using LGR with radial Gaussian kernel weighting. For each tracer at each time step, the 25 nearest neighbors that persisted from the current frame to the next were used to regress the deformation operator by equation 3.11. The weighting was defined using equation 3.12, where the standard deviation was set to $s = 0.6$ meters. Velocity gradients were estimated according to equation 3.13. Finite-time analyses were computed over an 8 second interval by computing the 8-second flow map Jacobian by composition through equation 3.16.

Using the approximated gradients, the metrics discussed throughout this work are computed for the flow in the lower pool of the turtle ponds. Figure 7.4 displays the results of eight metrics that have been introduced earlier in this thesis. Figures 7.4a and 7.4b display the vorticity and vorticity deviation, which are both instantaneous metrics for rotation. Figures 7.4c and 7.4d, respectively, display the finite-time stretching metrics FTLE and TISM. Figures 7.4e and 7.4f display LAVD and DRA as time-integrated measures of rotation. Finally, Figures 7.4g and 7.4h display the Q-criterion and the kinematic action as metrics that segment the flow into domains of stretching and domains of rotation for instantaneous and finite-time analyses, respectively. All instantaneous metrics are computed at the nondimensionalized time $t^* = 0.29$ and the finite-time metrics are computed over the interval $t^* = [0.29, 0.41]$.

The instantaneous metrics presented in Figures 7.4a, 7.4b, and 7.4g are first discussed. Examining the vorticity presented in Figure 7.4a, the four vortices identified from the visualized trajectories begin to materialize. It seems that there are approximately two counter-clockwise vortices (one in the bulk flow at the upper left-hand side of the figure and one on the bottom right hand side by the edge of the pool) and two clockwise vortices (on the lower left and upper right hand portions of the flow). While analyzing the vorticity is useful with a stationary observer, it is not an objective metric (see Chapter 2 and Appendix

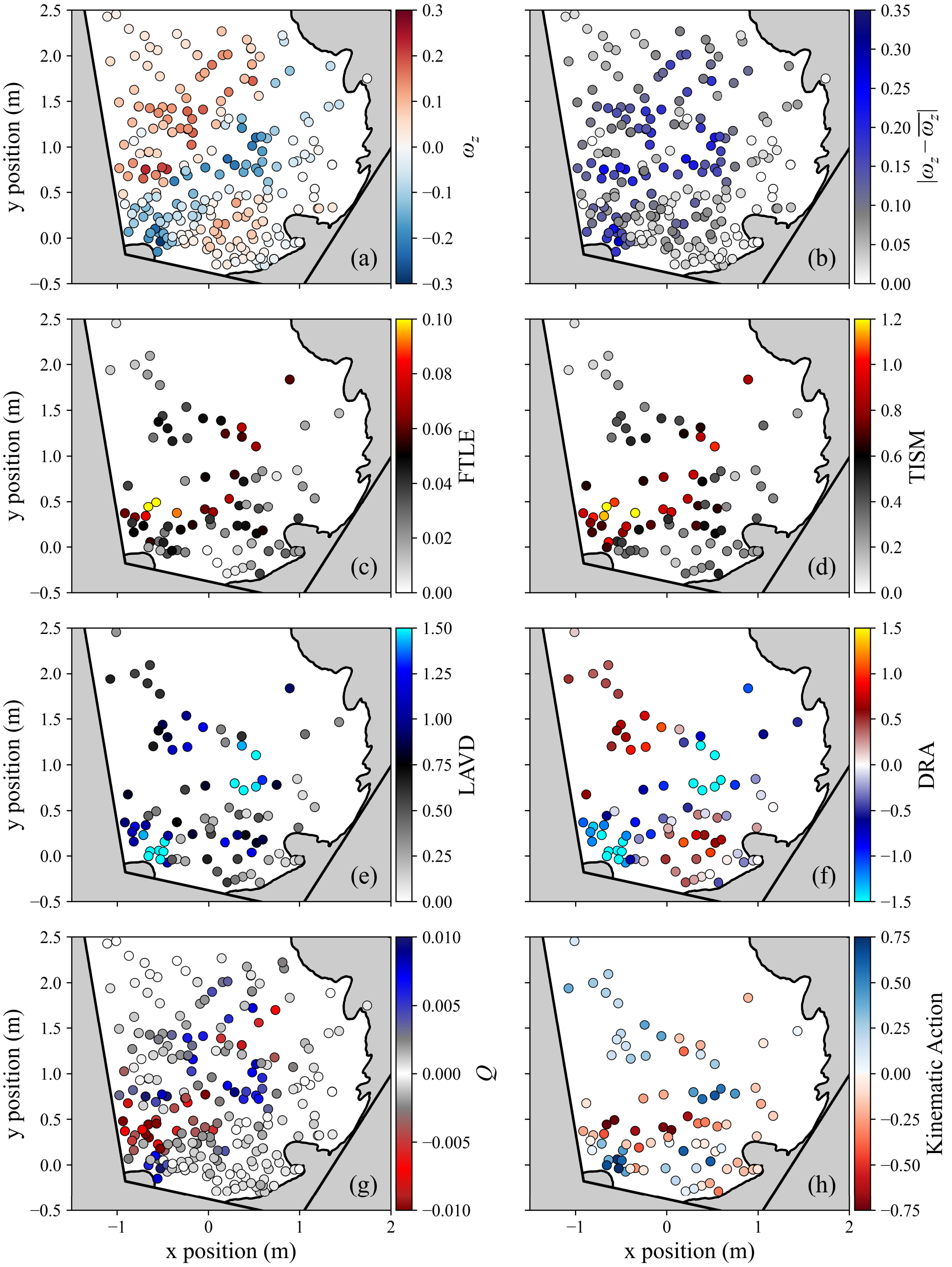


Figure 7.4: Gradient-based metrics computed via LGR from the flow tracers observed in the lower pool of the Caltech turtle ponds.

A). Vorticity deviation, indicated in Figure 7.4b, is objective, however. Its values highlight the same vortical features present in the vorticity data, albeit without sign information. Finally, the Q-criterion in Figure 7.4g is considered. The dark red regions represent portions of the flow that are dominated by tracer separation and the blue regions represent rotation dominated regions. As new particles enter into the observed domain on the lower left hand side of the frame, they are either carried off by the bulk rotation into the upper portions of the field of view or they are caught in the eddies close to the pond shore. This causes particles to separate rapidly and leads to large negative values of Q. On the other hand, tracers caught inside the vortices do not separate as much, and therefore exhibit large positive values of Q, displayed in blue.

The finite-time analyses indicated in Figures 7.4c–7.4f and 7.4h help to complete the picture of the flow measured in the lower pool of the turtle ponds. The FTLE and TISM results of Figures 7.4c and 7.4d highlight a ridge that runs from the bottom left of the frame to the upper right along the perimeter of the rotation observed by the vorticity in the bulk flow of the pond. This ridge is slightly more evident in the TISM than in the FTLE, and represents a material surface that separates tracers which continue in the bulk of the flow from those that are drawn off into the eddies by the shore. The LAVD and DRA from Figures 7.4e and 7.4f, respectively, support this notion by highlighting two strong counterclockwise vortices adjacent to the separatrix indicated by FTLE and TISM. From LAVD and DRA, the clockwise bulk flow and a smaller clockwise vortex near the shore are also evident. The kinematic action plotted in Figure 7.4h highlights all four vortices and the separatrices between them at once. In this figure, the red tracers indicate regions of the flow which are dominated by stretching over the 8 second integration time, and the blue tracers indicate regions dominated by rotation. Intuitively, there should be a region of stretching in between every vortex, which is clearly observed in the kinematic action data.

Considering together all eight metrics presented in Figure 7.4 paints a clear picture of the behavior of the observed flow field. The flow in this section of the lower pool of the turtle ponds is driven by a large, slow rotation in the body of the pool, which is the vortex observed in the upper left of each frame. This rotation, however, is hindered by the geometry of the shore of the pond, and therefore generates a shear layer between the water in the gulf of the pond and the water in the body of it. This shear layer is highlighted by the dominant FTLE/TISM ridge that moves diagonally across the flow and the diagonal swath of negative vorticity. Vortices form along the shear layer and accumulate into the stronger rotations observed by the LAVD and the negative DRA data. To maintain conservation of mass and momentum, the smaller counter-clockwise vortex in the bottom right hand corner of the frame is formed.

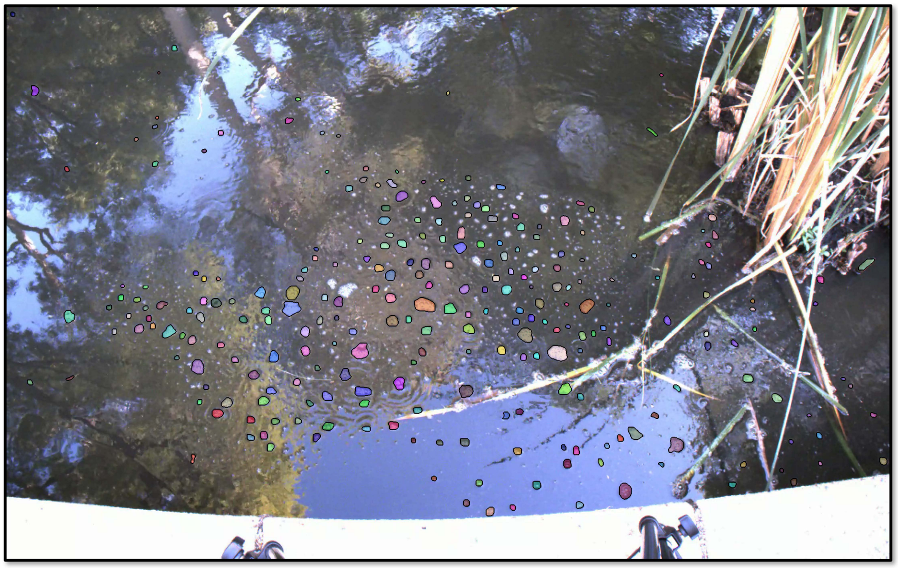
Critically, the synthesis of all of the observed metrics contributed to this analysis. The instantaneous metrics are able to highlight the fast motions of the flow and give a picture for their broader spatial influence, but the finite-time metrics elucidate which structures in the flow persist over time. Some of the smaller vortical features observed in the vorticity and vorticity deviation are not apparent in the LAVD or DRA, suggesting that they merge with other flow features over the 8 second integration window. By considering all of these metrics together, a more complete understanding of the flow is attained.

Results from the Upper Pool

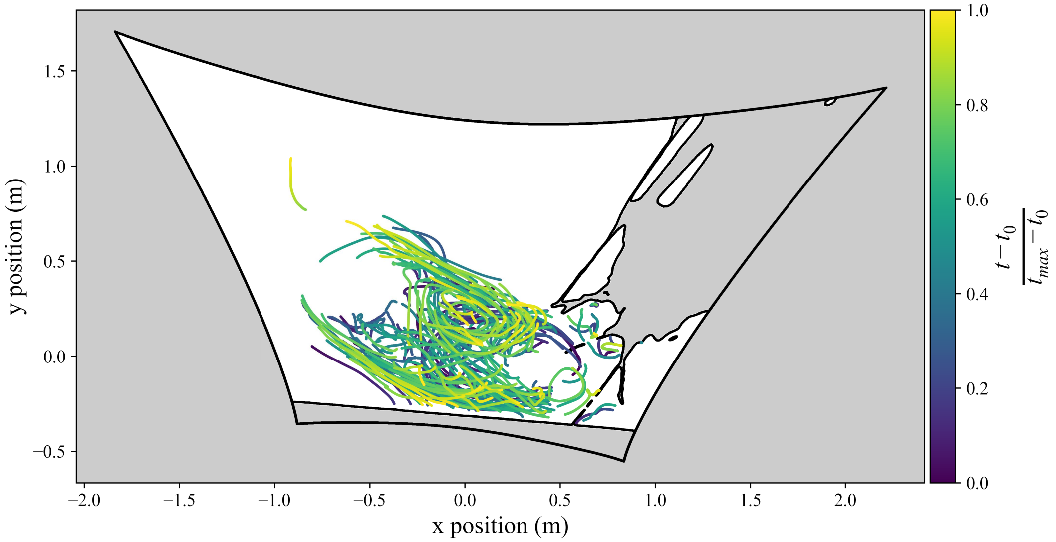
The second data set reported upon form the Caltech turtle ponds was recorded in the upper pool where an underwater jet aerates the water and drives fluid motion at the surface. The location where the footage was captured is indicated in Figure 7.2 of Appendix B by the box labeled with the number 1. Where the tracers available in the example shown from the lower pool were largely solid debris like leaves and sticks, the observable debris in the upper pool almost exclusively consisted of clusters of bubbles that gradually disappeared and often joined together or separated.

Images in the upper pool were collected using the Blackfly machine vision camera with a 4mm wide angle lens for a duration of 49 seconds. Due to the wide angle, significant radial distortion was present in the images and corrected through calibration. Detection in the upper pool was performed using the same detection model as the previous example on 400 × 400 pixel windows tiled throughout the image with 25% overlap. Tracking was implemented using the template matching approach discussed above. Calibration to world coordinates was applied to the mapped trajectories prior to LGR computation.

Sample results from the detection and tracking process in the upper pool are displayed in Figure 7.5. Sample tracers, which are mostly located centrally in the image frame, are indicated by colored masks overlaid on top of the original image in Figure 7.5a. A variety of complicating features are present in the scene of the upper pool data. For instance, the reeds which have fallen into the water do not respond as quickly to the currents, and can act as artificial barriers to the bubbles in the flow. Moreover, bubbles tend to accumulate



(a) A sample image from the data set with all detected surface tracer segmentations provided with randomly generated colors.



(b) All identified trajectories with at least 200 samples (6 and 2/3 seconds) calibrated to world coordinates and colored by nondimensional time t^* within the recording. The gray shaded regions indicate parts of the flow that are either masked out for visualization purposes or external to the field of view.

Figure 7.5: Tracer detection and tracking results from data collected in the upper pool of the Caltech turtle ponds.

onto the reeds or onto the patch of ground where reeds are growing. These bubbles are sometimes identified as tracers and can cause spurious results. Finally, based on the model parameters used, some of the legitimate tracers in the flow are not identified by the detector. However, as with the previous example, there seem to be many more false negatives than false positives, which is favorable.

All identified trajectories longer than 200 samples (again, 6 and 2/3 seconds) are displayed in Figure 7.5b with coloring representative of nondimensional time t^*. From the trajectories, a general counterclockwise rotation seems to be evident in the center of the frame. It also appears that smaller vortical features exist in the flow near the reeds towards the right hand side of the image. The trajectories in this example, however, are less illustrative than in the previous case. This flow evolves more rapidly, with features developing and dissipating within the 49 seconds of observation. The structure of the flow is more evident when considering the results of LGR.

Lagrangian gradient regression was performed along each trajectory using the 30 nearest neighbors of each detected particle. A radial Gaussian kernel was applied according to equation 3.12 with a standard deviation of $s = 0.15$ meters to ensure that the particles nearest to the trajectory were preferentially weighted in the computations. A small regularization constant of $\gamma = 10^{-6}$ was applied for numerical stability. Velocity gradients were estimated by equation 3.13 and flow map Jacobians by equation 3.16 over a 1.5 second interval.

As before, the character of the flow becomes more apparent when considering the various gradient based metrics together. Figure 7.6 displays instantaneous metrics based off of the velocity gradient at nondimensional time $t^* = 0.42$ and finite-time metrics at the same time computed over the interval $t^* = [0.42, 0.47]$.

This flow differs from that in the lower pool in that the spatial mean vorticity is not close to zero during a large duration of the recording. As seen in the vorticity of Figure 7.6a, The tracers exhibit a collective counterclockwise rotation on average over the observed tracers. When this mean is subtracted from the vorticity of each individual tracer and plotted in Figure 7.6b, some of the evidence of rotation is lost—for example, the counterclockwise rotation in the flow near the bottom right-hand corner of the displayed region.

Aside from considering the mean rotation, there seem to be three vortices expressed in the flow which dominate the motion at the observed snapshot.

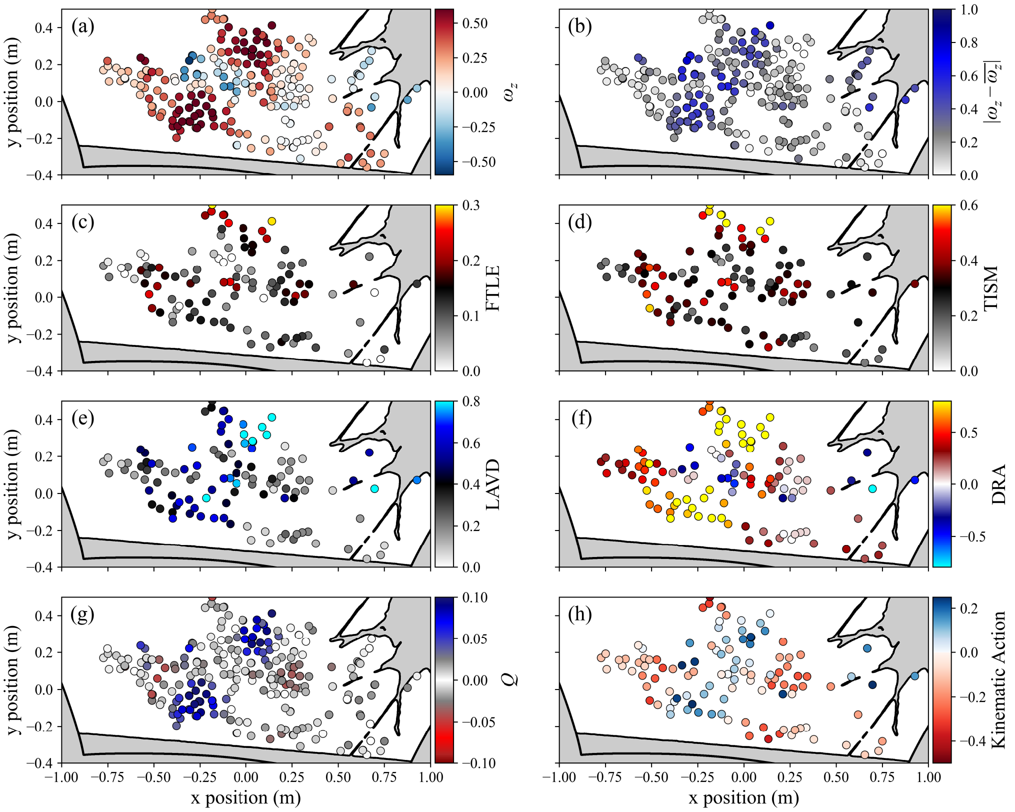


Figure 7.6: Gradient-based metrics computed via LGR from the flow tracers observed in the upper pool of the Caltech turtle ponds.

These are clearly visible in the data presented with the vorticity, vorticity deviation, LAVD, DRA, Q-criterion, and kinematic action of Figures 7.6a, 7.6b, 7.6e, 7.6f, 7.6g, and 7.6h, respectively. Two large, counterclockwise vortices seem to dominate the dynamics of the flow, but a third, clockwise vortex is formed in between them, which seems to persist over the finite time integration interval.

The FTLE, TISM, Q-criterion and kinematic action of Figures 7.6c, 7.6d, 7.6g, and 7.6h indicate which regions of the flow are dominated by stretching. As before, there is some evidence of lines of stretching in between the vortices observed in the flow. The counter rotation of the vortices forces tracers to separate as they are drawn into their respective vortices. At the present snapshot, there is also evidence that tracers on the periphery of the flow (i.e., those at left of the main cluster of tracers on both the top and the bottom of the image) are moved by the flow away from the rest of the tracers into the body of the pond. The evidence for this lies in the FTLE and TISM representations of the data where a ridge is apparent at the bottom left and a region of large values at the top center.

7.4 Discussion of Results

In this chapter, the flow tracer tracking and detection was applied in concert with LGR to analyze fluid motion in the Caltech turtle ponds using only the tracers naturally available on the pond surface. Results have been presented for two representative cases of the collected data. Here, the implications of the results are discussed.

Affordable Kinematic Flow Analysis Using Only Natural Tracers

The results presented in this chapter were all obtained using affordable, readily available hardware and modest computational resources. Recordings from the lower pond were taken using a hobbyist DSLR camera which, at the time of this writing, can be purchased with a lens for less than $2000 USD. The recordings from the upper pool were captured with an optical setup that cost less than $1000 USD. If someone desired to do so, they could capture videos with their cell phone and a tripod and achieve similar results. In terms of processing, Python and freely available packages were used for all of the analysis in this experiment. The detection models were trained on a desktop computer with single mid-grade GPU, and all other algorithms could be run on a standard CPU.

The accessibility of the materials required for these experiments marks a step towards the democratization of scientific analysis of fluid dynamics. Anyone who is interested in examining some fluid flow can choose to do so using only equipment that they likely already own. Hobbyists could perform experiments in their own back yards; researchers at institutions with fewer resources could expand their experimental capacity; practitioners in remote areas could examine difficult-to-access flows and leverage their gained understanding to engineer around them.

Additionally, since no additional tracers were added to the flow, these methods can be applied in an environmentally friendly manner. No dyes or other artificial additives need to be added to the flow if the conditions permit.

Semantic Flow Measurements Require Little Additional Processing

Measuring flow fields—especially when using natural images—typically requires significant pre- and post-processing. The data needs to be molded into an acceptable format prior to velocimetry and heavily cleaned afterwards. However, using the methods developed by this thesis and implemented in this chapter, less additional processing is required for the analysis to be successful. With semantic flow measurements, pre-processing is unnecessary and can even be a detriment to the analysis; if the typical features of a tracer are removed, then it may not be accurately detected. Post processing can still be used to clean up trajectories and computed metrics, but these are not strictly necessary to acquire functional results.

Flow Understanding is Enhanced by Considering Many Metrics

In the results presented in this chapter, all eight gradient-based metrics discussed and analyzed in this dissertation were applied to the flows. While each metric did, independently, provide information about the character of the flows, it was their synthesis—the consideration of all of them together—which yielded the most insight. The understanding gained by considering the instantaneous metrics (vorticity, vorticity deviation, and Q-criterion) was unique to that gained from the finite-time metrics (FTLE, LAVD, DRA, and KA). The metrics highlighting rotation complemented those highlighting stretch. Multiple analyses should be performed on the same flows to optimally understand their behavior. LGR provides the means to perform all of these analyses and more using the same set of trajectories with minimal additional effort.

Limitations

There are, of course, limitations when analyzing natural flows using only natural tracers. One significant limitation is the availability of natural tracers in the flow being studied. On many occasions in the turtle ponds, there are no tracers on the surface of the water, so no measurements can be made. Artificial tracers could be added to enable or enhance measurements, but this adds complexity and can become ethically questionable depending on what the added tracers consist of.

Even if some tracers are visible on the surface of the flow, the flow structures that they reveal are limited by the spacing of the tracers on the surface of the flow. If the tracers are spread far apart, then only large flow features will be discernable. Moreover, if tracers are gathered into clusters, only local features to the clusters will be identifiable.

Finally, processing speed remains a limitation of this approach. Ultimately, the desire is to embed these algorithms onto mobile measurement platforms such as drones so that autonomous mobile measurements of the flow fields can be collected. However, the processing speed still prevents this. Data must still be offloaded to a separate device for processing. As has been mentioned in previous chapters, improved models and algorithmic optimization must be sought before AMFM is possible by these methods.

7.5 Related Directions for Future Work

The continued development of the methods employed in this chapter could take a variety of forms.

1. While the turtle ponds technically constitute *the field*, they are literally as close to the laboratory as possible while still being in *the field*. Bringing these tools to other flows beyond Caltech's campus is of interest to the author, and would be beneficial from the standpoint of methodological development.

2. Though it has already been mentioned in previous chapters, it bears repeating that these algorithms need to be optimized and faster models need to be architected or selected. Without achieving these goals AMFM measurements remain grounded.

3. Even if the measurements cannot be made autonomous in the near future, recording data from an aerial drone perspective would still be an interesting and valuable scientific contribution. The drone mounted measurement scenario marks one of the primary use-cases of this technology since adding tracers or carefully modifying measurement parameters is not possible with remote measurements. In line with this avenue for future work, the ability to calibrate remotely would also represent a valuable contribution to the technology.

4. Finally, if the improvements discussed above are implemented, it may one day be possible to begin making measurements of flow fields autonomously. For example, the methods demonstrated in this Chapter are relevant to studying the transport and behavior of debris caught in

flash floods, tsunamis, and other environmental flows that are currently difficult to measure with existing technologies.

Chapter 8

CONCLUSIONS

> *There is no end to the writing of books*
>
> — Qoheleth, Ecclesiastes

The motivating thrust of this dissertation has been to expand the range of measurable flows beyond current experimental capabilities. This objective has been largely pursued under the overarching premise of developing strategies for autonomous mobile flow field measurements (AMFM), which aims to observe large-scale flows by autonomously following critical flow structures as they evolve. If fully developed, it could enable enhanced examination of numerous flows relevant to modern applications including monitoring and mitigation of pollution in waterways, optimization of large-scale wind farm installments, and remote measurement of environmental flows. While AMFM remains a technology of the future, this work has progressed towards its realization by developing algorithms and strategies for detecting flow structures under the similar technological constraints. Lagrangian gradient regression was developed to approximate gradients from sparse observations, semantic measurements of flow kinematics were conceived as a means for dealing with natural images and tracers, and these methods were applied to novel experiments and complex field measurements. Below, a summary of the dissertation is provided, followed by principal contributions and opportunities for continued inquiry.

8.1 Summary of the Dissertation

Three challenges facing AMFM measurements were outlined in the introduction to this dissertation. The first of these was the challenge of relative motion, which requires that detection methods be able to identify the same features regardless of observer dynamics. The second was the challenge of feature-rich images, which requires that methods handle real images with all of their natural complexity. The final challenge was that of imperfect tracers, which recognizes that observed tracer motion may not resemble the true motion of the fluid. The chapters of this work attempted to hurdle these three obstacles.

In Chapter 2, the theory of Lagrangian coherent structures (LCS) was introduced as a means for dealing with observer motion relative to the flow. Flow analysis metrics from the field of LCS are typically designed to be objective, in that they are invariant to Euclidean frame transformations. They accomplish this by approximating spatial flow gradients from measurements of tracer trajectories. As clouds of tracers evolve in the flow, their deformation can be leveraged to reveal features like vortices, boundaries to fluid transport, and more. If deformations are considered over extended intervals, finite-time analyses can be constructed from the flow map Jacobian (such as the finite-time Lyapunov exponent (FTLE)) or through integration (such as Lagrangian-averaged vorticity deviation (LAVD)). Over infinitesimal deformations, the velocity gradient might be computed and analyzed through vorticity, principal strain rate, or Q-criterion.

A severe limitation of conventional gradient-based LCS analyses is the demand for high tracer density in calculations. Over extended intervals, differential deformations of finite-sized tracer clouds accumulate, transforming the fluid into complex geometries that are not well represented by the linear mapping described by the flow map Jacobian. As a result, the typical process for identifying objective gradient-based LCS features involves numerically integrating a large number of numerical tracers defined on a uniform grid over the analysis interval and computing deformation gradients by finite differences. This process can quickly become computationally expensive, and is therefore not viable for applications like AMFM.

Chapter 3 proposes a solution to the demand of dense numerical tracers in the form of Lagrangian gradient regression (LGR). LGR leverages the nature of real, measurable tracer data to overcome the need for spatial proximity. Real trajectory data can almost always be sampled rapidly in time, but sparsely in space. To accommodate this, LGR uses the data-driven machinery of regression to compute deformation gradients over short intervals (shorter than the fastest movements of the flow) and synthesizes them to arbitrary finite intervals by asserting the smoothness trajectories. In this way, sparse trajectory data sampled quickly in time can provide finite-time flow gradients that are as accurate as those which are sampled with high spatial density. This format of estimation is often preferred for practical measurements, as most data collection modalities allow for high resolution collection of data in time but not in space.

By regressing deformation gradients over short intervals, LGR also provides a framework for estimating instantaneous velocity gradients from the same sparse data to acquire flow map Jacobians. The details of this process are provided in Chapter 3. Because the velocity gradients are computed at the position of the tracers in the flow, they can be conveniently integrated over time to compute other finite-time metrics like LAVD. Chapter 3 concludes by introducing two new integrated metrics: the trajectory-integrated stretching magnitude (TISM) and the kinematic action (KA), both of which relate theoretically to LAVD and highlight different aspects of the flow's kinematic behavior.

Having introduced a novel approach to gradient estimation and structure determination in Chapter 3, Chapter 4 sets out to evaluate their efficacy relative to other similar approaches. First, LGR's capacity for gradient estimation is considered. After demonstrating that it replicates baseline gradients on numerically simulated data, LGR is tested against typical gradient estimation schemes from experimental fluids. LGR-estimated gradients are computed against those using particle image velocimetry (PIV) and Lagrangian particle tracking (LPT), and are seen to achieve comparable, and even improved, accuracy. This analysis is followed by a discussion of LGR's sensitivity to scales of motion, where it is observed that particle spacing acts as a low-pass filter removing scales of motion smaller than the distance between particles.

Chapter 4 also evaluates LGR's performance on the finite-time metrics introduced in earlier chapters. Here, it is observed that LGR yields significantly improved accuracy when computing flow map Jacobians and FTLE fields on sparse data over other methods from the literature. LGR's performance on sparse data is also explored through the use of integrated metrics. In these analyses, measures of rotation (i.e., elliptic LCS) are seen to exhibit increased robustness to sparsity over measures of dispersion (i.e., hyperbolic LCS). The KA was observed to show promise towards AMFM feature detection objectives, since it provides a clear threshold for assigning membership to finite-time vortices.

Chapters 2, 3, and 4 all serve to address the challenge of relative motion. Chapters 5 and 6 builds upon these results to attack the remaining challenges: those of feature-rich images and of imperfect tracers.

LGR is only useful for flow feature identification if trajectories can be identified in recorded image sequences. In LPT studies, this is achieved by carefully controlling experimental conditions and artificially seeding and illuminating passive tracers. In the field, however, it is unlikely that tracer visibility and scene complexity can be controlled. Therefore, dealing with feature-rich images means tracking the trajectories of arbitrary tracers in sequences of arbitrary images.

To overcome this challenge, a semantic approach to kinematic flow measurements was developed in Chapter 5 using modern advances in object detection and tracking. A modular algorithmic approach was proposed which involved detecting tracers using a deep detection model fine-tuned on training samples, tracking motion between frames using methods from LPT and multiple object tracking (MOT), and estimating flow kinematics using LGR. These methods were tested in the NOAH laboratory at GALCIT using wooden debris and affordable imaging hardware.

Even though the methods of Chapter 5 effectively compute flow structures from unconventional surface tracers, the relationship of the tracers to the flow is not necessarily clear. The particles contain many complicating attributes such as buoyancy, mass, and aspherical geometry which bring into question the relevance of estimated flow kinematics to underlying fluid behavior. Chapter 6 approaches this question experimentally. Using a simple test flow, the semantic flow measurement approach developed in Chapter 5 is applied to surface tracers simultaneously with PIV measurements made beneath the surface. The motion observed from both modalities are compared with one another, and it is concluded that, under many reasonable circumstances, surface tracers can legitimately be considered flow tracers.

Finally, Chapter 7 combines the results obtained in the previous chapters to analyze field data collected in the Caltech turtle ponds using only naturally available tracers. The motion of leaves and bubbles were examined to identify flow structures through the synthesis of the tools developed and tested throughout the preceding chapters. Such videos represent the kinds of data which may one day feed into AMFM algorithms.

8.2 Principal Contributions

The principal contributions of this work fall into three categories: Lagrangian gradient regression, semantic measurement of flow kinematics, and novel applications of the developed technologies. Here these contributions are discussed.

Lagrangian Gradient Regression

The principal theoretical contribution of this work comes in the form of Lagrangian gradient regression (LGR). which was developed as an engine for gradient-based analysis of sparsely populated flows. Typically, estimating deformation gradients (flow map Jacobians, equivalently) requires observing the deformation of infinitesimally proximal tracers over extended durations. Approximating the gradient amounts to fitting the deformation operator to first order accuracy—a linearization of the deformation in space. As a result, if the initial separation of tracers is not very small, then the nonlinear influence of the flow on their deformation over time becomes large. Thus, deformation gradients are typically approximated from gridded numerical tracers seeded on top of pre-computed Eulerian flow fields.

LGR shifts the paradigm for approximating deformations by linearizing in time rather than in space. Tracer deformations over very short intervals can reasonably be approximated as linear even if the initial spacing of the tracers is relatively large. This is due to the continuity and smoothness of trajectories. Therefore, deformation gradients computed from sparse trajectories over short times faithfully represent the true deformation experienced by the flow in that time. LGR leverages this attribute by approximating deformation gradients over short intervals and resampling tracers within a neighborhood in between. It accommodates real data by estimating gradients via regression rather than by finite-differences. A kernel weighting function is applied to the regression to tune the results based on the flow, and arbitrary intervals are analyzed by compositing gradients approximated from adjacent intervals.

Furthermore, by exploiting the connection between the flow maps and flow velocity fields, LGR provides a natural extension to computing velocity gradients. In experimental fluids, velocity gradients are almost exclusively computed from interpolated velocity fields. The framework of LGR, however, is entirely trajectory based. There is no need to ever explicitly compute velocities or perform numerical differentiation. Access to velocity gradients along a trajectory also enables many finite-time integrated metrics such as LAVD, DRA, and the trajectory-integrated stretching magnitude (TISM) and kinematic action (KA) developed in Chapter 3.

The nature of LGR as discussed above affords a variety of advantages over other approaches. These advantages, which have been justified in the body of the dissertation, are highlighted below:

Unified Approach to Velocity and Deformation Gradients: Using LGR, estimating velocity gradients and deformation gradients naturally occurs in the same set of operations using the same set of trajectory data. One need not integrate numerical tracers on Eulerian velocity fields to compute deformations, nor must one interpolate to a field from Lagrangian tracers to estimate velocity gradients. All of the calculations are naturally strung together along the course of naturally observed trajectories.

Velocity- and Derivative-Free Velocity Gradients: Because of the need for interpolation (if LPT) and subsequent differentiation, estimating velocity gradients from experimental fluids data can be time consuming and may amplify experimental error, depending on the quality of the data. By relying on regression of deformations as its numerical engine, LGR avoids computing velocity—either on a trajectory or in a field—and differentiation. It is a fully Lagrangian approach to gradient approximation.

Superior Performance on Sparse Data: When computing deformation-based quantities like the FTLE on sparse data, LGR vastly out-performs other gradient estimation approaches from the literature. When applied to velocity gradients, LGR performs at least comparably well, and in many cases with improved accuracy.

Computationally Affordable Alternative to PIV/LPT: Because LGR is fully Lagrangian, many steps in the gradient estimation pipeline can be skipped in its application. For instance, there is no need to separately interpolate or differentiate when using LGR as opposed to LPT. Moreover, the required data storage can be much smaller than for studies where Eulerian fields are required, as the number of tracers (where all of the data is stored in LGR) is often much smaller than the number of grid locations (where data is stored for Eulerian fields). This can be especially beneficial for large 3D experiments.

Semantic Measurement of Flow Kinematics

The principal algorithmic contribution of this work has been the development and implementation of a semantic formulation for kinematic flow measurements. Virtually all analyses of flow kinematics are syntactic, in the sense that they base their computations on the structure of data (images, in this case) rather than on the meaning (semantics) of features in the data. Operations such as PIV and optical flow (OF) are examples of syntactic analyses that can be very effective if experimental conditions are favorable. Thus, in controlled laboratory flows, PIV, LPT, and OF can be efficient and effective means of measuring flow kinematics.

However, syntactic analysis of flow kinematics is easily performed in general. Natural images containing natural tracers are wrought with complexity of many forms. Reflections, non-flow regions, varied lighting, multifarious legitimate tracers, and flow-independent tracers are all examples of complicating features that may be present in natural images. The developed solution to overcoming this challenge is to train a powerful detector to semantically identify only the objects in an image which represent a part of the flow. Such tools have been developed and deployed to great success in general across disciplines; this work incorporates them into kinematic flow measurements. Since powerful open-source models have been and are being trained for general detection purposes, fine-tuning them for effect in a specific flow requires relatively little effort.

If tracers can be semantically identified with confidence, then object tracking algorithms from the world of LPT or multiple object tracking (MOT) can be implemented to string detections together into trajectories. These trajectories form the basis for kinematic analysis which is now possible due to the sparsity robustness of LGR. Thus, the semantic flow tracer detections integrate smoothly with LGR and the results discussed earlier in the dissertation.

This semantic approach to flow kinematics is enabling for a variety of reasons which are outlined below:

Selectively Identifies Tracers of Interest: Modern semantic detectors can be made to identify over 1000 classes simultaneously within images. Extending this capability to the realm of flow measurement suggests that studies could be conducted which simultaneously and independently measure flows espoused by various classes of objects. Extending the example of Chapters 5 and 6, this might mean tracking rods and spheres separately. It might also mean simultaneously measuring the kinematics of blood cells with bacteria or predators with prey. Many opportunities for novel experiments exist under this framework.

Analyzes Complex Scenes with Minimal Additional Processing: When measuring a flow in this manner, little additional processing of images is required. For instance, PIV experiments often involve heavy pre-processing to put images into a convenient format for velocimetry. Such pre-processing removes the generality of the algorithm, meaning that it must be carefully tuned to a given experimental flow. The semantic approach, however, thrives on the complex features which need to be eliminated in typical laboratory studies. This is why, in Chapter 7, the same processing parameters could be applied in both pools, even though the tracers and scene varied significantly between both data sets.

Enables Affordable Experimentation in Fluids: This semantic approach to kinematic measurements may also provide an opportunity for the democratization and popular adoption of flow field measurements. The experiments performed in Chapter 7 were conducted with hardware costing less than $1000 USD. Videos could easily have been captured on a personal mobile device. No special equipment was required in the experimental setup or in the processing of the data. It is not unreasonable to imagine that a self-contained mobile app which calculates real kinematic flow quantities could be made available to general users in the near future. The accessibility of this technology may provide admittance into the beauties of fluid mechanics for many who may never have had an opportunity to experience them otherwise.

Applications to Laboratory and Field Measurements

The developments made in the early portion of this dissertation enabled experiments which could not have otherwise been performed. The first experiment, reported on in Chapter 6, systematically compared measurements of surface tracer dynamics to fluid motions just beneath the surface. Using the same set of images, both PIV and semantic flow tracing were computed. While the results of the experiment were useful for validating surface tracer tracking for flow measurements, the experiment itself is one that would be very difficult to perform using conventional techniques. The ability to perform semantic tracer tracking allows for direct comparison of arbitrary surface tracer motion to experimentally measured flow motion.

The second experiment, conducted in the Caltech turtle ponds and discussed in Chapter 7, computed instantaneous and finite-time flow gradients solely from naturally occurring surface tracers. The measured medium consisted of leaves, twigs, bubbles, and other miscellaneous debris scattered across the surface of the ponds. While the observed flows are relatively simple to comprehend, quantifying gradients and gradient-based properties in them is, by other means than those employed here, a complex and flow-specific task. Semantic flow tracing and LGR provide general access to many field experiments by eliminating the need for highly controlled experimental environments, specialized equipment, and bespoke data processing routines.

These two experiments represent examples of the type of problems that can be addressed using the tools developed in this dissertation. Semantic flow measurements and the ability to compute gradients from sparse data provide easy access to a wide range of studies that currently require complex apparatus and processing. Moreover, they provide a platform from which AMFM systems can be further established.

8.3 Reflections and Recommendations Regarding AMFM

Three or four years ago, when the notion of AMFM began to form as the subject of this thesis, the technological obstacles to be surmounted seemed formidable. In many regards, they still do; real world implementation of AMFM may still be years off. Nevertheless, the progress made in this dissertation provides insight and direction as the technology continues to develop. In this final section, reflections and recommendations pertaining to further development in AMFM are provided. Future work relative to the specific contributions of this thesis have been supplied in their respective chapters, and are therefore left unaddressed here.

Consider Vortices as Tracking Objectives

Early on in this work it was established that AMFM requires objective flow measurements in order to reliably follow structures through the flow. The quintessential objective measurement may be the FTLE, which is notoriously difficult to compute with accuracy on sparse data. Even with the improved performance of LGR, FTLE information would likely prove to be a difficult tracking objective to implement in AMFM.

Vorticity, and vorticity-related metrics like LAVD and KA, seem to present a much more favorable metric for autonomous tracking purposes for a variety of reasons. First, the domain of a vortex is compact in space—its boundaries are enclosed, and it is characterized by a single vortex core at the center. Thus, tracking a vortex is likely far less ambiguous than tracking a hyperbolic (FTLE) ridge over time.

Additionally, in Chapter 4 vortices were found to be more robust to sparsity than hyperbolic features. Using KA in the Double Gyre flow, vortex boundaries were still reasonably identified with one fifth of the tracers required to resolve a semblance of FTLE ridges. Moreover, the KA provided an easily implemented binary criteria for vortex membership.

Finally, in Chapters 6 and 7, vorticity and vorticity-related quantities were more identifiable and interpretable when computed from the motion of imperfect and natural tracers than were the stretch-based metrics. Indeed, the buoyant surface tracers considered in Chapter 6 seemed to be drawn into strong vortices, which highlighted the vortical motion while altering stretching behavior.

Together, these observations suggest that future implementations of AMFM may find greater success by following vortices or by using vortices as landmarks in their flow tracking algorithms than by seeking other objective features like hyperbolic LCS.

Build Backwards-Time Intuition for Integrated Quantities

Integrated gradient-based quantities like LAVD, DRA, and KA seem to be promising in their ability to discern flow structures from sparse, natural data. However, they require that flow quantities be integrated into future times. Thus, in their forward-time formulation, they would be inconvenient for AMFM structure tracking. These methods, however, could also be formulated in the backwards-time context, where integrations would occur over past times which have already been recorded. It is still not certain, however, whether backward-time integrated measures of rotation are useful for forward-time tracking of coherent structure dynamics. Understanding this behavior might be critical to the success of future AMFM systems.

Optimize, Optimize, Optimize

The greatest inhibitor to embedded implementations of AMFM is the computational burden required for tracking structures. AMFM systems must be able to process images in real time while also managing the rest of the overhead required for operation. With the implementations considered throughout this work, that is still unobtainable in most flows.

There are, however, many opportunities for algorithmic optimization that could dramatically improve computation speed. Perhaps the greatest gains may come from optimizing the specific detection procedure, which is currently the bottleneck. Using heavyweight models like Mask-RCNN on a ResNet50 backbone will inevitably be slow. Moreover, the windowing required to improve detection quality, and the masking necessary to identify centroids significantly slows the image processing. Deep detection models have been developed, however, to only identify object centroids. Perhaps if such a model were be applied for tracer detection, processing time would see large speedups.

Performance gains may also be achieved by improving the approach to tracer tracking. The current template-matching scheme requires significant memory and a large number of computations. Point-based algorithms which operate only on tracer positions and do not incorporate image information may be much more efficient.

Many other algorithmic improvements could be made to the current structure detection implementation employed in this dissertation. All of the software built for this project was developed by a single person who is not an experienced software engineer to demonstrate a proof-of-concept. Certainly, trained and talented software designers could affect significant speedups if given to the task.

Chase After the Wind

The outlook for developing AMFM systems seems optimistic. This work has sought to provide a theoretical and algorithmic foundation for the detection of flow structures upon which the remaining pieces of AMFM can be assembled. As AMFM technology and other technologies related to this dissertation develop, it is my hope that they will undergird engineers and scientists globally as they seek to foster universal human flourishing. In their endeavors, may they chase after the wind, measure it, and find it beautiful.

BIBLIOGRAPHY

[1] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Direct computation of velocity gradients from particle trajectories,” *15th International Symposium on Particle Image Velocimetry*, no. 1, Jul. 2023. [Online]. Available: `http://hdl.handle.net/20.500.12680/7p88cp96d`,

[2] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Lagrangian gradient regression for the detection of coherent structures from sparse trajectory data,” *Royal Society Journal of Open Science*, 2024. DOI: `10.48550/arXiv.2310.10994`. [Online]. Available: `https://arxiv.org/abs/2310.10994`,

[3] T. D. Harms, S. L. Brunton, and B. J. McKeon, “Estimating dynamic flow features in groups of tracked objects,” *Under Review*, 2024. DOI: `10.48550/arXiv.2408.16190`. [Online]. Available: `https://arxiv.org/abs/2408.16190`,

[4] T. D. Harms, S. L. Brunton, and B. J. McKeon, *LGR-for-the-detection-of-LCS-from- sparse-data: Associated code for the RSOS article "Lagrangian Gradient Regression for the Detection of Coherent Structures from Sparse Trajectory Data"*, version v1.0.0, Jul. 2024. DOI: `10.5281/zenodo.13126619`. [Online]. Available: `https://doi.org/10.5281/zenodo.13126619`,

[5] U. K. Müller, J. G. M. van den Boogaart, and J. L. van Leeuwen, “Flow patterns of larval fish: Undulatory swimming in the intermediate flow regime,” *Journal of Experimental Biology*, vol. 211, no. 2, pp. 196–205, 2008.

[6] A. Etminan, Y. S. Muzychka, K. Pope, and B. Nyantekyi-Kwakye, “Flow visualization: State-of-the-art development of micro-particle image velocimetry,” *Measurement Science and Technology*, vol. 33, no. 9, p. 092 002, 2022.

[7] C. Jux, A. Sciacchitano, J. F. Schneiders, and F. Scarano, “Robotic volumetric PIV of a full-scale cyclist,” *Experiments in fluids*, vol. 59, pp. 1–15, 2018.

[8] C. Schwarz, A. Bauknecht, C. C. Wolf, A. Coyle, and M. Raffel, “A full-scale rotor-wake investigation of a free-flying helicopter in ground effect using bos and piv,” *Journal of the American Helicopter Society*, vol. 65, no. 3, pp. 1–20, 2020.

[9] N. J. Wei, I. D. Brownstein, J. L. Cardona, M. F. Howland, and J. O. Dabiri, “Near-wake structure of full-scale vertical-axis wind turbines,” *Journal of Fluid Mechanics*, vol. 914, A17, 2021.

[10] G. Shirah, M. Horace, E. Sokolowsky, H. Zhang, D. Menemenlis, and M. Starobin, *Global sea surface currents and temperature, NASA scientific visualization studio*, Accessed: 2024-06-20, 2012. [Online]. Available: `https://svs.gsfc.nasa.gov/3912`.

[11] R. Morrow, L.-L. Fu, F. Ardhuin, *et al.*, "Global observations of fine-scale ocean surface topography with the surface water and ocean topography (swot) mission," *Frontiers in Marine Science*, vol. 6, p. 232, 2019.

[12] E. C. M. Service, *Global ocean physics analysis and forecast*, Accessed: 2024-06-18, 2024. [Online]. Available: `http://marine.copernicus.eu/`.

[13] R. Lumpkin and L. Centurioni, "NOAA global drifter program quality-controlled 6-hour interpolated data from ocean surface drifting buoys," NOAA National Centers for Environmental Information, Dataset, 2019.

[14] S. Businger, R. Johnson, and R. Talbot, "Scientific insights from four generations of Lagrangian smart balloons in atmospheric research," *Bulletin of the American Meteorological Society*, vol. 87, no. 11, pp. 1539–1554, 2006.

[15] R. J. Doviak and D. S. Zrnić, *Doppler radar and weather observations.* Courier Corporation, 2006.

[16] Z. Liu, J. F. Barlow, P.-W. Chan, *et al.*, "A review of progress and applications of pulsed doppler wind LiDARs," *remote sensing*, vol. 11, no. 21, p. 2522, 2019.

[17] A. Sathe and J. Mann, "A review of turbulence measurements using ground-based wind LiDARs," *Atmospheric Measurement Techniques*, vol. 6, no. 11, pp. 3147–3167, 2013.

[18] M. Raffel, C. E. Willert, F. Scarano, C. J. Kähler, S. T. Wereley, and J. Kompenhans, *Particle Image Velocimetry: A Practical Guide.* Cham: Springer International Publishing, 2018, ISBN: 978-3-319-68851-0 978-3-319-68852-7. DOI: `10.1007/978-3-319-68852-7`.

[19] A. Schröder and D. Schanz, "3D Lagrangian particle tracking in fluid mechanics," *Annual Review of Fluid Mechanics*, vol. 55, pp. 511–540, 2023.

[20] C. E. Willert and M. Gharib, "Digital particle image velocimetry," *Experiments in fluids*, vol. 10, no. 4, pp. 181–193, 1991.

[21] R. J. Adrian and J. Westerweel, *Particle Image Velocimetry.* Cambridge University Press, 2011.

[22] K. Nishino, Y. Liu, and R. J. Adrian, "Measurement of Lagrangian velocity by a particle tracking method," *Center for Turbulence Research Report*, vol. CTR-S89, pp. 193–208, 1989.

[23] N. A. Malik and C. Bruecker, "Development of a high precision particle tracking system and its application in a turbulent flow," *Experiments in Fluids*, vol. 14, no. 6, pp. 405–413, 1993.

[24] L. R. Maas, R. J. Adrian, and C. E. Willert, "Visualizations of the flow field in a turbulent boundary layer using high-resolution particle tracking," *Experiments in Fluids*, vol. 15, no. 1, pp. 1–16, 1993.

[25] D. Schanz, S. Gesemann, and A. Schröder, "Shake-the-box: Lagrangian particle tracking at high particle image densities," *Experiments in Flu-*

ids, vol. 57, no. 5, p. 70, May 2016, ISSN: 0723-4864, 1432-1114. DOI: 10.1007/s00348-016-2157-1.

[26] J. J. Gibson, "The perception of the visual world.," 1950.

[27] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *IJCAI'81: 7th international joint conference on Artificial intelligence*, vol. 2, 1981, pp. 674–679.

[28] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial intelligence*, vol. 17, no. 1-3, pp. 185–203, 1981.

[29] C. Lagemann, K. Lagemann, S. Mukherjee, and W. Schröder, "Generalization of deep recurrent optical flow estimation for particle-image velocimetry data," *Measurement Science and Technology*, vol. 33, no. 9, p. 094 003, 2022.

[30] D. N. Beal, F. S. Hover, M. S. Triantafyllou, J. C. Liao, and G. V. Lauder, "Passive propulsion in vortex wakes," *Journal of fluid mechanics*, vol. 549, pp. 385–402, 2006.

[31] M. Moaven, A. Gururaj, V. Raghav, and B. Thurow, "Improving depth uncertainty in plenoptic camera-based velocimetry," *Experiments in Fluids*, vol. 65, no. 4, p. 49, 2024.

[32] M. Shamai, S. T. Dawson, I. Mezić, and B. J. McKeon, "Unsteady dynamics in the streamwise-oscillating cylinder wake for forcing frequencies below lock-on," *Physical Review Fluids*, vol. 6, no. 7, p. 074 702, 2021.

[33] S. Zheng and E. K. Longmire, "Perturbing vortex packets in a turbulent boundary layer," *Journal of fluid mechanics*, vol. 748, pp. 368–398, 2014.

[34] M. Koehler, D. Hess, and C. Brücker, "Flying piv measurements in a 4-valve ic engine water analogue to characterize the near-wall flow evolution," *Measurement Science and Technology*, vol. 26, no. 12, p. 125 302, 2015.

[35] F. Tauro, M. Porfiri, and S. Grimaldi, "Surface flow measurements from drones," *Journal of Hydrology*, vol. 540, pp. 240–245, 2016.

[36] D. Pinton, A. Canestrelli, and L. Fantuzzi, "A uav-based dye-tracking technique to measure surface velocities over tidal channels and salt marshes," *Journal of Marine Science and Engineering*, vol. 8, no. 5, p. 364, 2020.

[37] D. L. Rudnick, "Ocean research enabled by underwater gliders," *Annual review of marine science*, vol. 8, pp. 519–541, 2016.

[38] M. M. Flexas, M. I. Troesch, S. Chien, *et al.*, "Autonomous sampling of ocean submesoscale fronts with ocean gliders and numerical model forecasting," *Journal of Atmospheric and Oceanic Technology*, vol. 35, no. 3, pp. 503–521, 2018.

[39] G. Haller, "Lagrangian coherent structures," *Annual Review of Fluid Mechanics*, vol. 47, pp. 137–162, 2015. DOI: 10.1146/annurev-fluid-010313-141322.

[40] A. Hadjighasem, M. Farazmand, D. Blazevski, G. Froyland, and G. Haller, "A critical comparison of Lagrangian methods for coherent structure detection," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 27, no. 5, p. 053 104, May 2017, ISSN: 1054-1500, 1089-7682. DOI: 10.1063/1.4982720.

[41] M. R. Allshouse and T. Peacock, "Lagrangian based methods for coherent structure detection," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 25, no. 9, p. 097 617, Sep. 2015, ISSN: 1054-1500, 1089-7682. DOI: 10.1063/1.4922968.

[42] M. E. Gurtin, E. Fried, and L. Anand, *The Mechanics and Thermodynamics of Continua*. 32 Avenue of the Americas, New York NY 10013-2473, USA: Cambridge University Press, 2010, ISBN: 978-0-521-40598-0.

[43] G. Haller, *Transport Barriers and Coherent Structures in Flow Data: Advective, Diffusive, Stochastic and Active Methods*. Cambridge University Press, 2023. DOI: 10.1017/9781009225199.

[44] J. C. R. Hunt, A. A. Wray, and P. Moin, "Eddies, streams, and convergence zones in turbulent flows," *Center for Turbulence Research Report*, vol. CTR-S88, pp. 193–208, 1988.

[45] J. Jeong and F. Hussain, "On the identification of a vortex," *Journal of Fluid Mechanics*, vol. 285, pp. 69–94, 1995, ISSN: 0022-1120, 1469-7645. DOI: 10.1017/S0022112095000462.

[46] A. Okubo, "Horizontal dispersion of floatable particles in the vicinity of velocity singularities such as convergences," *Deep Sea Research and Oceanographic Abstracts*, vol. 17, no. 3, pp. 445–454, Jun. 1970, ISSN: 00117471. DOI: 10.1016/0011-7471(70)90059-8.

[47] J. Weiss, "The dynamics of enstrophy transfer in two-dimensional hydrodynamics," *Physica D: Nonlinear Phenomena*, vol. 48, no. 2-3, pp. 273–294, Mar. 1991, ISSN: 01672789. DOI: 10.1016/0167-2789(91)90088-Q.

[48] S. C. Shadden, F. Lekien, and J. E. Marsden, "Definition and properties of Lagrangian coherent structures from finite-time Lyapunov exponents in two-dimensional aperiodic flows," *Physica D: Nonlinear Phenomena*, vol. 212, no. 3-4, pp. 271–304, Dec. 2005, ISSN: 01672789. DOI: 10.1016/j.physd.2005.10.007.

[49] G. Froyland and K. Padberg, "Almost-invariant sets and invariant manifolds—connecting probabilistic and geometric descriptions of coherent structures in flows," *Physica D: Nonlinear Phenomena*, vol. 238, no. 16, pp. 1507–1523, 2009, ISSN: 01672789. DOI: 10.1016/j.physd.2009.03.002.

[50] G. Froyland and K. Padberg-Gehle, "Finite-time entropy: A probabilistic approach for measuring nonlinear stretching," *Physica D: Nonlinear Phenomena*, vol. 241, no. 19, pp. 1612–1628, Oct. 2012, ISSN: 01672789. DOI: 10.1016/j.physd.2012.06.010.

[51] G. Haller, A. Hadjighasem, M. Farazmand, and F. Huhn, "Defining coherent vortices objectively from the vorticity," *Journal of Fluid Me-*

chanics, vol. 795, pp. 136–173, May 2016, ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2016.151.

[52] G. Froyland and K. Padberg-Gehle, "Almost-invariant and finite-time coherent sets: Directionality, duration, and diffusion," in *Ergodic Theory, Open Dynamics, and Coherent Structures*, Springer, 2014, pp. 171–216.

[53] A. Hadjighasem, D. Karrasch, H. Teramoto, and G. Haller, "Spectral-clustering approach to Lagrangian vortex detection," *Physical Review E*, vol. 93, no. 6, p. 063 107, 2016.

[54] G. Froyland and K. Padberg-Gehle, "A rough-and-ready cluster-based approach for extracting finite-time coherent sets from sparse and incomplete trajectory data," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 25, no. 8, p. 087 406, 2015.

[55] K. L. Schlueter-Kuck and J. O. Dabiri, "Coherent structure colouring: Identification of coherent structures from sparse data using graph theory," *Journal of Fluid Mechanics*, vol. 811, pp. 468–486, 2017, ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2016.755.

[56] B. E. Husic, K. L. Schlueter-Kuck, and J. O. Dabiri, "Simultaneous coherent structure coloring facilitates interpretable clustering of scientific data by amplifying dissimilarity," *PLOS ONE*, vol. 14, no. 3, e0212442, 2019.

[57] G. Froyland and O. Junge, "Robust FEM-based extraction of finite-time coherent sets using scattered, sparse, and incomplete trajectories," *SIAM Journal on Applied Dynamical Systems*, vol. 17, no. 2, pp. 1891–1924, 2018.

[58] N. Schilling, G. Froyland, and O. Junge, "Higher-order finite element approximation of the dynamic Laplacian," *ESAIM: Mathematical Modelling and Numerical Analysis*, vol. 54, no. 5, pp. 1777–1795, 2020.

[59] K. Padberg-Gehle and C. Schneide, "Network-based study of Lagrangian transport and mixing," *Nonlinear Processes in Geophysics*, vol. 24, no. 4, pp. 661–671, Oct. 2017, ISSN: 1607-7946. DOI: 10.5194/npg-24-661-2017.

[60] S. Mowlavi, M. Serra, E. Maiorino, and L. Mahadevan, "Detecting Lagrangian coherent structures from sparse and noisy trajectory data," *Journal of Fluid Mechanics*, vol. 948, A4, 2022, ISSN: 0022-1120, 1469-7645. DOI: 10.1017/jfm.2022.652.

[61] G. Haller and G. Yuan, "Lagrangian coherent structures and mixing in two-dimensional turbulence," *Physica D: Nonlinear Phenomena*, vol. 147, no. 3-4, pp. 352–370, Dec. 2000, ISSN: 01672789. DOI: 10.1016/S0167-2789(00)00142-1.

[62] G. Haller, "Distinguished material surfaces and coherent structures in three-dimensional fluid flows," *Physica D: Nonlinear Phenomena*, vol. 149, no. 4, pp. 248–277, Mar. 2001, ISSN: 01672789. DOI: 10.1016/S0167-2789(00)00199-8.

[63] G. Haller and F. J. Beron-Vera, "Geodesic theory of transport barriers in two-dimensional flows," *Physica D: Nonlinear Phenomena*, vol. 241, no. 20, pp. 1680–1702, Oct. 2012, ISSN: 01672789. DOI: `10.1016/j.physd.2012.06.012`.

[64] G. Haller and F. J. Beron-Vera, "Coherent Lagrangian vortices: The black holes of turbulence," *Journal of Fluid Mechanics*, vol. 731, R4, 2013, ISSN: 0022-1120, 1469-7645. DOI: `10.1017/jfm.2013.391`.

[65] M. Farazmand and G. Haller, "Polar rotation angle identifies elliptic islands in unsteady dynamical systems," *Physica D: Nonlinear Phenomena*, vol. 315, pp. 1–12, 2016, ISSN: 01672789. DOI: `10.1016/j.physd.2015.09.007`.

[66] T. Sapsis and G. Haller, "Inertial particle dynamics in a hurricane," *Journal of the Atmospheric Sciences*, vol. 66, no. 8, pp. 2481–2492, 2009.

[67] S. C. Shadden, F. Lekien, J. D. Paduan, F. P. Chavez, and J. E. Marsden, "The correlation between surface drifters and coherent structures based on high-frequency radar data in monterey bay," *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 56, no. 3-5, pp. 161–172, 2009.

[68] A. J. H. M. Reniers, J. H. MacMahan, F. J. Beron-Vera, and M. J. Olascoaga, "Rip-current pulses tied to Lagrangian coherent structures," *Geophysical Research Letters*, vol. 37, no. 5, 2010.

[69] M. J. Olascoaga, F. J. Beron-Vera, G. Haller, *et al.*, "Drifter motion in the Gulf of Mexico constrained by altimetric Lagrangian coherent structures," *Geophysical Research Letters*, vol. 40, no. 23, pp. 6171–6175, 2013.

[70] A. Hadjighasem and G. Haller, "Geodesic transport barriers in jupiter's atmosphere: A video-based analysis," *SIAM Review*, vol. 58, no. 1, pp. 69–89, 2016.

[71] M. Filippi, I. I. Rypina, A. Hadjighasem, and T. Peacock, "An optimized-parameter spectral clustering approach to coherent structure detection in geophysical flows," *Fluids*, vol. 6, no. 1, p. 39, 2021.

[72] P. J. Nolan, J. Pinto, J. González-Rocha, *et al.*, "Coordinated unmanned aircraft system (UAS) and ground-based weather measurements to predict Lagrangian coherent structures (LCSs)," *Sensors*, vol. 18, no. 12, p. 4448, 2018.

[73] M. A. Green, C. W. Rowley, and G. Haller, "Detection of Lagrangian coherent structures in three-dimensional turbulence," *Journal of Fluid Mechanics*, vol. 572, pp. 111–120, 2007.

[74] K. Mulleners and M. Raffel, "The onset of dynamic stall revisited," *Experiments in fluids*, vol. 52, pp. 779–793, 2012.

[75] M. P. Rockwood, K. Taira, and M. A. Green, "Detecting vortex formation and shedding in cylinder wakes using Lagrangian coherent structures," *AIAA journal*, vol. 55, no. 1, pp. 15–23, 2017.

[76] D. Ahmed, A. Javed, M. S. Uz Zaman, M. Mahsud, M.-N. Hanifatu, *et al.*, “Efficient sensor location for HVAC systems using Lagrangian coherent structures,” *Mathematical Problems in Engineering*, vol. 2023, 2023.

[77] N. Amahjour, G. García-Sánchez, M. Agaoglou, and A. M. Mancho, “Analysis of the spread of SARS-CoV-2 in a hospital isolation room using CFD and Lagrangian coherent structures,” *Physica D: Nonlinear Phenomena*, p. 133 825, 2023.

[78] K. Yang, S. Wu, H. Zhang, D. N. Ghista, O. W. Samuel, and K. K. L. Wong, “Lagrangian-averaged vorticity deviation of spiraling blood flow in the heart during isovolumic contraction and ejection phases,” *Medical and Biological Engineering and Computing*, vol. 59, pp. 1417–1430, 2021.

[79] P. J. Nolan, H. Foroutan, and S. D. Ross, “Pollution transport patterns obtained through generalized Lagrangian coherent structures,” *Atmosphere*, vol. 11, no. 2, p. 168, 2020.

[80] P. Tallapragada, S. D. Ross, and D. G. Schmale, “Lagrangian coherent structures are associated with fluctuations in airborne microbial populations,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 21, no. 3, 2011.

[81] S. C. Shadden and C. A. Taylor, “Characterization of coherent structures in the cardiovascular system,” *Annals of biomedical engineering*, vol. 36, pp. 1152–1162, 2008.

[82] 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*, vol. 623, pp. 75–84, 2009.

[83] S. C. Shadden, M. Astorino, and J.-F. Gerbeau, “Computational analysis of an aortic valve jet with Lagrangian coherent structures,” *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 1, 2010.

[84] F. Lekien, S. C. Shadden, and J. E. Marsden, “Lagrangian coherent structures in n-dimensional systems,” *Journal of Mathematical Physics*, vol. 48, no. 6, 2007.

[85] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields* (Applied Mathematical Sciences). New York, NY: Springer New York, 1983, vol. 42, ISBN: 978-1-4612-7020-1 978-1-4612-1140-2. DOI: `10.1007/978-1-4612-1140-2`.

[86] C. Truesdell and K. R. Rajagopal, *An introduction to the mechanics of fluids*. Springer Science and Business Media, 2000.

[87] G. H. Golub and C. F. Van Loan, *Matrix computations*. JHU press, 2013.

[88] G. Haller, “Dynamic rotation and stretch tensors from a dynamic polar decomposition,” *Journal of The Mechanics and Physics of Solids*, 2016. DOI: `10.1016/j.jmps.2015.10.002`.

[89] T. H. Solomon and J. P. Gollub, “Passive transport in steady Rayleigh-Bénard convection,” *The Physics of fluids*, vol. 31, no. 6, pp. 1372–1379, 1988.

[90] T. H. Solomon and J. P. Gollub, “Chaotic particle transport in time-dependent Rayleigh-Bénard convection,” *Physical Review A*, vol. 38, no. 12, p. 6280, 1988.

[91] F. Scarano, “Iterative image deformation methods in piv,” *Measurement science and technology*, vol. 13, no. 1, R1, 2001.

[92] P. T. Tokumaru and P. E. Dimotakis, “Image correlation velocimetry,” *Experiments in Fluids*, vol. 19, no. 1, pp. 1–15, 1995.

[93] X. Ruan, X. Song, and F. Yamamoto, “Direct measurement of the vorticity field in digital particle images,” *Experiments in fluids*, vol. 30, no. 6, pp. 696–704, 2001.

[94] S. Gesemann, F. Huhn, D. Schanz, and A. Schröder, “From noisy particle tracks to velocity, acceleration and pressure fields using B-splines and penalties,” in *18th international symposium on applications of laser and imaging techniques to fluid mechanics, Lisbon, Portugal*, vol. 4, 2016.

[95] J. F. G. Schneiders and F. Scarano, “Dense velocity reconstruction from tomographic PTV with material derivatives,” *Experiments in fluids*, vol. 57, pp. 1–22, 2016.

[96] A. Sciacchitano, B. Leclaire, and A. Schroeder, “Main results of the first Lagrangian particle tracking challenge,” *14th International Symposium on Particle Image Velocimetry*, vol. 1, no. 1, Aug. 2021, ISSN: 2769-7576. DOI: `10.18409/ispiv.v1i1.197`.

[97] L. Graftieaux, M. Michard, and N. Grosjean, “Combining PIV, POD and vortex identification algorithms for the study of unsteady turbulent swirling flows,” *Measurement Science and Technology*, vol. 12, no. 9, pp. 1422–1429, 2001, ISSN: 0957-0233, 1361-6501. DOI: `10.1088/0957-0233/12/9/307`.

[98] S. Essink, V. Hormann, L. R. Centurioni, and A. Mahadevan, “On characterizing ocean kinematics from surface drifters,” *Journal of Atmospheric and Oceanic Technology*, vol. 39, no. 8, pp. 1183–1198, 2022.

[99] I. I. Rypina, T. R. Getscher, L. J. Pratt, and B. Mourre, “Observing and quantifying ocean flow properties using drifters with drogues at different depths,” *Journal of Physical Oceanography*, vol. 51, no. 8, pp. 2463–2482, 2021.

[100] W. J. Saucier, *Principles of meteorological analysis*. University of Chicago Press Chicago, Ill., 1955, vol. 438.

[101] H. Kawai, “Scale dependence of divergence and vorticity of near-surface flows in the sea,” *Journal of the Oceanographical Society of Japan*, vol. 41, pp. 157–166, 1985.

[102] R. Molinari and A. D. Kirwan, “Calculations of differential kinematic properties from lagrangian observations in the western caribbean sea,” *Journal of Physical Oceanography*, vol. 5, no. 3, pp. 483–491, 1975.

[103] I. I. Rypina, T. Getscher, L. J. Pratt, and T. Ozgokmen, "Applying dynamical systems techniques to real ocean drifters," *Nonlinear Processes in Geophysics*, vol. 29, no. 4, pp. 345–361, 2022.

[104] F. Lekien and S. D. Ross, "The computation of finite-time Lyapunov exponents on unstructured meshes and for non-euclidean manifolds," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 1, p. 017 505, Mar. 2010, ISSN: 1054-1500, 1089-7682. DOI: 10.1063/1.3278516.

[105] S. G. Raben, S. D. Ross, and P. P. Vlachos, "Computation of finite-time Lyapunov exponents from time-resolved particle image velocimetry data," *Experiments in Fluids*, vol. 55, no. 1, p. 1638, Jan. 2014, ISSN: 0723-4864, 1432-1114. DOI: 10.1007/s00348-013-1638-8.

[106] S. L. Brunton and C. W. Rowley, "Fast computation of finite-time Lyapunov exponent fields for unsteady flows," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 1, p. 017 503, Mar. 2010, ISSN: 1054-1500, 1089-7682. DOI: 10.1063/1.3270044.

[107] G. Haller, N. Aksamit, and A. P. Encinas-Bartos, "Quasi-objective coherent structure diagnostics from single trajectories," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 31, no. 4, p. 043 131, Apr. 2021, ISSN: 1054-1500, 1089-7682. DOI: 10.1063/5.0044151.

[108] A. P. Encinas-Bartos, N. O. Aksamit, and G. Haller, "Quasi-objective eddy visualization from sparse drifter data," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 32, no. 11, 2022.

[109] B. Kaszás, T. Pedergnana, and G. Haller, "The objective deformation component of a velocity field," *European Journal of Mechanics-B/Fluids*, vol. 98, pp. 211–223, 2023.

[110] N. O. Aksamit, A. P. Encinas-Bartos, G. Haller, and D. E. Rival, *Relative fluid stretching and rotation for sparse trajectory observations*, 2023. arXiv: 2310.05500 [physics.flu-dyn].

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

[112] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

[113] C. M. Bishop and N. M. Nasrabadi, *Pattern recognition and machine learning*. Springer, 2006, vol. 4.

[114] J. Mercer, "Functions of positive and negative type, and their connection the theory of integral equations," *Philosophical Transactions of the Royal Society of London. Series A*, vol. 209, no. 441-458, pp. 415–446, 1909.

[115] D. M. Luchtenburg, S. L. Brunton, and C. W. Rowley, "Long-time uncertainty propagation using generalized polynomial chaos and flow map composition," *Journal of Computational Physics*, vol. 274, pp. 783–802, Oct. 2014, ISSN: 00219991. DOI: 10.1016/j.jcp.2014.06.029.

[116] M. Tabor and I. Klapper, “Stretching and alignment in chaotic and turbulent flows,” *Chaos, Solitons and Fractals*, vol. 4, no. 6, pp. 1031–1055, 1994.

[117] G. Lapeyre, B. L. Hua, and P. Klein, “Dynamics of the orientation of active and passive scalars in two-dimensional turbulence,” *Physics of Fluids*, vol. 13, no. 1, pp. 251–264, 2001.

[118] G. Haller, “An objective definition of a vortex,” *Journal of Fluid Mechanics*, vol. 525, pp. 1–26, 2005. DOI: `10.1017/S0022112004002526`.

[119] A. Etebari and P. P. Vlachos, “Improvements on the accuracy of derivative estimation from DPIV velocity measurements,” *Experiments in Fluids*, vol. 39, pp. 1040–1050, 2005.

[120] S. J. Beresh, N. E. Miller, and B. Smith, “Practical challenges in the calculation of turbulent viscosity from PIV data,” in *2018 Aerodynamic Measurement Technology and Ground Testing Conference*, 2018, p. 2987.

[121] A. Liberzon, T. Käufer, A. Bauer, P. Vennemann, and E. Zimmer, *Openpiv/openpiv-python: Openpiv-python v0.23.6*, version 0.23.6, 2021. DOI: `10.5281/zenodo.5009150`. [Online]. Available: `https://doi.org/10.5281/zenodo.5009150`.

[122] D. H. Kelley and N. T. Ouellette, “Using particle tracking to measure flow instabilities in an undergraduate laboratory experiment,” *American Journal of Physics*, vol. 79, no. 3, pp. 267–273, 2011. DOI: `10.1119/1.3536647`.

[123] N. T. Ouellette, H. Xu, and E. Bodenschatz, “A quantitative study of three-dimensional lagrangian particle tracking algorithms,” *Experiments in Fluids*, vol. 40, pp. 301–313, 2006. DOI: `10.1007/s00348-005-0068-7`.

[124] A. Badza, T. W. Mattner, and S. Balasuriya, “How sensitive are Lagrangian coherent structures to uncertainties in data?” *Physica D: Nonlinear Phenomena*, vol. 444, p. 133 580, 2023.

[125] D. Fortun, P. Bouthemy, and C. Kervrann, “Optical flow modeling and computation: A survey,” *Computer Vision and Image Understanding*, vol. 134, pp. 1–21, 2015.

[126] M. Zhai, X. Xiang, N. Lv, and X. Kong, “Optical flow and scene flow estimation: A survey,” *Pattern Recognition*, vol. 114, p. 107 861, 2021.

[127] S. Baker and I. Matthews, “Lucas-Kanade 20 years on: A unifying framework,” *International journal of computer vision*, vol. 56, pp. 221–255, 2004.

[128] A. Dosovitskiy, P. Fischer, E. Ilg, *et al.*, “Flownet: Learning optical flow with convolutional networks,” in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[129] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, “Flownet 2.0: Evolution of optical flow estimation with deep networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2462–2470.

[130] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, "PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8934–8943.

[131] Z. Teed and J. Deng, "RAFT: Recurrent all-pairs field transforms for optical flow," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, Springer, 2020, pp. 402–419.

[132] Y. Xu, X. Zhou, S. Chen, and F. Li, "Deep learning for multiple object tracking: A survey," *IET Computer Vision*, vol. 13, no. 4, pp. 355–368, 2019.

[133] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, and T.-K. Kim, "Multiple object tracking: A literature review," *Artificial intelligence*, vol. 293, p. 103 448, 2021.

[134] P. Voigtlaender, M. Krause, A. Osep, *et al.*, "MOTS: Multi-object tracking and segmentation," in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 2019, pp. 7942–7951.

[135] P. Sun, J. Cao, Y. Jiang, *et al.*, "Transtrack: Multiple object tracking with transformer," *arXiv preprint arXiv:2012.15460*, 2020.

[136] T. Meinhardt, A. Kirillov, L. Leal-Taixe, and C. Feichtenhofer, "Trackformer: Multi-object tracking with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 8844–8854.

[137] Q. Liu, Q. Chu, B. Liu, and N. Yu, "GSM: Graph Similarity Model for Multi-Object Tracking.," in *IJCAI*, 2020, pp. 530–536.

[138] K. G. Quach, P. Nguyen, H. Le, *et al.*, "Dyglip: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 784–13 793.

[139] D. Guo, Y. Shao, Y. Cui, Z. Wang, L. Zhang, and C. Shen, "Graph attention tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9543–9552.

[140] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, and N. Yu, "Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 4836–4845.

[141] B. Zhan, D. N. Monekosso, P. Remagnino, S. A. Velastin, and L.-Q. Xu, "Crowd analysis: A survey," *Machine Vision and Applications*, vol. 19, pp. 345–357, 2008.

[142] F. L. Sánchez, I. Hupont, S. Tabik, and F. Herrera, "Revisiting crowd behaviour analysis through deep learning: Taxonomy, anomaly detection, crowd emotions, datasets, opportunities and prospects," *Information Fusion*, vol. 64, pp. 318–335, 2020.

[143] A. M. Cheriyadat and R. J. Radke, "Detecting dominant motions in dense crowds," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 4, pp. 568–581, 2008.

[144] N. Verma, S. Patil, B. Sinha, and V. Kulkarni, "Object Detection for COVID Rules Response and Crowd Analysis," in *2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, IEEE, 2021, pp. 1–6.

[145] C. A. S. Pouw, F. Toschi, F. van Schadewijk, and A. Corbetta, "Monitoring physical distancing for crowd management: Real-time trajectory and group analysis," *PloS one*, vol. 15, no. 10, e0240963, 2020.

[146] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[147] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[148] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[149] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of yolo algorithm developments," *Procedia Computer Science*, vol. 199, pp. 1066–1073, 2022.

[150] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*, Springer, 2020, pp. 213–229.

[151] F. Zhuang, Z. Qi, K. Duan, *et al.*, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[152] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *2016 IEEE international conference on image processing (ICIP)*, IEEE, 2016, pp. 3464–3468.

[153] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE international conference on image processing (ICIP)*, IEEE, 2017, pp. 3645–3649.

[154] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[155] A. Kirillov, E. Mintun, N. Ravi, *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[156] Idaho Rangeland Resources Commission, *Amazing drone footage shows thousands of sheep herded down idaho highway | swns*, YouTube, Accessed: 2024-08-07, 2023. [Online]. Available: `https://www.youtube.com/watch?v=MSjbOlQVN58`.

[157] M. R. Maxey and J. J. Riley, "Equation of motion for a small rigid sphere in a nonuniform flow," *The Physics of Fluids*, vol. 26, no. 4, pp. 883–889, 1983.

[158] M. R. Maxey, "The equation of motion for a small rigid sphere in a nonuniform or unsteady flow," *ASME-PUBLICATIONS-FED*, vol. 166, pp. 57–57, 1993.

[159] C. M. Tchen, "Mean value and correlation problems connected with the motion of small particles in a turbulent fluid," *The Hague*, 1947.

[160] S. G. Prasath, V. Vasan, and R. Govindarajan, "Accurate solution method for the maxey–riley equation, and the effects of basset history," *Journal of Fluid Mechanics*, vol. 868, pp. 428–460, 2019.

[161] M. B. Pinsky and A. P. Khain, "Turbulence effects on droplet growth and size distribution in clouds—a review," *Journal of aerosol science*, vol. 28, no. 7, pp. 1177–1214, 1997.

[162] C. P. Cummins, O. J. Ajayi, F. V. Mehendale, R. Gabl, and I. M. Viola, "The dispersion of spherical droplets in source–sink flows and their relevance to the covid-19 pandemic," *Physics of Fluids*, vol. 32, no. 8, 2020.

[163] F. J. Beron-Vera, M. J. Olascoaga, and P. Miron, "Building a maxey–riley framework for surface ocean inertial particle dynamics," *Physics of Fluids*, vol. 31, no. 9, 2019.

[164] M. Sudharsan, S. L. Brunton, and J. J. Riley, "Lagrangian coherent structures and inertial particle dynamics," *Physical Review E*, vol. 93, no. 3, p. 033 108, 2016.

[165] R. Lopez-Acosta, M. Schodlok, and M. Wilhelmus, "Ice floe tracker: An algorithm to automatically retrieve Lagrangian trajectories via feature matching from moderate-resolution visual imagery," *Remote Sensing of Environment*, vol. 234, p. 111 406, Dec. 2019, ISSN: 00344257. DOI: `10.1016/j.rse.2019.111406`.

[166] G. E. Manucharyan, R. Lopez-Acosta, and M. M. Wilhelmus, "Spinning ice floes reveal intensification of mesoscale eddies in the western arctic ocean," *Scientific Reports*, vol. 12, no. 1, p. 7070, 2022, ISSN: 2045-2322. DOI: `10.1038/s41598-022-10712-z`.

[167] E. van Sebille, M. H. England, and G. Froyland, "Origin, dynamics and evolution of ocean garbage patches from observed surface drifters," en, *Environmental Research Letters*, vol. 7, no. 4, p. 044 040, Dec. 2012, ISSN: 1748-9326. DOI: `10.1088/1748-9326/7/4/044040`.

[168] E. van Sebille, S. M. Griffies, R. Abernathey, *et al.*, "Lagrangian ocean analysis: Fundamentals and practices," en, *Ocean Modelling*, vol. 121, pp. 49–75, Jan. 2018, ISSN: 14635003. DOI: `10.1016/j.ocemod.2017.11.008`.

[169] L. Baker and M. DiBenedetto, "Large-scale particle shadow tracking and orientation measurement with collimated light," *Experiments in Fluids*, vol. 64, no. 3, p. 52, 2023.

[170] M. H. DiBenedetto, N. T. Ouellette, and J. R. Koseff, "Transport of anisotropic particles under waves," *Journal of Fluid Mechanics*, vol. 837, pp. 320–340, 2018.

[171] J. O. Dabiri, M. F. Howland, M. K. Fu, and R. H. Goldshmid, “Visual anemometry for physics-informed inference of wind,” *Nature Reviews Physics*, vol. 5, no. 10, pp. 597–611, 2023.

[172] L. J. Baker and F. Coletti, “Experimental investigation of inertial fibres and disks in a turbulent boundary layer,” *Journal of Fluid Mechanics*, vol. 943, A27, 2022.

[173] M. L. Hooper, *Characterization and Optimization of a Fully Passive Flapping Foil in an Unsteady Environment for Power Production and Propulsion*. California Institute of Technology, 2022.

[174] M. L. Hooper and B. J. McKeon, “A representative driven system to interrogate passive dynamics of an airfoil in the wake of a cylinder,” in *14th International Symposium on Particle Image Velocimetry*, vol. 1, 2021.

[175] D. D. Joseph, J. Wang, R.-y. Bai, B. H. Yang, and H. H. Hu, “Particle motion in a liquid film rimming the inside of a partially filled rotating cylinder,” *Journal of Fluid Mechanics*, vol. 496, pp. 139–163, 2003.

[176] M. Gharib and A. Weigand, “Experimental studies of vortex disconnection and connection at a free surface,” *Journal of Fluid Mechanics*, vol. 321, pp. 59–86, 1996. DOI: `10.1017/S0022112096007641`.

[177] C. E. WIllert and M. Gharib, “The interaction of spatially modulated vortex pairs with free surfaces,” *Journal of Fluid Mechanics*, vol. 345, pp. 227–250, 1997. DOI: `10.1017/S0022112097006265`.

[178] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: `10.1038/s41592-019-0686-2`.

[179] I. Fouxon, “Distribution of particles and bubbles in turbulence at a small stokes number,” *Physical review letters*, vol. 108, no. 13, p. 134 502, 2012.

[180] R. C. Hale, M. E. Seeley, M. J. La Guardia, L. Mai, and E. Y. Zeng, “A global perspective on microplastics,” *Journal of Geophysical Research: Oceans*, vol. 125, no. 1, Jan. 2020, ISSN: 2169-9275, 2169-9291. DOI: `10.1029/2018JC014719`.

[181] A. R. Khojasteh, W. van de Water, and J. Westerweel, “Practical object and flow structure segmentation using artificial intelligence,” *Experiments in Fluids*, vol. 65, no. 8, p. 119, 2024.

[182] M. Shamai, *Complexity reduction of fluid-structure systems at low forcing frequencies*. California Institute of Technology, 2021.

[183] D. P. Huynh, *Spatio-Temporal Response of a Compliant-Wall, Turbulent Boundary Layer System to Dynamic Roughness Forcing*. California Institute of Technology, 2019.

[184] E. A. L. Hufstedler, *Experimental Generation and Modeling of Vortical Gusts and Their Interactions with an Airfoil*. California Institute of Technology, 2017.

[185] R. Szeliski, *Computer vision: algorithms and applications*. Springer Nature, 2022.

[186] A. Dosovitskiy, L. Beyer, A. Kolesnikov, *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” *arXiv preprint arXiv:2010.11929*, 2020.

[187] T.-Y. Lin, M. Maire, S. Belongie, *et al.*, “Microsoft coco: Common objects in context,” in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, Springer, 2014, pp. 740–755.

[188] V. Tian, B. McKeon, and I. Leyva, “Split stream flow past a blunt trailing edge with application to combustion instabilities,” in *48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, 2012, p. 3807.

[189] P. J. Schmid, D. S. Henningson, and D. F. Jankowski, “Stability and transition in shear flows. applied mathematical sciences, vol. 142,” *Appl. Mech. Rev.*, vol. 55, no. 3, B57–B59, 2002.

[190] H. R. Quintanilha Jr and L. S. Alves, “On the existance of multiple self-excitation frequencies in the instability of interacting planar mixing layer and wake,” in *31st Congress of the International Council of the Aeronautical Sciences*, ICAS, Belo Horizonte, MG, Brazil, Oct. 2018.

Appendix A

OBJECTIVITY OF FLOW FIELD MEASUREMENTS

A hallmark of the theory of Lagrangian coherent structures is the property of objectivity, which ensures that the computed quantities remain consistent regardless of the motion of the observer. While it has been briefly touched on in the body of the paper, objectivity has not yet been thoroughly discussed in this work, but rather assumed. In this appendix, various discussions of objectivity from relevant works are synthesized to provide an overview of the objectivity of Lagrangian quantities. With the exception of the discussion regarding the objectivity of the composition operation, all of the results can be found throughout the literature. The purpose of this appendix is to summarize the objectivity of some relevant Lagrangian quantities and operations in one location so that the reader does not need to sift the literature for them. Readers interested in greater depth should refer to the helpful chapter on objectivity from Haller's recent textbook on LCS [43] and to many articles touching the subject [42], [65], [86], [88], [118], among others.

We say that a quantity is objective if it exhibits invariance under Euclidean transformations of the form

$$\tilde{\mathbf{x}}(t) = \mathbf{Q}(t)\mathbf{x}(t) + \mathbf{p}(t), \tag{A.1}$$

where $\mathbf{Q}(t)$ is a proper orthogonal rotation tensor and $\mathbf{p}(t)$ is a translation. The objectivity of a scalar, vector, or tensor quantity is examined by considering the influence of changes of frame in the form of equation A.1 on the resulting value. To aid the discussion, some definitions from continuum mechanics are useful [42].

Definition 1 (Objectivity or Frame Indifference) *A scalar field g is frame-indifferent if it is unchanged by frame rotation and translation*

$$\tilde{g} = g, \tag{A.2}$$

where $\tilde{\cdot}$ represents the transformed quantity. Moreover, a vector field $\mathbf{g}$ is frame-indifferent if it simply rotates with the frame rotation

$$\tilde{\mathbf{g}} = \mathbf{Q}\mathbf{g}, \tag{A.3}$$

and a tensor field $\mathbf{G}$ *is frame-indifferent if, given frame-indifferent vector fields* $\mathbf{g}$ *and* $\mathbf{h}$,

$$\mathbf{h} = \mathbf{G}\mathbf{g} \implies \tilde{\mathbf{h}} = \tilde{\mathbf{G}}\tilde{\mathbf{g}}. \tag{A.4}$$

Then, the transformation law for a frame-indifferent tensor field is

$$\tilde{\mathbf{G}} = \mathbf{Q}\mathbf{G}\mathbf{Q}^\top. \tag{A.5}$$

In other words, a frame indifferent vector is one such that its magnitude remains unchanged by the change of frame, and a frame indifferent tensor is one that maps indifferent vectors into indifferent vectors [86]. Quantities that are frame-indifferent are objective in the sense discussed throughout this paper.

The concept of invariance is stronger than that of frame-indifference.

Definition 2 (Invariance) *A vector field or tensor field is invariant if it remains unchanged under transformations of the form presented in equation A.1. Therefore, any invariant quantity* $\mathbf{H}$ *obeys the transformation law*

$$\tilde{\mathbf{H}} = \mathbf{H}. \tag{A.6}$$

Furthermore, any scalar valued function of a tensor $f(\mathbf{G})$ *is invariant if and only if*

$$f(\mathbf{Q}\mathbf{G}\mathbf{Q}^\top) = f(\mathbf{G}) \quad \forall |\det \mathbf{Q}| = 1. \tag{A.7}$$

Invariance implies objectivity.

Using these definitions, the objectivity of $D\mathbf{F}_{t_0}^t$, $\mathbf{C}_{t_0}^t$, and $\sigma_{t_0}^t$ is readily assessed [42], [48]. Using the definition of the flow map from equation 2.4, the transformed flow map is given

$$\tilde{\mathbf{x}}(t) = \mathbf{Q}(t)\mathbf{x}(t) + \mathbf{p}(t) = \mathbf{Q}(t)\mathbf{F}_{t_0}^t(\mathbf{x}_0) + \mathbf{p}(t). \tag{A.8}$$

Computing the gradient with respect to $\mathbf{x}_0$ yields

$$D\tilde{\mathbf{F}}_{t_0}^t = \mathbf{Q}(t)D\mathbf{F}_{t_0}^t. \tag{A.9}$$

Hence, $D\mathbf{F}_{t_0}^t$ is not objective. This result can be used to see that $\mathbf{C}_{t_0}^t$ is invariant:

$$\begin{aligned} \tilde{\mathbf{C}}_{t_0}^t &= \left(\mathbf{Q}(t)D\mathbf{F}_{t_0}^t\right)^\top \mathbf{Q}(t)D\mathbf{F}_{t_0}^t \\ &= (D\mathbf{F}_{t_0}^t)^\top \mathbf{Q}^\top(t)\mathbf{Q}(t)D\mathbf{F}_{t_0}^t \\ &= (D\mathbf{F}_{t_0}^t)^\top D\mathbf{F}_{t_0}^t \\ &= \mathbf{C}_{t_0}^t. \end{aligned} \tag{A.10}$$

Then, since $\mathbf{C}_{t_0}^t$ is invariant, $\sigma_{t_0}^t$ is also invariant.

Objectivity of the Composition Operation

Computing the FTLE using initial and final times alone is objective because the analysis is based on the right Cauchy-Green tensor which, as has been shown, is invariant. However, computing the FTLE via composition according to equation 3.16 depends on many instances of the relative deformation gradient, which is not objective. For a single instance of composition,

$$D\mathbf{F}_{t_0}^{t} = D\mathbf{F}_{s}^{t} D\mathbf{F}_{t_0}^{s}, \tag{A.11}$$

from which an expression for the relative Jacobian is obtained

$$D\mathbf{F}_{s}^{t} = D\mathbf{F}_{t_0}^{t} \left(D\mathbf{F}_{t_0}^{s} \right)^{-1}. \tag{A.12}$$

Under a change of reference, this is expressed as

$$\begin{aligned} D\tilde{\mathbf{F}}_{s}^{t} &= D\tilde{\mathbf{F}}_{t_0}^{t} \left(D\tilde{\mathbf{F}}_{t_0}^{s} \right)^{-1} \\ &= \mathbf{Q}(t) D\mathbf{F}_{t_0}^{t} \left(\mathbf{Q}(s) D\mathbf{F}_{t_0}^{s} \right)^{-1} \\ &= \mathbf{Q}(t) D\mathbf{F}_{t_0}^{t} D\mathbf{F}_{s}^{t_0} \mathbf{Q}^{\top}(s) \\ &= \mathbf{Q}(t) D\mathbf{F}_{s}^{t} \mathbf{Q}^{\top}(s). \end{aligned} \tag{A.13}$$

Therefore, in general,

$$D\tilde{\mathbf{F}}_{t_i}^{t_{i+1}} = \mathbf{Q}(t_{i+1}) \mathbf{F}_{t_i}^{t_{i+1}} \mathbf{Q}^{\top}(t_i). \tag{A.14}$$

Applying this to equation 3.16, an expression for the composite flow map Jacobian under a change of reference is

$$\begin{aligned} D\tilde{\mathbf{F}}_{t_0}^{t_n}(\mathbf{x}_0) &= \prod_{i=0}^{n-1} D\tilde{\mathbf{F}}_{t_i}^{t_{i+1}} \left(\mathbf{x}(t_i) \right) \\ &= \prod_{i=0}^{n-1} \mathbf{Q}(t_{i+1}) D\mathbf{F}_{t_i}^{t_{i+1}} \left(\mathbf{x}(t_i) \right) \mathbf{Q}^{\top}(t_i), \\ &= \mathbf{Q}(t_n) D\mathbf{F}_{t_{n-1}}^{t_n}(\mathbf{x}(t_{n-1})) \mathbf{Q}^{\top}(t_{n-1}) \\ &\quad \mathbf{Q}(t_{n-1}) D\mathbf{F}_{t_{n-2}}^{t_{n-1}}(\mathbf{x}(t_{n-2})) \mathbf{Q}^{\top}(t_{n-2}) \cdots \\ &\quad \mathbf{Q}(t_1)^{\top} \mathbf{Q}(t_1) D\mathbf{F}_{t_1}^{t_0}(\mathbf{x}(t_0)) \mathbf{Q}^{\top}(t_0) \\ &= \mathbf{Q}(t_n) \prod_{i=0}^{n-1} D\mathbf{F}_{t_i}^{t_{i+1}} \left(\mathbf{x}(t_i) \right) \\ &= \mathbf{Q}(t_n) D\mathbf{F}_{t_0}^{t_n}(\mathbf{x}_0), \end{aligned} \tag{A.15}$$

which is identical to $D\tilde{\mathbf{F}}_{t_0}^{t_n}$ computed from the only the first and last time instances. Thus, the operation of composition does not affect the objectivity of the flow map Jacobian or associated quantities.

Objectivity of the polar decomposition

As discussed in Chapter 2, the polar decomposition separates the flow map Jacobian into a proper orthogonal rotation tensor $\mathbf{R}_{t_0}^t$ and a symmetric positive definite right stretch tensor $\mathbf{U}_{t_0}^t$ or left stretch tensor $\mathbf{V}_{t_0}^t$ such that

$$D\mathbf{F}_{t_0}^t = \mathbf{R}_{t_0}^t \mathbf{U}_{t_0}^t = \mathbf{V}_{t_0}^t \mathbf{R}_{t_0}^t.$$

A property of the right stretch tensor $\mathbf{U}_{t_0}^t$ is that $(\mathbf{U}_{t_0}^t)^2 = \mathbf{C}$. As a result, the right stretch tensor is objective. Then, if we consider the transformation of the flow map Jacobian,

$$D\tilde{\mathbf{F}}_{t_0}^t = \mathbf{Q} D\mathbf{F}_{t_0}^t = \mathbf{Q}\mathbf{R}_{t_0}^t \mathbf{U}_{t_0}^t, \tag{A.16}$$

and $\tilde{\mathbf{R}}_{t_0}^t = \mathbf{Q}\mathbf{R}_{t_0}^t$. Moreover, it follows that

$$\tilde{\mathbf{V}}_{t_0}^t = \mathbf{Q} D\mathbf{F}_{t_0}^t (\mathbf{R}_{t_0}^t)^{-1} \mathbf{Q}^\top = \mathbf{Q}\mathbf{V}_{t_0}^t \mathbf{Q}^\top. \tag{A.17}$$

Therefore, the right stretch tensor is invariant, the left stretch tensor is objective, but not objective, and the rotation tensor is not objective.

Since the tensor components of the dynamic polar decomposition have not factored significantly into this work, their objectivity characteristics are not developed here. They are thoroughly documented by [88], where the decomposition is developed.

Objectivity of $\nabla\mathbf{v}$, $\mathbf{W}$, and $\mathbf{D}$

The objectivity of the velocity gradient and its spin and dilatation (often called stretch or stretching) components begins by considering the objectivity of the rate of change of the flow map Jacobian [86]. Using the chain rule and abbreviating $\mathbf{Q}(t)$ as $\mathbf{Q}$,

$$\frac{d}{dt} D\tilde{\mathbf{F}}_{t_0}^t = \mathbf{Q}\left(\frac{d}{dt} D\mathbf{F}_{t_0}^t\right) + \left(\frac{d}{dt}\mathbf{Q}\right) D\mathbf{F}_{t_0}^t. \tag{A.18}$$

Now, by equation 2.11, $\frac{d}{dt} D\mathbf{F}_{t_0}^t = \nabla\mathbf{v}(D\mathbf{F}_{t_0}^t)$. Inserting this into the above and applying equation A.9 yields

$$\tilde{\nabla}\mathbf{v}\left(D\tilde{\mathbf{F}}_{t_0}^t\right) = \mathbf{Q}\nabla\mathbf{v}\left(\mathbf{Q}^\top D\tilde{\mathbf{F}}_{t_0}^t\right) + \frac{d}{dt}\mathbf{Q}\left(\mathbf{Q}^\top D\tilde{\mathbf{F}}_{t_0}^t\right). \tag{A.19}$$

Since $D\tilde{\mathbf{F}}_{t_0}^t$ is invertible, it can be removed from the equation

$$\nabla\tilde{\mathbf{v}} = \mathbf{Q}(\nabla\mathbf{v})\mathbf{Q}^\top + \boldsymbol{\Omega}, \tag{A.20}$$

where the the frame spin is defined as

$$\mathbf{\Omega} = \frac{d}{dt}\mathbf{Q}\mathbf{Q}^\top, \tag{A.21}$$

and is a skew tensor defining the rate of rotation of the observer. Here, equation A.20 represents the transformation equation for the velocity gradient. Therefore, the velocity gradient is neither objective nor frame-indifferent.

Splitting this into its components,

$$\tilde{\mathbf{D}} + \tilde{\mathbf{W}} = \mathbf{Q}(\mathbf{D} + \mathbf{W})\mathbf{Q}^\top + \mathbf{\Omega}. \tag{A.22}$$

Since both $\mathbf{W}$ and $\mathbf{\Omega}$ are skew, and $\mathbf{D}$ is symmetric, the transformation equations for $\mathbf{D}$ and $\mathbf{W}$ are given as

$$\tilde{\mathbf{D}} = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top, \tag{A.23}$$

$$\tilde{\mathbf{W}} = \mathbf{Q}\mathbf{W}\mathbf{Q}^\top + \mathbf{\Omega}. \tag{A.24}$$

Therefore, the dilatation tensor $\mathbf{D}$ is objective but not invariant, and the spin tensor $\mathbf{W}$ is not objective. As a consequence, the principal strain as defined in equation 2.15 is objective and the vorticity is not.

Objectivity of metrics for material rotation

Using the transformation laws for the dilatation and spin tensors allows for transformation laws of scalar rotation metrics to be derived. As discussed with equation 2.14, the vorticity at the location of a particle along its trajectory may be computed according to

$$\mathbf{W}\mathbf{e} = -\frac{1}{2}\omega \times \mathbf{e}, \quad \forall \mathbf{e} \in \mathbb{R}^d.$$

Applying the transformation, one obtains

$$\begin{aligned}
\tilde{\mathbf{W}}\mathbf{e} &= (\mathbf{Q}\mathbf{W}\mathbf{Q}^\top + \mathbf{\Omega})\mathbf{e} \\
-\frac{1}{2}\tilde{\omega} \times \mathbf{e} &= \mathbf{Q}(-\frac{1}{2}\omega \times \mathbf{e}) - \frac{1}{2}\dot{\mathbf{q}} \times \mathbf{e} \\
\tilde{\omega} \times \mathbf{e} &= \mathbf{Q}\omega \times \mathbf{e} + \dot{\mathbf{q}} \times \mathbf{e} \\
\tilde{\omega} &= \mathbf{Q}\omega + \dot{\mathbf{q}},
\end{aligned} \tag{A.25}$$

where $\dot{\mathbf{q}}$ is a vector representation of rate of frame rotation. Furthermore, the transformation law for Q from the Q-criterion (equation 2.16) can be assessed:

$$\begin{aligned}
2\tilde{Q} &= \|\tilde{\mathbf{W}}\|_F^2 - \|\tilde{\mathbf{D}}\|_F^2 \\
&= \operatorname{tr}(\tilde{\mathbf{W}}^\top \tilde{\mathbf{W}}) - \operatorname{tr}(\tilde{\mathbf{D}}^\top \tilde{\mathbf{D}}) \\
&= \operatorname{tr}((\mathbf{QWQ}^\top + \mathbf{\Omega})^\top (\mathbf{QWQ}^\top + \mathbf{\Omega})) - \operatorname{tr}((\mathbf{QDQ}^\top)^\top (\mathbf{QDQ}^\top)) \\
&= \operatorname{tr}(\mathbf{QW}^\top \mathbf{WQ}^\top) - \operatorname{tr}(\mathbf{QD}^\top \mathbf{DQ}^\top) + 2\operatorname{tr}(\mathbf{QWQ}^\top \mathbf{\Omega}) + \operatorname{tr}(\mathbf{\Omega}^\top \mathbf{\Omega}) \\
&= 2Q + 2\operatorname{tr}(\mathbf{QWQ}^\top \frac{d}{dt}\mathbf{QQ}^\top) + \|\mathbf{\Omega}\|_F^2 \\
&= 2Q - 2\operatorname{tr}(\mathbf{W}^\top \mathbf{\Omega}) + \|\mathbf{\Omega}\|_F^2 \\
\tilde{Q} &= Q - \langle \mathbf{W}, \mathbf{\Omega} \rangle_F + \frac{1}{2}\|\mathbf{\Omega}\|_F^2 ,
\end{aligned} \tag{A.26}$$

where $\langle \cdot \rangle_F$ represents the Frobenius inner product. Here, the term $\|\mathbf{\Omega}\|_F^2$ represents the magnitude of the frame rotation irrespective of the flow and the term $\langle \mathbf{W}, \mathbf{\Omega} \rangle_F$ represents the magnitude of the relative rotation between the flow and the observer. Because the transformation laws of both vorticity and Q-criterion indicate that the observed value varies as a function of frame rotation, they are not objective. This is intuitive, as one would expect that the observed rotation of a flow would appear different if observed from a rotating vantage.

The objectivity of the LAVD (also the IRA) and the DRA are thoroughly discussed in [88] and [51] where they are developed. The LAVD and the IRA $\psi_{t_0}^t$ are both objective and dynamically consistent in two and three dimensions. On the other hand, the dynamic rotation angle $\varphi_{t_0}^t$ is not objective. In [65], the authors remark that closed level sets of the polar rotation angle are objective, though it is not in general. For more information about the usage and objectivity of these quantities, refer to the sources listed above.

Table A.1: Summary of objectivity and transformation laws for quantities of importance in the study of geometric LCS

Quantity	Objective	Transformation Law
Flow map Jacobian	✗	$D\tilde{\mathbf{F}}_{t_0}^{t} = \mathbf{Q} D\mathbf{F}_{t_0}^{t}$
Flow map Jacobian (by composition)	✗	$D\tilde{\mathbf{F}}_{t_0}^{t} = \mathbf{Q}\prod_{i=0}^{n-1} D\mathbf{F}_{t_i}^{t_{i+1}}\left(\mathbf{x}(t_i)\right)$
Cauchy-Green tensor	✓	$\tilde{\mathbf{C}}_{t_0}^{t} = \mathbf{C}_{t_0}^{t}$
Rotation tensor	✗	$\tilde{\mathbf{R}}_{t_0}^{t} = \mathbf{Q}\mathbf{R}_{t_0}^{t}$
Right stretch tensor	✓	$\tilde{\mathbf{U}}_{t_0}^{t} = \mathbf{U}_{t_0}^{t}$
Left stretch tensor	✓	$\tilde{\mathbf{V}}_{t_0}^{t} = \mathbf{Q}\mathbf{V}_{t_0}^{t}\mathbf{Q}^{\top}$
Velocity gradient	✗	$\nabla\tilde{\mathbf{v}} = \mathbf{Q}\nabla\mathbf{v}\mathbf{Q}^{\top} + \mathbf{\Omega}$
Spin tensor	✗	$\tilde{\mathbf{W}} = \mathbf{Q}\mathbf{W}\mathbf{Q}^{\top} + \mathbf{\Omega}$
Dilatation tensor	✓	$\tilde{\mathbf{D}} = \mathbf{Q}\mathbf{D}\mathbf{Q}^{\top}$
FTLE	✓	$\tilde{\sigma}_{t_0}^{t} = \sigma_{t_0}^{t}$
Vorticity	✗	$\tilde{\omega} = \mathbf{Q}\omega + \dot{\mathbf{q}}$
Q-Criterion	✗	$\tilde{Q} = Q - \langle \mathbf{W}, \mathbf{\Omega}\rangle_F + \frac{1}{2}\left\Vert\mathbf{\Omega}\right\Vert_F^2$
Intrinsic rotation angle (LAVD)	✓	$\tilde{\psi}_{t_0}^{t} = \psi_{t_0}^{t}$
Polar rotation angle	✗	—
Dynamic rotation angle	✗	—

Appendix B

ADDITIONAL EXPERIMENTAL DETAILS

A variety of experiments were conducted throughout the duration of this project. While the most pertinent details are included in the body of the thesis text, many details are left out for the purpose of flow and brevity. This appendix aims to provide all necessary details for the laboratory and field experiments which factored into the presented work.

B.1 Experimental Facilities

Three main experiments are presented in this work: a split flow over a blunt trailing edge, an evaluation of imperfect particle surface trajectories to near-surface flow fields, and field experiments of flows on the surface of the Caltech turtle ponds. Since both of the laboratory experiments were conducted using the NOAH water channel at the Graduate Aerospace Laboratories at Caltech (GALCIT), the experimental facilities are discussed here separately from the particular details of each experiment.

NOAH Water Channel

The NOAH water channel is a recirculating free surface flow measurement facility shown in figure B.1. The test section of the channel is 46 cm wide by 150 cm long, with a wall height of 61 cm. In the presented experiments, the channel is filled to a depth of approximately 46 cm. Prior to entering the test section, flow passes through a settling chamber with a series of flow conditioning elements including 7.6 cm thick honeycomb panels, perforated stainless steel plates, fine wire meshes, and a 4:1 contraction perpendicular to the flow and water depth. All of these elements serve to reduce the turbulence intensity of the flow entering into the test section. During operation, the VFD-powered pump is capable of moving water at speeds up to approximately 1 m/s. The speeds achieved in these experiments, however, are kept much lower. The side panels and floor of the test section are built from transparent Plexiglass to allow for optical access to flow phenomena on all sides. Recent studies conducted in the NOAH water channel include the work of Hooper [173], Shamai [182], Huynh [183], and Hufstedler [184].

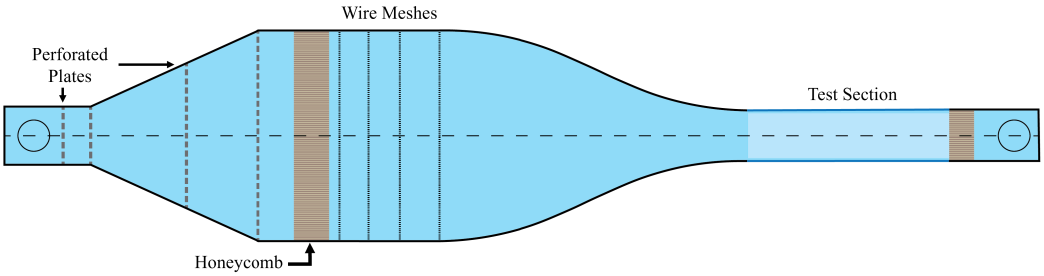


Figure B.1: Top-down schematic of the NOAH water channel facility.

Optical Hardware

Various optical diagnostics were implemented throughout the duration of this work. In the laboratory, the primary mode of flow field measurement was 2-dimension, 2-component particle image velocimetry (2D2C PIV), and therefore required illumination and imaging of flow tracers in a planar sheet. The flow tracers used during the experiments were approximately 10 microns in diameter, and were therefore assumed to be ideal tracers.

Tracer illumination in early experiments was achieved using a Photonics DM20-527(nm) YLF dual-head laser operated in single-pulse mode. This laser is capable of achieving pulses with as short as 100 ns width and up to 200 mJ while operating at frequencies up to 10 kHz. In later experiments this laser became inoperable and a 10 W 445 nm continuous diode laser was implemented. Since the flow speed during all experiments was relatively low speed, the need for high-speed pulsing was not necessary, and the continuous diode laser was suitable. For both lasers, the beam was spread using a cylindrical lens and reflected into the test section by a mirror.

Three camera types were used during the experiments performed in the experiments. PIV studies were performed using high-speed Phantom Miro Lab 320 cameras capable of sampling at 1380 Hz with 1920×1200 pixel resolution with 10-bit depth on a single channel. The cameras have 6GB of memory and 12GB of high-speed internal RAM, meaning that the camera was capable of sampling 3767 images in a single run before requiring data to be sent to an external storage device. The Phantom Miro cameras are equipped with Nikon F-mount lens connectors, and a variety of lenses were available in the laboratory.

A large portion of this work involved sampling flows using cheaply available or hobbyist imaging equipment. For such studies, two cameras were used. For both laboratory and field work, a Teledyne/FLIR Blackfly model BFS-U3-23S3C-C machine vision camera was often selected. This camera is inexpensive (< \$500 USD), fits a small form factor (roughly 2.5 cm^3), and is capable of sampling color images with 1920 × 1200 pixel resolution at 163 Hz with a global shutter. Because of its small size, the on-board memory of the camera is limited. Therefore, experiments involving its use were tethered to a computer by a high-bitrate USB 3.1 cable. The transmission rate of the cable and the RAM capacity on the local device were the limiting factors for the achievable frame-rate and image quality. In most instances during these experiments, full-size images were recorded at 100 Hz or less. The Blackfly uses a C-mount, so additional lenses were required for this device.

The final camera that was occasionally used throughout this work is a Nikon D800 DSLR Camera. The D800 is a full frame (35.9 × 24 mm sensor) single-lens reflex (SLR) camera with a medium-sized body that is commonly used by hobbyist and professional photographers. The maximum resolution of the camera is 7360 × 4912 pixels, making it an excellent device for dye visualization photography and other single-frame diagnostics. In video mode, however, the camera is only capable of achieving 1920 × 1020 pixel resolution at 30 Hz. Like the Phantom Miro cameras, the D800 uses a Nikon F-mount.

Captive Trajectory System

A prominent feature of the NOAH facility is the captive trajectory system (CTS) which is fixed to the top of the water channel test section. The CTS is a robotic flow manipulation device capable of motions in the three translational axes and in pitch.

The CTS enables flow manipulation elements to be moved along prescribed and captive trajectories through the flow. A prescribed trajectory is one which is pre-defined prior to execution of the test, while a captive trajectory is one that is programmed to respond to the flow given some means of flow sensing. Both forms of actuation have been implemented in recent studies in the lab.

Work by Shamai and collaborators defines oscillatory cylinder motion through prescribed CTS trajectories [32], [182], and the studies by Hooper and collaborators incorporate both prescribed and captive trajectories to study the dynamics of an airfoil in the wake of a cylinder [173], [174]. The CTS motions are rate-limited to 200 Hz by hardware constraints, which limits the experimental range of the device. However, given the nature of flows in the NOAH water channel, this limitation does not pose a serious constraint. In this work, only simple prescribed trajectories are implemented on the CTS device.

B.2 Camera Calibration

The experiments performed in this work required camera calibration to ground the results in physical reality. As will be discussed in more depth with each relevant experiment, calibration for PIV images was performed in the PIV image processing software DaVis by LaVision. The methods developed for LGR, however, required that an alternative approach to calibration be implemented. For this, the ChArUco calibration approach was selected for its versatility.

Calibration Background

Camera calibration defines a relationship from world coordinates $[X, Y, Z]^\top$ to camera coordinates $[X_c, Y_c, Z_c]^\top$ to pixel coordinates $[x_p, y_p]^\top$. The standard calibration procedure, and the one that was selected for this work, involves three steps: 1. rectilinear distortion coefficient estimation, 2. intrinsic matrix calibration, and 3. homography to dimensional units. While calibration basics are provided in the following discussion, a more thorough explanation can be found in any computer vision text book (e.g., the text by Szeliski [185]).

The recticlinear lens distortion model assumes that light reaching the sensor through the lens is subjected to both radial and tangential distortions. In this model, the distorted position of viewed images on the camera sensor is defined by the equation

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = \begin{bmatrix} 1 + k_1 r^2 + k_2 r^4 + k_3 r^6 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} 2p_1 x_n y_n + p^2 \left(r^2 + 2x_n^2 \right) \\ p^2 \left(r^2 + 2y_n^2 \right) + 2p_2 x_n y_n \end{bmatrix}, \quad \text{(B.1)}$$

where $r = \sqrt{x_n^2 + y_n^2}$, k_i are the radial distortion parameters, p_i are the tangential distortion parameters, and

$$\begin{bmatrix} x_n \\ y_n \end{bmatrix} = \begin{bmatrix} X_c / Z_c \\ Y_c / Z_c \end{bmatrix}$$

are the normalized coordinates in the frame of the camera.

Given distorted pixel positions, the intrinsics matrix $\mathbf{M}_i \in \mathbb{R}^{3\times 3}$ relates to the actual pixel coordinates by

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & s \end{bmatrix} \begin{bmatrix} x_d \\ y_d \\ 1 \end{bmatrix},$$

where f_x and f_y are focal lengths in x and y, c_x and c_y are the pixel coordinates of the center of perspective, and s is a shear parameter that is often set to 0 in practical calculations.

Fitting the parameters in these equations is sufficient to calibrate the relationship between the 3D camera coordinates and the 2D image coordinates. It is constant for a fixed optical setup regardless of the orientation of the observer. Calibration with respect to some object or plane in the physical world requires fitting an extrinsic matrix comprised of a rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times3}$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$ such that $\mathbf{M}_e \in \mathbb{R}^{3\times4} = [\mathbf{R}\,\mathbf{t}]$. Then, the fully defined camera matrix is computed $\mathbf{P} = \mathbf{M}_i\mathbf{M}_e$.

Knowing extrinsic parameters are not always necessary when calibrating a camera, but they are required for flow measurement. This is because they effectively situate the camera in 3D space with respect to the flow. In the calibration procedure implemented in this work, the extrinsic matrix is implicitly calibrated in the form of a homography $\mathbf{H}$ which maps one planar projection of an image to another by a linear transformation:

$$\begin{bmatrix} x'_p \\ y'_p \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix}, \tag{B.2}$$

where x'_p and y'_p are the image coordinates from another perspective. In the flow measurement context, the homography accounts for misalignment of the camera to the flow and dimensionalization according to the calibration target.

ChArUco Board Calibration

ChArUco calibration extends standard checkerboard calibration techniques to allow for circumstances where only incomplete checkerboards are visible. ArUco fiducial markers—a type of ARTag which was devised for the purpose of 3D registration and pose tracking in augmented reality applications—are placed inside the white spaces on the checkerboard pattern to enable unique determination of the board location even if portions of it are off screen. Each ArUco marker consists of a black border with an $n \times n$ binary matrix of pixels

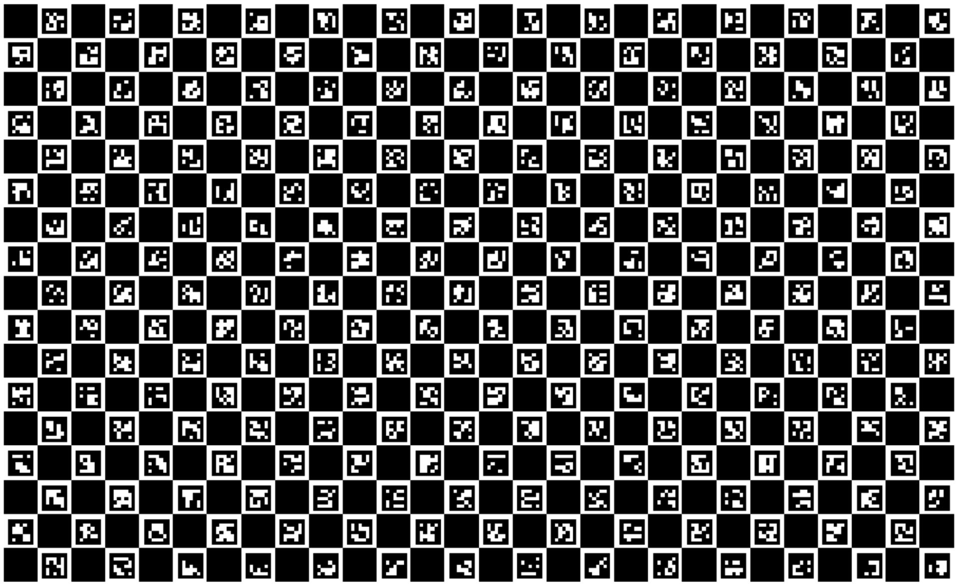


Figure B.2: A computer generated image of the large ChArUco board used for calibration in LGR experiments conducted for this work. The board is 28×19 squares with 5×5 ArUco markers inside the white spaces.

inscribed that determines a unique identifier. Therefore, by finding any of the markers, the position of the entire checkerboard is known. An example of a ChArUco board is provided in Figure B.2.

The specific process of calibrating a camera for flow measurement using a ChArUco board is provided below:

1. Prepare the camera to take images of the flow. This involves focusing on the correct measurement plane, fixing the aperture, and setting the camera sensitivity and shutterspeed. While sensitivity and shutterspeed can be adjusted after calibration, everything else should remain fixed.
2. Calibrate the intrinsic matrix. Without making any adjustments to the lens, take many images $\mathcal{O}(50-100)$ of the ChArUco board. Vary the angle and distance from the lens between images for better results.
3. Fit the homography between the camera's true position and the dimensional position directly normal to the ChArUco board. This step only requires a single image of the ChArUco board placed in the plane of the flow.

Figure B.3: Using Segment Anything Model (SAM, [155]) to identify and mask semantic objects within an image. Top: An image of Beckman Auditorium at Caltech. Bottom: A sample image from the experimental data presented in Chapter 5. Left: Original images. Right: SAM used to identify all semantic masks in an image based on the online demo implementation provided at https://segment-anything.com/

A python script built on the OpenCV library was written to enable ChArUco calibration for the experiments done in this thesis.

B.3 Training Data Collection

In Chapter 5, a process was described for identifying flow features using the imperfect which may be observed in field studies. The method relied upon the success of deep detection models to be able to identify flow tracers in the natural setting, and therefore required high-quality training data to fine-tune the models. In particular, the training procedure required images with masks delineated around all tracers contained. For the data considered, there may be hundreds or even thousands of particles contained in a frame; drawing masks on individual tracers would be far too time-consuming to be practical (even for a graduate student!).

In order to facilitate the mask generation procedure, the Segment Anything Model (SAM) developed by Facebook AI Research (FAIR) was utilized [155].

The SAM model is a model based on visual transformers (ViT, [186]) and, after encoding an image, is able to identify all semantic segmentations that it contains without need for labels (see Figure B.3). In other words, it identifies all sets of pixels in an image that, with high probability, represent an object in physical reality. While very effective at segmentation, the SAM model is not well suited for object detection (at least, at the time of this writing). Initialization of images can take 10 or more seconds, and the structure of the network is not designed to idenfity class instances.

In order to leverage the capabilities of SAM, an in-house GUI app was built to collect the necessary training data. Directly applying SAM to raw images of particles was ineffectual, since SAM does not automatically output mask data to a trainable format and it often misses or incorrectly segments tracers without additional supervision (see Figure B.3). The GUI app was built to allow the user to load images or videos containing objects to be masked and click them independently to generate masks. Clicking the object generates a mask around it and prepares the mask data in a JSON file in COCO format [187] with the addition of the mask centroid. Using the app, sets of training images sufficient for fine-tuning detection models can be quickly generated with little effort.

Two sets of training images were collected for the experiments conducted during this work. In the laboratory experiments, training data was collected to identify the spheres, rods, and bees wax pellets used as surface tracers in NOAH. For the field data taken at the Caltech turtle ponds, bubbles, leaves, and other surface debris were highlighted as the tracers to detect. Example training images from both of these data sets are provided in Figure B.4.

B.4 Asymetric Wake Past a Blunt Trailing Edge

Since the asymetric wake experiment is featured in multiple places througout the body of this work and in different capacities, its details are included here rather than inline with the narrative of the dissertation. Specifically, the experiment features in Chapter 4, where particles are numerically seeded and propagated on PIV velocity fields and in Chapter 5 where deep detection and tracking methods are demonstrated with miscellaneous wooden debris. The background and details of this experiment are provided below.

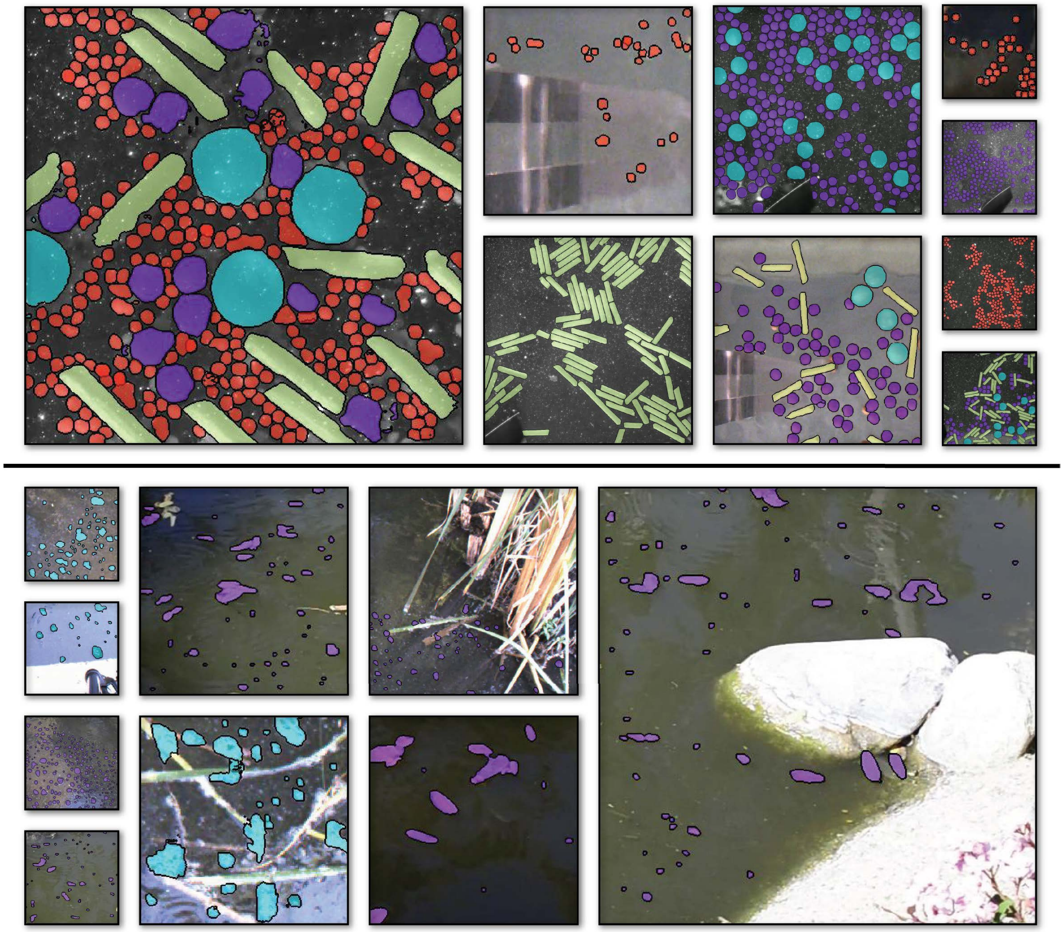


Figure B.4: Sample training images used to train the detector models employed in Chapter 5. The images on the top half of the figure were used to train the model used in the laboratory and include small and large spheres, rods, and bees wax pellets. The images on the bottom half were used to train the turtle pond detector and include masks of floating debris and bubbles.

Background

The original iteration of the asymmetric wake experiment was conducted by Tian et al. in 2012 [188] at Caltech to study stability characteristics of geometries associated with injectors to combustion chambers. The experiment studies the behavior of a split flow as it passes a blunt trailing edge with a finite thickness. A schematic of the flow is provided in Figure B.5. The experiment was designed to allow for various velocity ratios $\frac{U_2}{U_1}$ of the flow on either side of the wall to be studied. In all cases, the flow immediately downstream of the trailing edge is characterized by bluff-body vortex shedding. For velocity ratios $\frac{U_2}{U_1} < 1$, however, a shear layer develops as the flow progresses downstream.

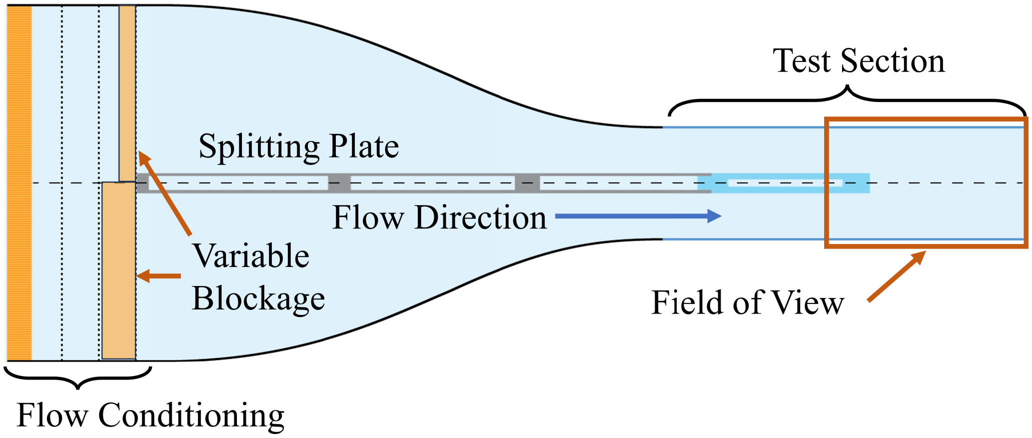


Figure B.5: Top-down schematic of the asymmetric wake to shear layer experiment.

The dual nature of this flow is significant in its hydrodynamic stability profile—the bluff body shedding is an absolutely unstable phenomenon whereas the the shear layer is convectively unstable [189]. Which mode dominates is a function of the flow parameters and the downstream location from the trailing edge. Understanding these stability characteristics was the emphasis of work done by Quintanilha and Alves [190], where preliminary validation was performed using the 2012 experiment. This experiment was designed primarily to support that work using modern instrumentation. The results of the experiment and stability analysis are currently under preparation, but the data and setup has also served as a convenient case study for LGR, which was being developed in parallel. Photographs of the experimental configuration are provided in Figure B.6.

Experimental Setup

All experiments conducted on the asymmetric wake were performed in the NOAH water channel at GALCIT described in section B.1. Since the NOAH facility was not designed to handle split flow conditions, a removable plate was designed to sit inside the channel extending from the flow conditioning, through the contraction, into the test section. The plate was constructed in segments from aluminum plates and spacers, with the final segment made from acryllic panels to allow the laser sheet to pass through. The assembled plate was 0.0572 m thick and 3.66 m long. The plate was placed in the water channel so that it was centered horizontally throughout the test section and

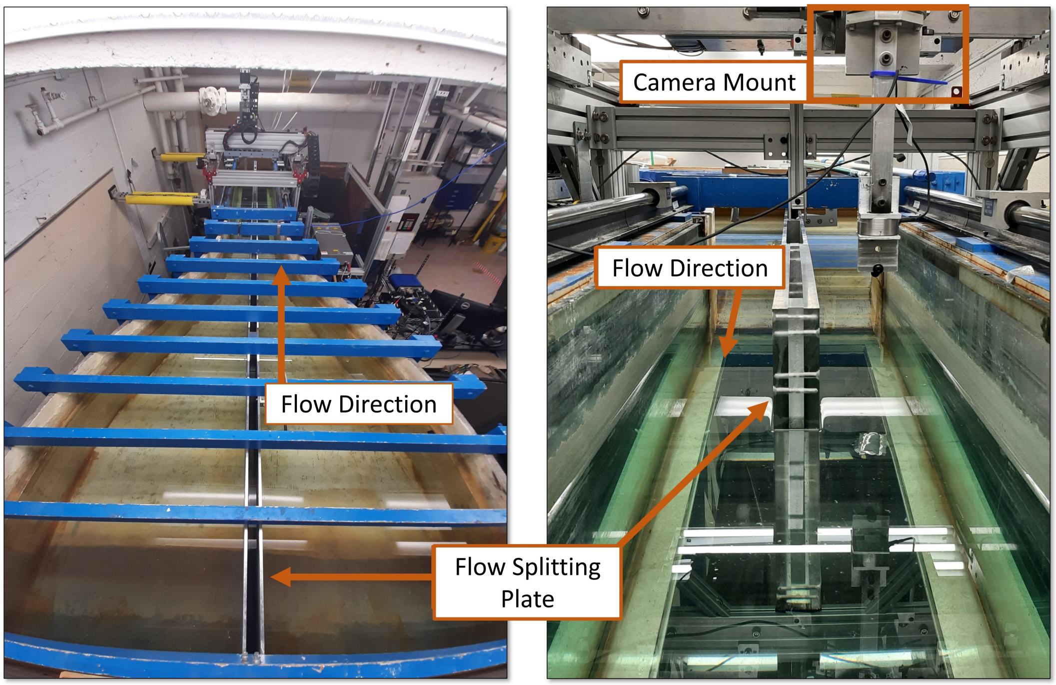


Figure B.6: Labeled photographs of the asymmetric wake to shear layer experiment setup.

contraction. Photographs of the plate inside the water channel are provided in Figure B.6.

Velocity ratios $\frac{U_2}{U_1}$ were enforced through the addition of various blockage materials just upstream of the leading edge of the plate. Blockage material included honeycomb panels (2.54 cm and 5.08 cm thicknesses), perforated aluminum plates (51% open area), wire meshes, and Organza fabric (similar to toole). Once a blockage configuration was assembled, a few short runs of PIV were performed just upstream of the trailing edge where the free stream on either side of the plate could be sampled. The PIV data sets were temporally averaged and used to assess the velocity ratio. The conditions desired for the stability analysis required that $\frac{U_2}{U_1} \approx 0.5$, which is where most of the data was collected.

PIV images were captured using two Phantom Miro Lab 320 cameras placed beneath the test section near the trailing edge of the plate. Various fields of view were examined during the experiments. The wake behind the trailing edge was imaged through Nikon AF Nikkor 50mm f/1.8D lenses where the fields-of-view were stitched together lengthwise and oriented so that the trailing edge of the plate was situated at the upstream edge of the collective field of view. The stability experiments also required that the momentum thickness of the boundary layer on either side of the plate be measured. Therefore, additional PIV images were taken just upstream of the trailing edge of the plate through Nikon AF Micro-Nikkor 200mm f/4D IF-ED lenses. While these were used to study the momentum thickness and velocity ratio during the experiment, the results are not presented in this thesis.

As previously mentioned, the asymmetric wake example served as a testbed for surface tracer tracking algorithms and LGR structure determination. To this effect, an assortment of various buoyant debris was obtained and placed upstream of the trailing edge of the splitting plate and allowed to flow into the vortices shed behind the blunt body. Debris acquired for this purpose included Birch spheres with diameters of 0.9525 cm (3/8 in) and 1.905 cm (3/4 in), birch rods with 0.645 cm (1/4 in) diameter and length ranging from 1.27 cm to 5.08 cm (1/2 to 2 in), and beeswax pellets with approximately 0.3175 cm (1/8 in) diameter. The density of Birch ranges from approximately 510 to 770 kg/m^3 and the density of beeswax is 961 kg/m^3.

Images of the buoyant flow tracers were recorded with the Teledyne/FLIR Blackfly USB3 color machine vision camera, which was affixed to the CTS mount above the free surface of the flow. A 4mm Edmund Optics UC series wide angle lens was used to image the entire wake of the flow at once.

Tested Conditions

Multiple experimental campaigns were completed using the asymmetric wake setup. The first campaign involved an exploratory sweep over a wider range of flow parameters and served to pinpoint conditions to be reconsidered in later iterations. During the first campaign, three Reynolds nubmers (based on the plate thickness) were targeted. These were $Re_t \in \{6000, 14000, 22000\}$. For each Reynolds number, multiple tests were conducted with varying blockage conditions. Ultimately, the goal was to achieve a velocity ratio as close to $U_2/U_1 = 0.5$ as possible. Six independent runs were collected per set of conditions. Averaged values from sample cases are provided in Table B.1.

One of the principal objectives of revisiting this experiment from the work of Tian et al. [188] was to sample lower frequencies of the flow. In particular, sampling needed to be performed such that the minimum nondimensionalized frequency was $\min(\omega_0) < 0.1$ such that $\omega_0\ \frac{2\pi f t}{U_{mean}}$ and f is measured frequency and t is the plate thickness. This meant sampling over longer durations. Since the Phantom cameras were limited to recording 3676 images, sampling rates

FS were selected to be as slow as possible and yet still provide decent PIV results. These values are also provided in Table B.1.

During the first campaign of experiments, the aluminum plate placed in the water channel to split the flow frequently corroded and needed to be cleaned. After the first round of data was collected, the plate was removed and treated with corrosion resistant paints in order to preserve its finish during further testing.

Once the plate was placed back in the NOAH water channel, additional data were collected. Thickness Reynolds nubmers of $Re_t \in \{15000, 24000\}$ were tested with blockage conditions attempting to enforce $U_2/U_1 = 0.5$. Six runs were recorded at each set of condtions. The average recorded conditions are provided in Table B.1. Conditions of the data presented in Section 4.2 of Chapter 4 are emphasized by bold font.

Since the flow-splitting plate was installed in the NOAH water channel for a large portion of the development of LGR, it was used to generate the laboratory test data presented in Section 5.3 of Chapter 5. These tests were performed with no upstream blockage, although the velocity ratio was visually observed to be $U_2/U_1 < 1$. The water channel was run at the same speed as those that generated $Re_t \approx 15000$. However, because the optical setup was separate from that used for PIV, exact values of velocity ratio and Re_t are not known. If necessary, those quantities could be estimated from the tracked tracer trajectories.

A variety of simulated surface debris was tested on the asymmetric wake flow. In particular, debris included two diameters of birch spheres (9.53 mm and 19.05 mm), 6.35 mm diameter birch rods cut between 12.7 and 50.8 mm, and beeswax pellets which were approximately spherical with diameter 3.175

Table B.1: Summary of flow conditions for the first and second experimental campaigns in the asymmetric wake flow experiment. The data presented in Chapter 4 was sampled from the records made with the bold conditions.

Campaign	Re_t	U_{mean} m/s	$\frac{U_2}{U_1}$ actual	FS Hz
1	6000	0.10	0.39	95
	14000	0.24	0.44	220
	22000	0.38	0.42	350
2	**15000**	**0.26**	**0.582**	**195**
	24000	0.40	0.570	280

mm. All combinations of the birch tracers were tested on the asymmetric wake flow, and the beeswax were tested separately. At least three runs of data were collected per surface tracer condition. In Section 5.3 of Chapter 5 two conditions were considered: a set of data using only the 9.53 mm spheres, and a set using all wooden debris.

Image Processing

PIV image processing was performed using LaVision DaVis, version 10.2.0. Pre-processing involved background subtraction followed by time resolved 2D2C PIV with a single pass of a 96×96 pixel circular window with 50% overlap and three passes of a 24×24 pixel adaptive window with 75% overlap. Vector post-processing was used to remove spurious vectors and the resulting empty cells were filled by interpolation.

Surface tracer images did not require additional processing prior to trajectory identification and computation of LGR.

Appendix C

ALGORITHMS

The success of the work contained in this thesis was contingent upon the development and software implementation of a variety of algorithms. In this appendix, details of those algorithms are provided. In the first section, the Lagrangian gradient regression (LGR) algorithms are discussed. The second section addresses the template matching approach taken for object tracking. While other algorithms were implemented throughout this work, these are the primary ones which were developed and implemented by the author.

C.1 The LGR Algorithm

LGR is implemented using the procedure outlined in Algorithm 1. In most practical applications, tracer trajectory information is provided as a list of indexed tracers with position histories recorded. The algorithm first re-orients the data to be indexed in time rather than by particle, so that nearest neighbors can be easily identified for the regression operation. The neighbors are then identified at each time step and for every particle, and their relative positions at time t_i and t_{i+1} are recorded. If neighbors do not exist at t_{i+1}, they are removed from the operation and the next viable neighbor is identified. Their

Algorithm 1 Lagrangian Gradient Regression (LGR) on Sparse Trajectory Data

Input: Indexed particle trajectories; number of neighbors n; kernel function $k(\Delta\mathbf{x}, \Delta\mathbf{x})$; regularization parameter γ.

Output: Velocity gradients recorded along particle trajectories.

```
$N \leftarrow$ number of snapshots recorded.
for each $t_i$, $\forall i \in \{0, 1, \ldots, N-1\}$ do
    for each tracer $\mathbf{x}(t_i) \in \mathcal{P}$ at $t_i$ do
        Find nearest neighbors $\mathbf{x}_j(t_i)$ of $\mathbf{x}(t_i)$, $\forall j \in \{0, 1, \ldots, n\}$.
        Compute kernel matrix $\mathbf{K}$ using $k(\Delta\mathbf{x}, \Delta\mathbf{x})$ with $\Delta\mathbf{x}_j = \mathbf{x}_j(t_i) - \mathbf{x}(t_i)$.
        Compute $D\mathbf{F}_{t_i}^{t_{i+1}}$ using equation 3.11.
        Compute $\nabla\mathbf{v}$ using equation 3.13.
    end for
end for
```

Algorithm 2 FTLE from Sparse Trajectory Data

```
Input: Indexed particle trajectories with $D\mathbf{F}_{t_i}^{t_{i+1}}$ available at each time step; observation time $\Delta t$.
Output: FTLE $\sigma_{t_i}^{t_i+\Delta t}$ along each trajectory.
$N \leftarrow$ number of snapshots recorded.
for each $t_i$, $\forall i \in \{0, 1, \ldots, N-1\}$ do          ▷ At each time step,
    for each tracer $\mathbf{x}(t_i) \in \mathcal{P}$ at $t_i$ do          ▷ along each trajectory,
        $s \leftarrow i$;   $t_n \leftarrow t_i$;   $D\mathbf{F}_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow \mathbf{I}_d$.
        while $t_n < t_i + \Delta t$ do          ▷ perform Jacobian composition.
            $D\mathbf{F}_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow D\mathbf{F}_{t_s}^{t_{s+1}}(\mathbf{x}(t_s)) D\mathbf{F}_{t_i}^{t_s}(\mathbf{x}(t_i))$
            $s \leftarrow s+1$;   $t_n \leftarrow t_s$
        end while
        $\Delta t_{true} \leftarrow t_n - t_i$;   $D\mathbf{F}_{t_i}^{t_i+\Delta t_{true}}(\mathbf{x}(t_i)) \leftarrow D\mathbf{F}_{t_i}^{t_s}(\mathbf{x}(t_i))$
        Compute $\sigma_{t_i}^{t_i+\Delta t_{true}}(\mathbf{x}(t_i))$ using equation 2.22.
    end for
end for
```

deviations from the analyzed particle are then stored in matrices for both times.

With the difference matrices in place, equation 3.11 is applied to compute the flow map Jacobian over the short interval, and equation 3.13 computes the velocity gradient. The result is stored with the tracer at time t_i.

Having computed the velocity gradient at each time step, instantaneous metrics can be computed along each trajectory. This is simply accomplished by following the appropriate formulae from Chapters 2 and 3. Computing finite-time gradients and the FTLE requires composition of gradients along a tracers trajectory, which is achieved using Algorithm 2.

Algorithm 2 receives indexed particle trajectories with position, time, and short-time flow map Jacobians stored at each temporal snapshot. The algorithm begins by iterating through each time step. At every instant, all trajectories existing at that time are entered into a second loop, where pre-calculated short-time Jacobians are left-multiplied until the desired time interval has been reached. Completing both loops will provide finite-time Jacobians for all available data. FTLE values are computed at each time step by applying equation 2.22.

Of the metrics discussed throughout this work, the only one which relies upon the finite-time flow map Jacobian $D\mathbf{F}_{t_i}^{t_n}$ is the FTLE. The rest of the finite-time metrics depend on integrated components of the velocity gradient. To approximate these other useful metrics, an approach typified in Algorithm

Algorithm 3 LAVD and DRA from Sparse Trajectory Data

Input: Indexed particle trajectories with $\nabla \mathbf{v}$ available at each time step; observation time Δt.

Output: LAVD $\mathrm{LAVD}_{t_i}^{t_i+\Delta t}$ and DRA $\varphi_{t_i}^{t_i+\Delta t}$ along each trajectory.

```
$N \leftarrow$ number of snapshots recorded.
for each $t_i$, $\forall i \in \{0, 1, \ldots, N-1\}$ do                    ▷ At each time step,
    for each tracer $\mathbf{x}(t_i) \in \mathcal{P}$ at $t_i$ do              ▷ along each trajectory,
        $s \leftarrow i$;  $t_n \leftarrow t_i$;  $\psi_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow 0$;  $\varphi_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow 0$.
        while $t_n < t_i + \Delta t$ do                     ▷ accumulate rotation angles.
            Compute vorticity $\omega(\mathbf{x}_s, t_s)$ by equation 2.14.
            Compute spatial averaged vorticity $\overline{\omega}(t_s)$.
            $\psi_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow \psi_{t_i}^{t_s}(\mathbf{x}(t_i)) + \frac{1}{2}\left|\omega(\mathbf{x}_s, t_s) - \overline{\omega}(t_s)\right|(t_s - t_i)$
            $\varphi_{t_i}^{t_s}(\mathbf{x}(t_i)) \leftarrow \varphi_{t_i}^{t_s}(\mathbf{x}(t_i)) - \frac{1}{2}\omega(\mathbf{x}_s, t_s) \cdot \mathbf{g}(\mathbf{x}(t_s), t_s)(t_s - t_i)$
            $s \leftarrow s + 1$;  $t_n \leftarrow t_s$
        end while
        $\Delta t_{true} \leftarrow t_n - t_i$;  $\mathrm{LAVD}_{t_i}^{t_i+\Delta t_{true}}(\mathbf{x}(t_i)) \leftarrow 2\psi_{t_i}^{t_s}(\mathbf{x}(t_i))$;  $\varphi_{t_i}^{t_i+\Delta t_{true}}(\mathbf{x}(t_i)) \leftarrow \varphi_{t_i}^{t_s}(\mathbf{x}(t_i))$
    end for
end for
```

3 might be applied. Here, LAVD and DRA are provided as examples for computing integrated quantities on sparse trajectory data. For each time step, identify all tracers which exist at that time step. Then, iterating through each of these trajectories, sum the necessary quantity until the desired time interval is reached and store the result at the initial time. While LAVD ($\psi_{t_0}^{t_n}$) and DRA ($\psi_{t_0}^{t_n}$) are presented in Algorithm 3, other integrated quantities like TISM and kinematic action are similarly computed.

C.2 Object Tracking by Template Matching

Object tracking factors heavily into the results and discussion of Chapter 5. Achieving high-quality gradients is contingent upon the ability to identify the trajectories of objects observed in the experimental data. Many algorithms have been developed to accomplish tracking tasks—some extremely simple, and others more complex. As the results discussed in this thesis are largely proof-of-concept, the chosen algorithm is on the simpler side of the spectrum.

Tracking by Template Matching

A template matching approach for tracer tracking is implemented in all experimental cases considered in this study, the basic mechanics of which are

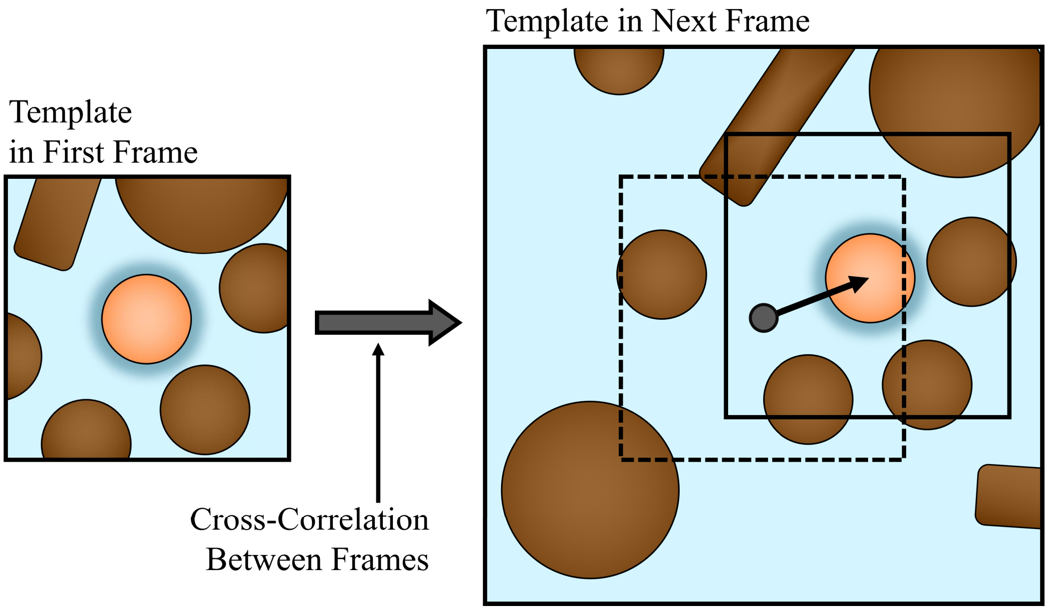


Figure C.1: Illustration of the template matching approach to object tracking. The template is defined as a window encompassing the observed object and a small neighborhood of its surroundings. This template is then correlated with a larger window in the subsequent frame. If the peak of the correlation corresponds to a detection, then the centroid of the new detection may be added to the examined trajectory.

illustrated in Figure C.1. First, the specific implementation of the algorithm assumes that detections have been made in all frames prior to the target association step. The centroids of these detections are provided as an input to the algorithm along with the images that contain them and various algorithmic parameters.

Algorithmic implementation begins by initializing an empty list of trajectories. In the first frame F_i, where $i = 0$, all centroids $\mathbf{c}_{i,j}$ of detected tracers are initialized as new trajectories T_k. For all trajectories ending at frame i a square window called a template is defined around the identified centroid $\mathbf{c}_{i,j}$ of the last detection in the trajectory with edges containing w_0 pixels. In the subsequent frame F_{i+1}, a larger search window with edge length $w_1 = w_0 + \Delta w$ centered at $\mathbf{c}_{i,j}$ is defined. Because the time step between images is assumed to be small, it is expected that, if the tracer continues to be visible in frame F_{i+1}, then it will not have moved far and should be contained within the search window. Additionally, as a result of the smoothness of trajectories, it is assumed that surrounding tracers and image features do not significantly change between snapshots. Thus, the expected centroid position in the subsequent frame $\mathbf{c}'_{i+1,j}$ is defined by the peak of the cross-correlation between the template and the search window.

Given the expected particle position in frame F_{i+1}, target association is achieved by identifying the centroid of the nearest-neighbor detection of the expected centroid position $\mathbf{c}'_{i+1,j}$. In certain cases, such as if the detector is known to have sufficiently high accuracy, it may be acceptable now to append the centroid of the nearest neighbor to the trajectory T_k. However, in circumstances where missed detections are expected, the position of the nearest neighbor should be within some nearness threshold θ_{near} of the expected position $\mathbf{c}'_{i+1,j}$. Additionally, the resulting trajectory should satisfy velocity θ_{vel} and acceleration θ_{acc} constraints defined by the user. Assuming these thresholds are met, the centroid of the nearest neighbor is appended to the trajectory. If it is not, then the analyzed trajectory is truncated and left alone for the remainder of this stage of the algorithm. New detections that do not receive association with existing trajectories are defined as the initial instance of a new trajectory and are similarly analyzed in subsequent frames. The algorithm concludes once all detections in all frames have been processed.

Post-Processing

The above algorithm is sufficient to track trajectories for use with LGR, but can be improved via post-processing. The need for enhancement arises from the nature of trajectories identified from mask centroids. In many instances, the mask generated by the detector is not perfectly consistent between frames. This creates artificial low-amplitude noise that can cause spurious results when additional analysis is performed. To overcome trajectory noise, median filtering is typically performed, followed by Gaussian smoothing. By replacing every entry with the median value from inside a local window, median filtering eliminates outliers from the trajectory. Gaussian smoothing further smooths the trajectory by softening normally distributed variations. For both filters, trajectory edges are padded with duplicates of the edge value when they are applied at trajectory boundaries.

Trajectory Merging

Trajectory merging was implemented to improve average trajectory quality and thereby LGR performance. The premise of merging hinges on the notion that many trajectories are shorted by the existence of a relatively small

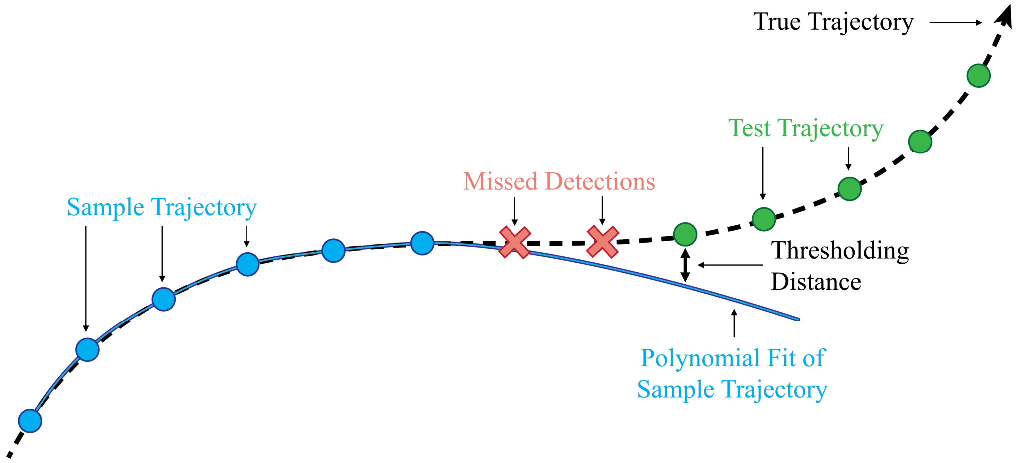


Figure C.2: Schematic depicting the trajectory merging algorithm. A polynomial is fit to a sample trajectory and extrapolated n time steps into the future. If, at any point, another trajectory begins within some nearness threshold of the extrapolated polynomial, then the missed detections are filled in by interpolation. If the resulting trajectory satisfies an acceleration threshold, it is kept.

number of missed detections randomly distributed over the duration of a true trajectory. Each missed detection splits a parent trajectory into two smaller trajectories with no overlap. To merge trajectories that are potentially split by this mechanism, each recorded trajectory is treated as a sample trajectory for which a polynomial of degree k is fit. The polynomial is extrapolated to all future time steps in $t_j \in [t_{i+1},\, t_{i+2},\, \ldots,\, t_{i+n}]$ for some integer n and for t_i the time instant of the last recorded snapshot in the sample trajectory. If, at any point, a test trajectory begins at time t_j within some specified distance threshold, then the potentially missed detections at times between t_i and t_j are interpolated by a user specified interpolation scheme. If the new additions satisfy an acceleration threshold, then the merged trajectory is accepted. This process is repeated for all trajectories until there are no more potential merges to be made. An illustration of the process undertaken for a single trajectory is presented in Figure C.2.

Avenues for Improvement

Improving tracking capability is, perhaps, the efficient means to increasing LGR accuracy and efficiency. As discussed, the algorithms highlighted above are relatively simple, and could be greatly improved for accuracy and computational cost. Here selected shortcomings of the above approach are briefly discussed along with possible improvements which could improve tracking quality with relatively low effort.

The first deficiency of the implemented tracking algorithm is that it requires all detections to be made in each frame prior to operation. If LGR is to be used as a structure detection strategy for autonomous mobile flow field measurement (AMFM) technologies, tracking must be possible in a real-time single-pass fashion. The above algorithms could be cast in that framework, where tracks are extended before detections are made in the new frame. However, all post-processing and trajectory merging would be unavailable in single-pass operation.

Tracking procedures may also be improved by causally linking detections and tracking. As it stands, detections and trajectories are formed independent of one another. However, many state of the art approaches in object detection and tracking inform where detections are made by the trajectories formed from their motion (see, for example, SORT [152] and DeepSORT [153]). Implementing a Kalman filter or a particle filter for state estimation of the trajectory centroid to inform where detections should be sought in future images would be a relatively straightforward algorithmic addition that could greatly improve trajectory quality and decrease computational cost.

A third improvement to the tracking strategy would be to eliminate the need for image content in target association. As defined above, windows of the original images are required so that cross-correlations can be performed to find the tracer in the next image. However, association could be done in such a way that does not require correlation data. Indeed, this is how most LPT algorithms work [19]. In fact, it is likely possible that tracking can be implemented in such a way that deformation gradients (short-time flow map Jacobians) are naturally produced as a byproduct. This algorithm would work by minimizing the regression of deformations between spatially proximal clouds of particles between subsequent frames. All that would be required from the detection stage would be the locations of centroids. In this case, detection, tracking, and gradient estimation could all be achieved in a single pass of the data, with great potential for reducing computational burden.

Of course, many other improvements may be made to the tracking approaches applied throughout this thesis. Those listed here represent those which seem to offer the greatest benefit for the least amount of innovation.