Adaptive Sampling in Particle Image Velocimetry

By

MATTHEW EDWARDS

Department of Aerospace Engineering
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Engineering.

JULY 2020

Word count: 50,000

Abstract

Particle Image Velocimetry (PIV) provides a way to analyse the instantaneous fluid displacement within some field of view, without influencing the underlying fluid. The displacement is measured by comparing two images of the fluid separated by a small timestep, extracting the observed displacement between the imaged particles through the use of cross-correlation windows. The size and locations (grid spacing) of such are traditionally user-defined and have a significant impact on the accuracy of displacement measurement.

To simplify the process, image analysis algorithms have, over time, reduced the number of controllable parameters through coupling to a small subset. This may accelerate the learning curve for users, yet greatly reduces the flexibility of the analysis process. Accordingly, optimising such parameters is user-dependent and often not possible due to unavoidable trade-offs between resolution and robustness.

The work presented in this thesis, therefore, investigates methods to maximise the extraction of information from PIV images, autonomously and efficiently. To achieve this, the research was focussed in three areas; reduction of unnecessary computation, improving the flexibility of analysis, and adaptivity, i.e. automatic decision making. The thesis presents methods for each theme, analysing their performance and implications for optimal, automatic, PIV image analysis.

The first approach is to optimise the existing analysis architecture, by preventing unnecessary computation based on local convergence of the displacement field, to achieve greater utilisation of computational resources. This is shown within to be an effective approach, reducing the number of correlations by almost half in typical conditions.

Alternatively, the analysis algorithm can be made *adaptive* such that it can self-select better analysis conditions. This approach hinges on the de-coupling of analysis parameters, which is found to raise many questions, regarding the optimal choice of such parameters, that are clarified and explored within. One such question is the initial window size, for which a novel algorithm is presented which automatically selects the locally optimal value.

Further to this, fully unstructured and semi-structured analysis algorithms are explored, with novel implementations presented in both cases. The semi-structured approach which is found to be simple, robust, and highly computationally efficient, while still providing much of the flexibility of fully unstructured, representing a promising avenue for future research.

Dedication and acknowledgements

It's done, it's over. It's actually complete. This PhD has been one hell of a journey, that's for sure; There's been ups, and my word have there been downs. I can't believe that I am actually writing the acknowledgements of my doctoral thesis, and, yet, here I am. It sounds extremely cliché but I honestly could not have achieved this on my own, and I would like to take this opportunity to express my gratitude - not just to the people mentioned in the following sentences, but to the many, many, others who have shaped my journery in one way, or another, that I could not hope to acknowledge in just a few short paragraphs.

The first and, in my eyes, most important person to mention is Emily, for without her I do not know how I would have achieved this. Finishing this thesis is as much of an achievement for her as it is for me. She has been my rock, my shoulder to lean on, and my soundboard to ramble at non-sensically for hours at a time. I truly am a very lucky man.

I must also thank Raf, for without him I would not have even started this journey. Your enthusiasm has always been infectious, right from the day we first met to talk about a possible PhD. Somewhat less clearly do I remember our nights in Busan and Lisbon... Truly, thank you for all that you have done for me over the years.

I feel that I have been extremely fortunate to have not one, but three great supervisors during my PhD. Chris, you have always been there to provide help and support, academically or otherwise, even before you took the post of main supervisor when Raf moved back to Belgium. I've never felt that you've had anything other than my best interests in mind, and from someone of your calibre, I could not feel more supported in my efforts. Dan, while we started this journey as peers, and end the journey with you as my supervisor, first and foremost I see you as a mate. Whether you realise it or not, your attitude towards life, work, and friendship, impact on others more than you know, myself included.

I'm also lucky to have started and finished this journey with one of my best mates, Ben. There's not enough I could say to acknowledge our friendship over the last eight years, other than I hope there's eight-y more. Maybe we're still imposters, but at least we've finished.

Of course, along the journey I made some new friends for life; Dibble (Must I use your first name, Robert?), Alex, and Will. You three have made the journey more fun than I could have hoped. Bikes, Beers, and mates truly are an amazing combination. Here's to all the fun we had, and to making it through the other side!

Although last, but by no means least, my family. I must thank my parents, aptly named ‘mum’ and ‘dad’, my sister, Rhian, and, of course, Chester, and Max (Yes, dogs do count as family). They have been there for me right from the *very* beginning and continue to provide more love, care and support than one person could expect in life.

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: MATTHEW EDWARDS DATE: 27/07/2020

PUBLICATIONS

This thesis has led to the following publications:

- M. Edwards and R. Theunissen, “Efficient, Robust, Distribution of correlation windows for unstructured adaptive piv sampling - Towards fully autonomous robust piv analysis,” *Proceedings, 19th International Symposium on the Application of Laser and Imaging Techniques to Fluid Mechanics*, *July 2018*.
- M. Edwards and R. Theunissen, “Adaptive incremental stippling for sample distribution in spatially adaptive PIV image analysis,” *Measurement Science and Technology*, *Mar 2019*, Vol. 30, no. 6.
- M. Edwards, R. Theunissen, C.B. Allen“On the feasibility of selective spatial analysis for temporal adaptivity based on confidence statistics,” *Proceedings, 13th International Symposium on Particle Image Velocimetry - ISPIV 2019*, *July 2019*.
- M. Edwards, R. Theunissen, C.B. Allen, D.J. Poole “On the feasibility of selective spatial correlation to accelerate convergence of PIV image analysis based on confidence statistics,” *Experiments in Fluids*, *Accepted pending final revisions*.

TABLE OF CONTENTS

List of Tables

List of Figures

Figure **Page**

NOMENCLATURE

α	Level of confidence as a percentage
δt	Time separation between images
Δz_0	Thickness of the light sheet
δ	Dirac delta function
λ	Expected number of particles in a window, based on seeding.
$\mathbf{s}$	Coordinates in correlation plane
$\mathbf{U}$	Velocity in object plane
$\mathbf{U}_p$	Velocity of a particle in the object plane
$\mathscr{F}$	Fast Fourier Transform
ν	Number of degrees of freedom, $N-1$
$\overline{u}$	Ensemble mean horizontal displacement
σ_u, σ_v	Population standard deviation of the displacement
τ	Point spread function of the lens
$\tilde{\sigma}_u, \tilde{\sigma}_v$	Local, spatial, standard deviation of the flow
d_p	Particle diameter
d_τ	Particle image diameter
h	Spacing between vectors
h_k	Spacing between vectors for the kth iteration, in pixels
I_A	The first image in a pair (discrete)
I_a	First image in pair (continuous)
I_B	The second image in a pair (discrete)
I_b	Second image in pair (continuous)
K	Total number of WIDIM iterations
k	Kernel size
N	Number of images in the ensemble
N_I	Number of particle images
N_p	Number of particles in the object plane
N_{eff}	The effective number of independent samples
NI_t	Target number of particle images
$p(x,y)$	Binary image indicating the location of particles
$P(x,y;m,n)$	Local particle count within an m x n area around (x,y)
$R(\mathbf{s})$	Correlation map
S_d	Seeding density in particles per pixel
$S_{u,N}$	Sample standard deviation for horizontal displacement
S_u, S_v	Sample standard deviation of the displacement

t	Critical value from the Student's t-distribution
t	Initial time
$t^{'}$	Initial time plus small delay
u, v	Displacement components
u,v	Displacement components in the image plane
U,V,W	Velocity components in the object plane
WS_0	Initial window size
WS_f	Final window size
x,y	Coordinate system in the image plane
X,Y,Z	Coordinate system in the object plane
z^*	Critical value from the standard normal distribution
AIS	adaptive incremental stippling
AIW	adaptive initial window sizing
AS	adaptive sampling
ASR	analysis support radius
BFS	backwards facing step
CDF	Cumulative density function
CFD	computational fluid dynamics
CI	Confidence interval
erf	error function
FFT	fast Fourier transform
FOV	field of view
HWA	hot-wire anemometer
KNN	K nearest neighbours
LDA	laser doppler anemometry
NMT	nomalised median threshold
PDF	Probability density function
PID	particle image distortion
PIP	particle image pattern
PIV	particle image velocimetry
ppp	particles per pixel
PTV	particle tracking velocimetry
ROI	region of interest
SFD	spring-force distribution
SNR	signal-to-noise ratio
TR	time resolved
WOR	window overlap ratio
WS	Window size
WS	window size
WT	wind tunnel

CHAPTER 1

INTRODUCTION

Aerodynamics has long been a topic of interest to humans, considering that the utilisation of aerodynamic forces, e.g. for sailing or windmills, dates back millennia. The explicit study of aerodynamics, however, is somewhat more recent (dating back *only* centuries) - typically being associated to the likes of Leonardo DaVinci in the late 15th century, or Sir Isaac Newton and Daniel Bernoulli in the early 18th century. Significant interest during the 19th century into heavier-than-air flight famously culminated in the first powered flight, by the Wright brothers, in 1903. Following this, the study of aerodynamics has continued to attract attention, leading to the start of commercial aviation around the early 1950s, which has continued to grow every subsequent decade, with over 100,000 commercial flights now being registered per day[1].

Owing to the significant financial interest and competition in commercial aviation, the ability to observe, model, and study aerodynamics has attracted a great deal of attention in recent years. Although the drastic rise in computational power over the last two decades has led to the emergence of computational fluid dynamics (CFD), experimental investigations are still required to develop models, to be used by CFD methods, which accurately portray the underlying physical behaviour. Particle image velocimetry (PIV) is one such experimental technique that allows for the non-intrusive measurement of fluid velocities. The fundamental principle of PIV is the imaging of illuminated particles,

[1] According to statistics from flight tracking website FlightRadar24 - https://www.flightradar24.com/data/statistics

suspended in a fluid, separated by some time delay. Small particles are introduced into the fluid of interest, which can then be recorded with high-speed cameras, with known lens magnification, to obtain two or more images separated by a known time interval. The displacement of particles between images can then be resolved, and, assuming the particles closely follow the fluid's motion, used as a heuristic for the fluids displacement.

Applications for this technology are virtually limitless, since it can be applied to any flow of interest provided it can be seeded and imaged. Traditionally the technique is used for aerodynamic investigations, i.e. developing understanding of the inherent fluid flow behaviour (turbulence, shockwaves, boundary layers, etc), or quantifying the aerodynamic properties of a particular object, e.g. an aerofoil, F1 car, or cyclist. Additionally, PIV finds itself being used to solve more and more real-world challenges, for example, in city planning to model heat transfer or ensure drainage performance, modelling arterial blood flow, or to investigate combustion. While PIV offers a means to get real, instantaneous, velocity measurements, it is also incredibly useful for the development and calibration of numerical methods such as CFD.

The experimental configuration of a PIV investigation is certainly not a simple operation. The experimentalist must ensure a great many aspects are considered, for example, optical access, sufficient illumination of the region of interest (ROI) (typically provided by a laser), synchronisation of illumination with the camera, appropriate configuration of the camera (zoom, focus, aperture), tracer particle choice, to name just a few. Data is then obtained in the form of a series of images which, ideally, are characterised by a dark background with many uniformly bright particles over the image. The images are then analysed to extract the displacement of particles from one image to the subsequent image, with sub-pixel accuracy.

Yet, analysis of the images requires the selection of an array of parameters, each with a complex, and often coupled, effect on the solution. These parameters are typically crude in their application and can be likened to a one-size-fits-all approach, thereby forcing the experimentalist to trade-off between reliability, accuracy, and computational cost. While typical values may exist for these parameters, optimising requires tedious trial-and-error from the experimentalist, where each trial may take many minutes, possibly hours depending on the type of investigation. Surely, a better approach, where parameters can be optimally self-selected and allowed to vary according to the conditions at hand, exists? This research, therefore, focusses on the ways to *maximise information extraction* from PIV images, moreover, such information extraction should be *autonomous*, i.e. *not dependent on user's experience or time*.

1.1 Motivations for PIV

It is worth distinguishing PIV from the range of experimental fluid measurement techniques that are available. Perhaps the most famous of flow measurement techniques is the Pitot tube, which provides a scalar value of the fluid velocity magnitude at a specific location in space. Pitot tubes are principally based upon Bernoulli's equation, (1.1); by measuring the total pressure at an inlet pointing directly into the flow, and the static pressure, at an inlet perpendicular to the flow, it is, therefore, possible to obtain the fluid velocity.

$$0.5\rho v^2 + \rho g h + P = P_0 \tag{1.1}$$

Pitot tubes are simple and effective at providing immediate fluid velocity measurements, however, they have several shortfalls, primarily that the presence of the Pitot tube itself, and its structural support, may influence the fluid's behaviour in a non-negligible way. Besides, measurements are limited to a single location in space, and hence can not be easily used to develop an understanding of larger-scale fluid interaction. Furthermore, pitot tubes only measure the magnitude of the flow velocity parallel with the pitot tube opening, and therefore are less accurate in the case of cross-flow.

Improving on some of these issues somewhat is a hot-wire anemometer (HWA). The basic principle of a HWA is to measure variations in electrical resistance of a heated wire as the fluid of interest flows over it, cooling the wire in the process. These devices are very sensitive to changes in ambient temperature which makes them unsuitable for investigating steady flow behaviour, where the temperature may drift over the experiment, yet have an extremely high-frequency response and so are particularly desirable for the study of turbulence. As with pitot tubes, HWAs measure only the flow velocity magnitude.

Laser doppler anemometry (LDA), otherwise known as Laser Doppler Velocimetry, uses the Doppler shift effect of particles, acting as a heuristic for the flow itself, passing through an interrogating laser to obtain the velocity. These particles may be specifically introduced into the flow, or may be naturally occurring contaminants. The principle is relatively straightforward; two collimated (aligned) beams of laser light probe the fluid, and the doppler shifted reflections from particles passing through indicates the velocity. Compared to pitot tubes and HWAs, this method boasts being non-intrusive, and, when using three separate, orthogonal, laser devices, can obtain all three flow components. Despite these benefits, LDA remains limited to measuring only single locations at once.

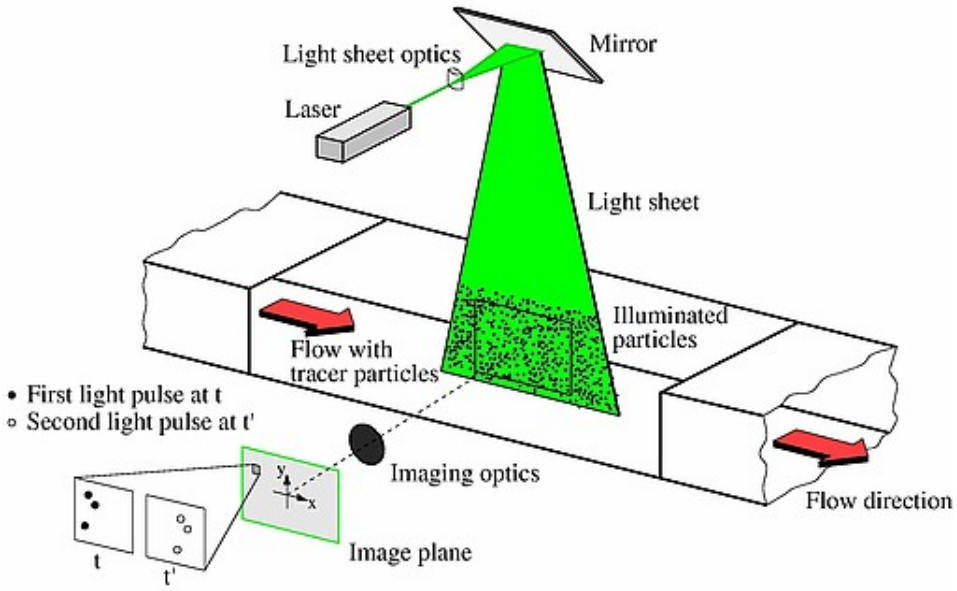


Figure 1.1: Diagram showing the principle of PIV. [1]

By comparison, PIV is non-intrusive, can measure all three velocity components, and can obtain simultaneous measurements from a range of spatial locations. PIV has a large range of experimental configurations, each with their advantages and disadvantages. These configurations range from standard PIV, which captures two-dimensional instantaneous displacement information in a single plane of the fluid, up to tomographic time-resolved which captures continuous[2] three-dimensional displacement over a three-dimensional region of the fluid. The various experimental configurations are discussed in more detail in Section 2.6.

Clearly, PIV offers a great tool to the experimentalist that allows for instantaneous, domain-wide, non-intrusive measurements of a fluid flow, yet all these approaches rely on accurate, and ideally timely, analysis of the resulting images.

1.2 Fundamentals of PIV

The fundamental principle of PIV is the imaging of illuminated particles separated by some time delay. Generally, these particles will have been deliberately introduced into the fluid sufficiently upstream to allow for good dispersal of these tracer particles within the fluid. The particles are convected, via stokes' drag, with the fluid motion and are

[2]Although still discrete snapshots of the flow, they are in quick enough succession to track specific particles for extended periods

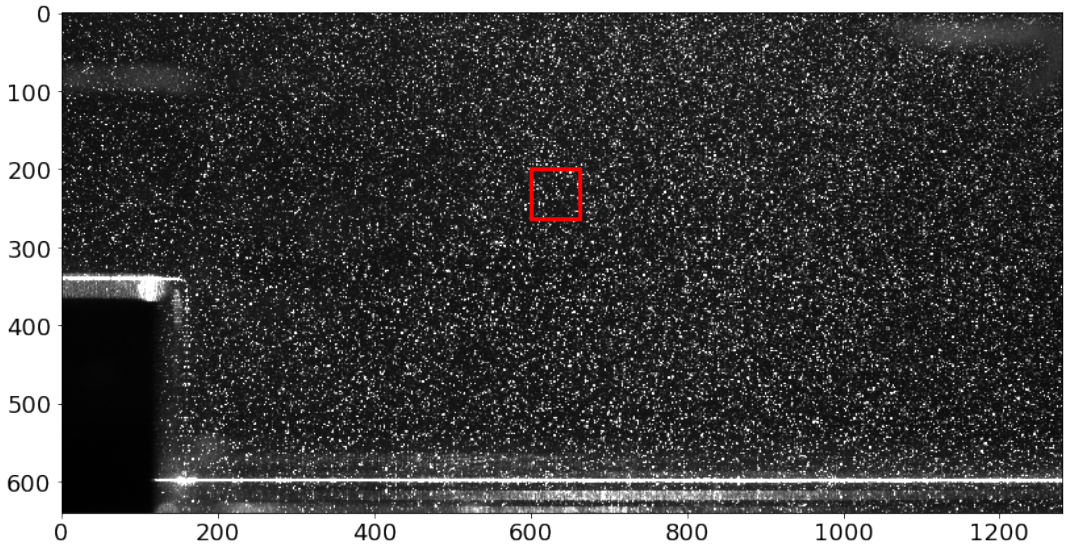


Figure 1.2: Exemplery PIV image showing the flow over a backwards facing step (BFS). Red square shows region for Figure 1.3

therefore assumed to act as heuristics for the fluid velocity. A pulsed light sheet, typically produced by a laser, illuminates the particles within a thin plane, which can then be recorded by high-speed cameras. This arrangement is demonstrated for a wind-tunnel application in Figure 1.1 and an exemplery resulting PIV image is shown in Figure 1.2.

Having recorded the images, the displacement of particles between successive snapshots must be resolved. This is achieved using a statistical operator known as the cross-correlation. The cross-correlation reveals the likelihood of each possible displacement of the particles in the first image, Figure 1.3(a), to the second image, Figure 1.3(b), resulting in a correlation map such as 1.3(c). By finding the location of the largest value in this correlation map, the *most probable* displacement is obtained for the defined region. For Figure 1.3 this corresponds to a horizontal displacement of ≈ 7px, where px is shorthand for pixels and will be used regularly throughout the thesis. This process is repeated for many of these, small, so-called interrogation windows so that a displacement field can be built up.

1.3 Motivation for current work

A local displacement estimate can be obtained for a small region of the image, and then agglomerated with many other local estimates to form the global displacement field. The displacement within the small region is obtained using cross-correlation, yet adequate

sizing of this region is challenging. Furthermore, the optimum size of this window is likely to vary depending on the location of the window, however, with traditional approaches this size must be fixed for all windows in the domain. Broadly speaking, the user is faced with a choice between robustness and resolution.

This dependence on the user and the parameter selections they make, can impact the solution in a significant way [2, 3] and therefore, the user's input is sought to be minimised. Expanding on this, in an ideal world there should be one, and only one, true solution to the displacement of particles between PIV images. In reality, unavoidable measurement uncertainties prevent the exact recovery of such solution, however, the philosophy that the solution, beyond these uncertainties, is independent of external influences, such as user input or random chance, holds true.

Since a single correlation window is capable of producing only a single displacement vector, an array of correlation windows are positioned over the image to build up a displacement field. Typically, this array is a cartesian grid with spacing obtained as a fraction of the uniformly chosen window size. Firstly, this strategy introduces an undesirable inherent coupling between the window size and sampling frequency. Yet furthermore, it is well agreed upon that regions of greater turbulence and fluctuations should be sampled more frequently than their more uniform counterparts; with the aforementioned gridding strategy, the sampling frequency remains fixed over the image.

It is therefore desirable to be able to automatically calculate the optimum window size, and to allow the size to vary over the domain. Moreover, by allowing the spacing between windows to vary, according to the underlying flow, it may be possible to improve

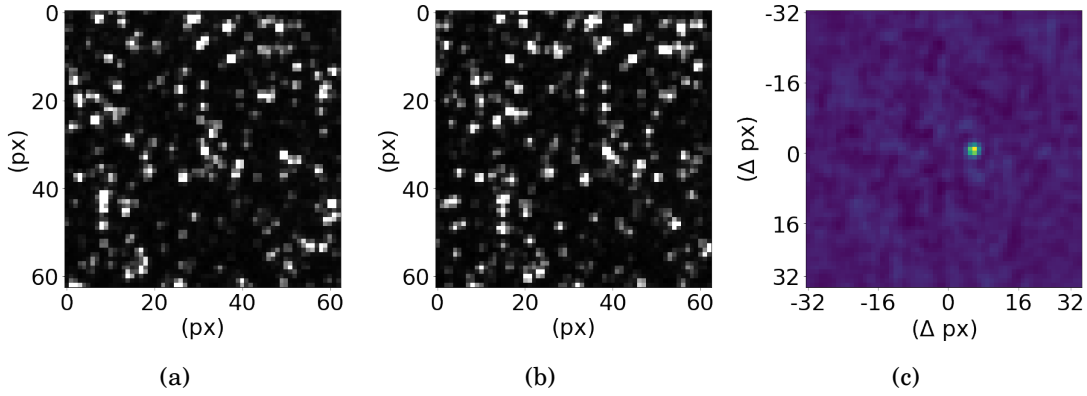


Figure 1.3: Exemplery cross-correlation window, showing (a) the first image, (b) the second image, and (c) the resulting cross-correlation between wsa and wsb for the region defined in Figure 1.2.

the solution accuracy in regions requiring higher fidelity, without compromising on the computational cost that would be associated with a globally more intensive analysis. The research, presented in this thesis, has investigated these ideas, with a focus on maximising the information extracted from PIV images.

Principally, this research investigates the autonomous (or semi-autonomous) analysis of PIV images, reducing un-necessary trade-offs where possible, allowing for maximum, reliable, data extraction which is independent of user-input. It will become clear, that in order to achieve this goal computational efficiency is vital. Beyond these objectives, any proposed method must also remain as simple as possible to aid in the uptake within the community and this is stressed throughout the research.

1.4 Thesis and research outline

In this introductory chapter, the motivations for PIV have been established, the fundamental setup of a PIV experiment has been described, and the motivations for the current work presented. That is, the desirable ability to analyse PIV images in a manner that allows for maximum information extraction, without being hindered by user-experience, parameter-coupling, or non-homogeneous conditions. The outline of the remainder of the thesis is as follows.

The following chapter is broken down into two key sections, first, technical background of the operational principle of PIV is presented, covering the main implications of the hardware used and a brief discussion on how displacement estimates are obtained from PIV images. The second half discusses the various components which constitute a conventional PIV image analysis routine, starting with the mathematics and implementation of obtaining a single displacement vector, and building to describe the current state-of-the-art for structured PIV image analysis.

Chapter 3 then presents a standalone method which significantly reduces the total number of correlations required to obtain an ensemble solution, by analysing only the regions which do not satisfy a statistical definition of convergence. By reducing the computational requirements in this way, more intensive analysis parameters may be used enabling greater information extraction in the same period of time. A strength of this method is that it can be incorporated into a conventional PIV image analysis routine with ease, and is shown to reduce the total number of correlations by 44% under typical conditions. This approach boasts being a very simple way to significantly increase information extraction.

The approach presented in Chapter 3, however, remains limited to the somewhat undesirable homogeneous sampling mentioned previously. To mitigate this, more flexibility is required in the sampling process. The remaining chapters, therefore, focus primarily on the explicit use of adaptivity within PIV image analysis routines, to address the desires to reduce user input and spatially adapt sampling conditions to the image. In Chapter 4, current adaptive methods are identified and the various challenges associated with an adaptive analysis algorithm explored. Addressing two of such challenges, two novel works are also presented, firstly, a method which improves the robustness of seeding density estimation is introduced. Following which, another new algorithm that allows for a robust initial window size to be determined, without any user-input or *a priori* displacement information, is shown and validated using synthetic PIV images. Notably, this chapter highlights the complexities of an adaptive sampling approach, in particular the issue of sample distribution.

A particularly crucial challenge for unstructured adaptive analysis routines is the ability to distribute samples according to a desired objective function. Fully unstructured distribution methods, for PIV image analysis, are, therefore, explored in Chapter 5, identifying the existing options and providing amendments to one, notably improving the robustness. Despite the improvements, the existing methods are deemed to either produce inferior distributions, which degrade the solution or require excessive computational resources. Accordingly, a novel method is presented, adapted form a computer graphics application, which produces sampling distributions with provably satisfactory quality, yet remains computationally efficient.

Throughout the aforementioned two chapters, challenges with the unstructured nature of the sampling distributions are repeatedly encountered. Furthermore, the necessary computational restrictions impose a degree of randomness to the analysis process, undesirably allowing the solution of a PIV image analysis of a specific pair of images to vary for the same parameter selection. Chapter 6 therefore explores whether a more deterministic, semi-structured, approach may resolve many of the challenges that arise from the jump from structured to unstructured. It is first considered as an alternative standalone distribution method that could replace one of the distribution methods discussed in Chapter 5, and be used in an existing adaptive sampling routine, before a novel adaptive analysis architecture is explored.

Finally, a range of conclusions and suggestions for future work are presented in Chapter 7, as well as an analysis of the research outcomes and how well the research objectives have been met.

CHAPTER 2

PIV: TECHNICAL BACKGROUND

In the previous chapter, the operational principle of particle image velocimetry (PIV) was introduced, along with the motivations for its use i.e. that it provides unintrusive, whole field, directional measurements of a fluid's velocity. This chapter provides a thorough background of PIV and is broadly split into two sections; First, a brief overview of the experimental configuration is given, discussing seeding, illumination, and imaging. Following this, a detailed overview of correlation-based image processing is provided, including image preparation, estimation of displacement from a single window, vector validation, and, finally, how such techniques form part of the larger analysis routine. During this chapter, the areas and motivations for this research will be highlighted, and a summary of such provided at the end.

2.1 Notation

It is helpful at this point to establish various notation conventions within PIV. Figure 2.1 shows the orientation of the axes in the object and image plane. A notable convention is the distinction of object plane from the image plane, whereby, the object plane is denoted by (X,Y), with velocities (U,V), while the image plane is associated with (x,y), with displacements (u,v). Moreover, Z_0, is used to denote the distance from the object plane to the lens, while z_0 denotes the distance from the lens to the image plane. Along these lines, Δz_0 is used to denote the light sheet thickness.

In the figure, a particle is shown moving from $\mathbf{X_t}$ to $\mathbf{X_{t'}}$, where $\mathbf{X} = [X,Y,Z]^T$, over a

time of Δt. Accordingly, this is imaged onto the sensor array where the particle moves from $\mathbf{x_t}$ to $\mathbf{x_{t'}}$, where $\mathbf{x} = [x, y, 0]^T$. This illustrates an important limitation of standard planar PIV, in that only the two-dimensional projection of the displacement is captured. It is also worth noting that in the case that either $\mathbf{X_t}$ or $\mathbf{X_{t'}}$ is not within the light sheet, it will not appear on the image sensor.

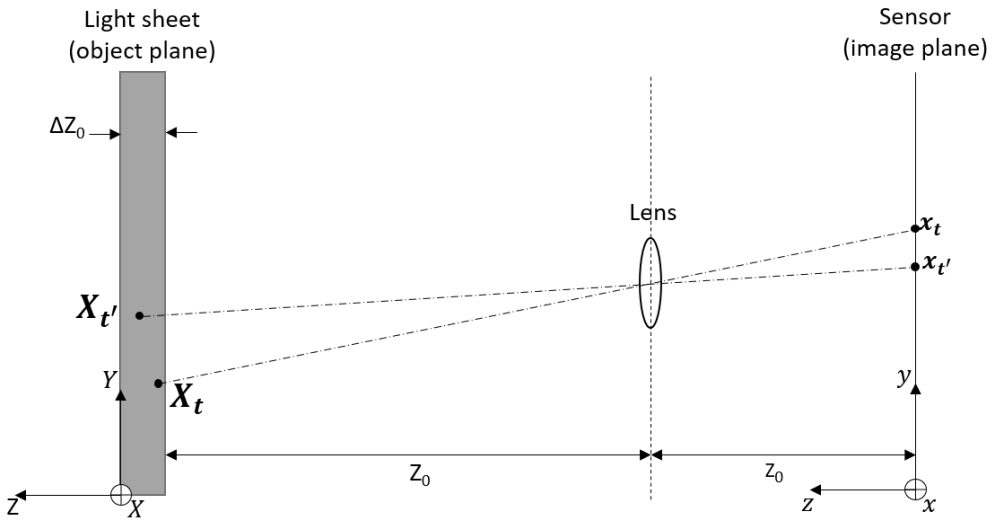


Figure 2.1: Illustration of the difference between object plane and image plane.

2.2 Seeding

PIV image analysis fundamentally resolves the displacement of the tracer particles suspended within the fluid, acting as a heuristic for the fluid's displacement, not the fluid itself. The ideal tracer particle should, therefore, perfectly following the surrounding medium to give the best results. The motion of small particles in a fluid is dominated by Stokes' drag [4], which allows the velocity lag of a particle under constant acceleration, $\mathbf{a}$, to be defined by (2.1), where d_p is the diameter of the particle, ρ_p and ρ are the densities of the particle and the fluid, respectively, and μ is the dynamic viscosity of the fluid. Note that $\mathbf{u}_p$ is the velocity of the particle, whereas $\mathbf{u}$ is the velocity of the fluid medium. From this equation it follows that, since the particle is likely to be of different density to the fluid, d_p should be minimised for the particle to follow the fluid as closely as possible.

$$\mathbf{U}_p - \mathbf{U} = d_p^2 \frac{(\rho_p - \rho)}{18\mu} \mathbf{a} \tag{2.1}$$

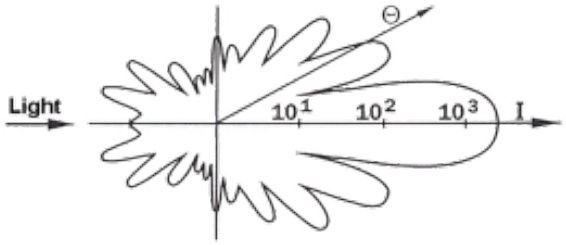


Figure 2.2: Diagram showing the scattering pattern (MIE) of PIV particles.

In contrast with this desire for smaller particles, for better fluid tracing, is the need for good light scattering behaviour which improves with increasing particle size. Light scattering behaviour when the wavelength of light is similar to the particle size can be characterised by Mie scattering, depicted in Figure 2.2. Under Mie scattering, the amount of light scattered depends on the relative refractive index of the particle to the fluid, the particle diameter squared, and the observation angle [1]. Subsequently, for a given angle and refractive index, Mie scattering dictates that the particle size be maximised to scatter the most light.

The choice of particle plays an important role in the characteristics of a PIV investigation, and depends on the properties of the fluid being used. Gaseous flows are typically seeded with oil or di-ethyl-hexyl-sebacate (dehs), that has been atomised using a Laskin nozzle, with diameters in the region of 1μm. Smoke (<1μm), helium filled soap bubbles (1000-3000μm), and various metal oxides (0.2-5μm) are also commonly used. For liquid flows, solid tracer particles such as polystyrene (100μm), hollow glass spheres (10-100μm), or metal oxides (3-5μm) are typically used. Fundamentally, any object which accurately follows a fluid's motion can be used as a tracer particle; for example Hong *et al*.used natural snowfall to reveal large flow structures behind a 2.5MW wind turbine [5].

A further key requirement for PIV experiments is homogeneous seeding within the test section. For liquid flow applications in re-circulating tunnels, this can be relatively easily achieved by allowing the fluid to mix prior to data collection, however, can cause challenges for gaseous flow applications in open-return facilities. Finally, another notable challenge is the deposition of tracing particles on the optical access; while solid particle deposition mainly only causes light beam attenuation, liquid particle deposition can accumulate to form non-uniform films which may further refract or distort incident light, producing optical aberrations [4].

2.3 Illumination

Accurate measurement of particle displacement relies directly on the ability to record such particles, which implies that they must be visible and, in order to obtain the particle's precise location, they should be stationary in the image. To achieve this, illumination takes the form of a short pulse, typically in the region of $\delta t \approx 5 - 10$ns. However, the brevity of this pulse reduces the integral of light incident on the image recording device, therefore demanding high energy light sources to improve particle brightness. The need for short bursts of high intensity light is where lasers are particularly of use.

The most common laser material found in PIV applications is Nd:YAG which stands for "Neodymium-doped yttrium aluminium garnet". Nd:YAG lasers are chosen as they have greater efficiency and can be constructed with more compact electronics [6]. Such compact electronics allow for multiple lasers to be housed in the same unit, allowing for δt to be arbitrarily small. They do, however, produce light at a wavelength, 1064nm, beyond the visible spectrum which requires additional crystals to bring the wavelength back down.

A further benefit of lasers is the ability to easily convert the laser beam into a thin light sheet using optics. Since a camera reduces three dimensional information onto a two dimensional image, out-of-plane motion cannot be resolved without further information, i.e. another camera's perspective. The thicker the light sheet, the more discrepancy there may be between the two dimensional projection and the true three dimensional displacement. Furthermore, if the light sheet is thicker than the focal depth of field then the light sheet will contain both in- and out-of focus particles, which can significantly deteriorate the performance of image analysis routines.

The separation time between pulses, δt, must also be carefully selected so as to minimise errors. The process for analysing PIV images carries a degree of error and uncertainty, which can be simplified to say is approximately uniform for a particular experimental configuration. The relative error, therefore, described by (2.2) where d_τ is the particle image diameter and c is a constant relating to the experimental conditions, benefits from large δt since $|\Delta \mathbf{X}| \approx \mathbf{u} M \delta t$, where M is the magnification and $\mathbf{u}$ is the fluid velocity. Additionally, the error due to poor fluid tracing must also be considered, which is approximately described by (2.3) and therefore benefits from smaller δt [7]. The origin of this error can be visualised by Figure 2.3 which shows that the observed particle displacement, $\Delta \mathbf{X}_p$, can be different from the particle's actual instantaneous velocity, $\mathbf{U}_p$,

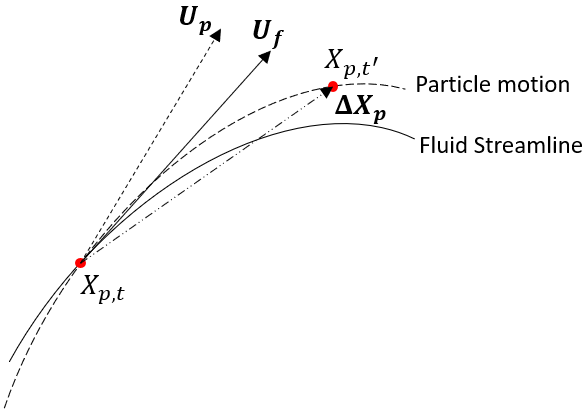


Figure 2.3: Diagram illustrating how the particle's motion may not exactly follow the fluid trajectory, and that acceleration of a streamline may not be captured well if δt, the delay between t and $t^{'}$, is too large, as described by (2.3).

which itself may also differ from the fluid's actual instantaneous velocity, $\mathbf{U}_f$.

$$\varepsilon_{random} = \frac{c d_\tau}{|\Delta \mathbf{X}|} \tag{2.2}$$

$$\varepsilon_{acceleration} \approx \left|\frac{\mathrm{d}\mathbf{u}}{\mathrm{d}t}\right| \frac{|\Delta \mathbf{X}|}{2M\mathbf{u}^2} \tag{2.3}$$

2.4 Imaging

When a point light source, i.e. a particle scattering light, is imaged onto a sensor, through a lens and an aperture, the resulting image no longer resembles a point, but forms a Fraunhofer diffraction pattern. The resulting pattern is mathematically described by an Airy disk surrounded by Airy rings, which can be closely approximated by a Gaussian function, as shown in Figure 2.4. The diameter of the Airy disk depends on the imaging configuration and represents the smallest particle image which can be obtained for that particular configuration.

The disk diameter, d_o, is the diameter at which the first zero of the Airy function is obtained, which is defined by (2.4).

$$d_o = 2.44(1+M)\lambda f_\# \tag{2.4}$$

Here, λ is the wavelength of light, M is the magnification factor, defined in (2.5) as the ratio of the distance between the image plane and the lens, z_o, and the distance from the

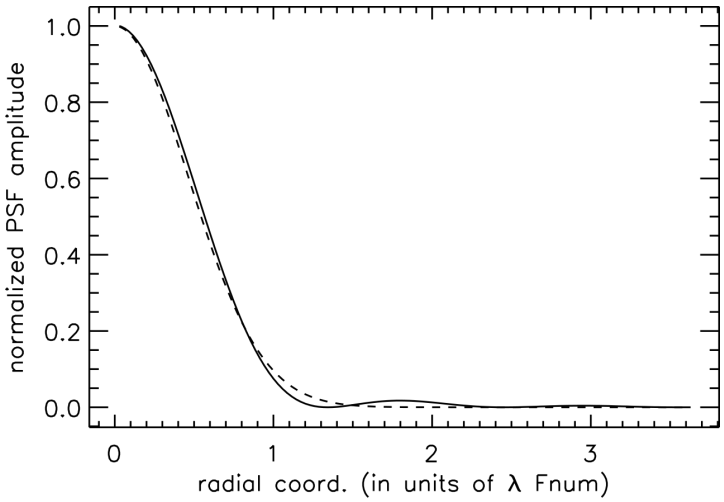


Figure 2.4: Diagram illustrating how the Airy function (solid line) is well approximated by a Gaussian (dashed line).

lens to the object plane, Z_o.

$$M = \frac{z_o}{Z_o} \tag{2.5}$$

Finally, $f_\#$ if the f-number of the lens which is defined as the ratio of the focal length, f, to the aperture diameter D_a.

$$f_\# = \frac{f}{D_a} \tag{2.6}$$

The particle image diameter, d_τ, can then defined as (2.7).

$$d_\tau = \sqrt{(Md_p)^2 + d_o^2} \tag{2.7}$$

Where d_p is the particle diameter, and thus Md_p is the geometric image. In the case for PIV, imaging is diffraction limited and thus $d_\tau \to d_o$.

Limitations in the shutter speeds of early image capture devices meant that PIV images were obtained as a doubly exposed image. That is, a single image frame contains the particle images for both t and $t + \delta t$. The displacement magnitude and angle could then be extracted by searching for peaks in the autocorrelation map. However, both the forward and reverse displacements would appear, with the correct direction unable to be identified without *a priori* information about the flow [8].

With higher frame-rates, modern implementations take singly exposed images such that a cross-correlation can return the displacement without directional ambiguity. For 30Hz, the time between successive snapshot pairs is 30ms, and thus conventional PIV recordings represent a series of independent, instantaneous recordings of the fluid flow. This is an important observation as it limits the user to quasi-steady flows where only

a mean flow, or ensemble statistics such as standard deviation or skewness, can be extracted.

Despite this, high frequency cameras with acquisition rates in the kHz ranges are now available, allowing for time-resolved PIV and the study of unsteady flows, though these are often complementary metal oxide semiconductor (CMOS). Early 'modern' cameras were CMOS sensors which were subsequently replaced by a tendency towards the use of charge coupled device sensors, which typically produce images of a slightly superior quality [9]. At the extreme, current CCD sensors have resolutions up to 50MP, however the frame rates at these resolutions are severely limited to around 5fps. Typically, resolutions in the 1-4MP range are common with frame rates around 30Hz. Due to the higher framerates achievable, CMOS sensors are still the camera of choice for high speed experiments.

2.4.1 Mathematical description

A particle within the illumination volume will diffract a certain amount of light onto the camera sensor at a particular location. The geometric image of a particle at small magnifications can be assumed to be represented by the Dirac delta function, $\delta(\mathbf{x}-\mathbf{x_i})$, for some particle image location $\mathbf{x_i}$. The point spread function of the camera lens is denoted by $\tau(\mathbf{x})$, and is commonly assumed to be gaussian in both x and y for typical lenses[1]. Therefore, the image intensity field can be written as the sum of intensity contributions from all particles within the volume, i.e.

$$I_a(\mathbf{x}) = \tau(\mathbf{x}) * \sum_{i=1}^{N_p} I_0(\mathbf{x}_i)\delta(\mathbf{x}-\mathbf{x_i}) = \sum_{i}^{N_p} I_0(\mathbf{x}_i)\tau(\mathbf{x}-\mathbf{x}_i) \tag{2.8}$$

Where $I_a(\mathbf{x})$ is the image intensity for some location $\mathbf{x} = (x, y)$, given N_p particles with locations described by $\mathbf{x}_i = (x_i, y_i)$, each with a maximum intensity $I_0(\mathbf{x_i})$, which itself will also follow a Gaussian profile based on the light sheet intensity variation.

Now consider that each particle is displaced by an amount $\mathbf{d_i} = (\Delta x_i, \Delta y_i)$ between image a, at time t, and image b, at time $t + \delta t$, such that the new ensemble of particles are described by $\mathbf{x}'_i = \mathbf{x}_i + \mathbf{d_i}$. The image intensity field for the second image can therefore be written as:

$$I_b(\mathbf{x}) = \sum_{i}^{N_p} I_0(\mathbf{x}'_i)\tau(\mathbf{x}-\mathbf{x}'_i) = \sum_{i}^{N_p} I_0(\mathbf{x}'_i)\tau(\mathbf{x}-\mathbf{x}_i-\mathbf{d_i}) \tag{2.9}$$

2.4.2 Synthetic image generation

In order to test the performance of various image analysis algorithms, it is often required to know the expected underlying displacement, such that a measurement error can be obtained. On approach is to simultaneously capture information that is of higher quality than the method being tested, e.g. using a higher resolution camera. The spatial resolution is increased, and the displacement gradients in terms of pixels/pixel are reduced, resulting in lower error (see Section 2.13), therefore allowing the higher resolution version to act as a benchmark for the lower resolution version. Yet the image is still being analysed with the methodology, and hence, any systematic errors that are present in the analysis algorithm are likely to be missed. To address this issue, a different velocity measurement technique may adopted, e.g. laser doppler anemometry (LDA). The obvious disadvantage of this approach is the poor spatial properties, since we can now only compare the PIV solution for a specific location in space.

Alternatively, it is possible to *create* PIV images, with a known displacement field, that accurately mimic real conditions. In doing so, we can compare the measured solution to the actual, known, displacement field and obtain an error measurement, instead of just an estimate. For planar two-dimensional two-component PIV (see Section 2.6 for more on method classification), the process is straightforward; particles are randomly created over the field of view (FOV), and the recorded intensity is calculated as per (2.10). This calculation is simply (2.8) re-written to use readily available error function (erf), where the only additional term is f_r which is the pixel fill factor which is the proportion of light sensitive area per pixel and is often close to unity for scientific grade CCD cameras. Creation of images with a third dimension or displacement component requires more consideration of the geometry of the imaging setup, for which a detailed information can be found in [10].

$$I(\mathbf{x},\mathbf{y}) = \sum_{i}^{N_p} I_{p_i} \left(\operatorname{erf}\left(\sqrt{8}\frac{x - x_i + 0.5f_r}{d_\tau}\right) - \operatorname{erf}\left(\sqrt{8}\frac{x - x_i - 0.5f_r}{d_\tau}\right)\right) \left(\operatorname{erf}\left(\sqrt{8}\frac{y - y_i + 0.5f_r}{d_\tau}\right) - \operatorname{erf}\left(\sqrt{8}\frac{y - y_i - 0.5f_r}{d_\tau}\right)\right) \tag{2.10}$$

2.5 Image Processing

There are two common approaches for extracting particle image displacement information. The first, referred to as particle tracking velocimetry (PTV), is to identify

and measure the displacement of individual particles in isolation. The second, referred to as particle image velocimetry (PIV), is to identify and measure the average displacement of a small number of particles within some region. Tracking individual particles allows for maximising the spatial resolution for a given image, however, suffers from lack of robustness when identifying the precise location of the particle image centres. Conversely, averaging over a small region mitigates such robustness concerns at the expense of spatial filtering. Furthermore, although PTV may provide better spatial resolution for the same image, there are limits on the acceptable seeding density of particles which may act to reduce the attainable spatial resolution below that of a highly seeded PIV analysis. The following section presents a more detailed discussion of the pros and cons of each.

2.5.1 Single particle displacement evaluation (PTV)

Particle tracking velocimetry (PTV) attempts to pair up particle images in sequential snapshots, that represent the same physical particle. In doing so, the maximum spatial resolution attainable can be realised, since these tracers represent the only discrete sampling of the fluid continuum.

Before particles can be paired, they must first be located in the image using particle detection routines. While particles are relatively easily identified by human users, automating this process is challenging due to the variation in particle shape and intensity, presence of image noise, and reflections from stationary objects, amongst other issues. These challenges can be somewhat overcome with advanced detection techniques, albeit at the expense of computation.

Particle pairs are found by searching a radius, in the second image, around each detected particle in the first image; If a single detected particle is present in this search region then the pair is formed. On the other hand, if multiple or no particles are identified the pairing can not be deduced, as per Figure 2.5(a). Such ambiguity is common where the seeding density if high or the displacement is sufficiently large. In such circumstances, δt may be reduced to maximise particle pairing, at the expense of relative error (Section 2.3).

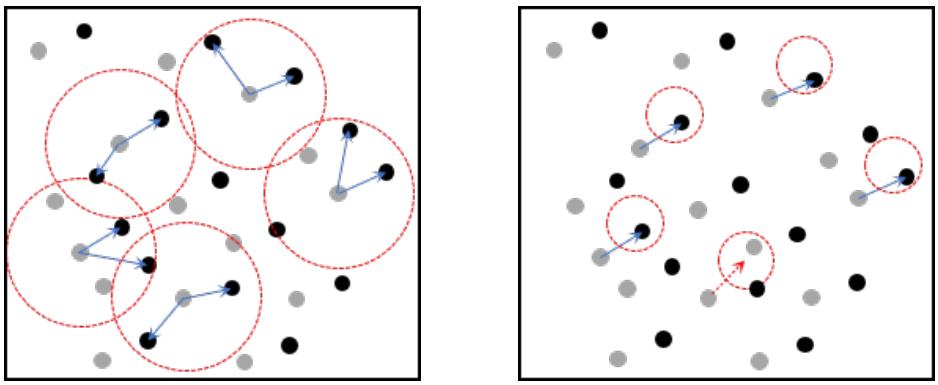
Hybrid PIV-PTV techniques can alleviate such ambiguity by minimising the search radius for particle image pairings, by first resolving the approximate displacement field using a coarse PIV based algorithm (See Section 2.5.2). Yet these potential improvements are dependent on the underlying PIV analysis - if it is well-tuned, the additional resolution that PTV might bring may no longer be significant to warrant the additional

computation, conversely, if the underlying PIV is poor then the reduction in search radius may be inadequate to notably aid the PTV step.

Finally, once the particles have been paired the exact displacement between the two must be resolved, which requires determining the precise centroid of the particle [11]. This step is known to possess considerable uncertainty and can hence degrade the solution, particularly if a small δt is adopted to reduce the displacement magnitude as per previously [2].

2.5.2 Multiple particle displacement evaluation (PIV)

Instead of operating on single particle images, particle image velocimetry (PIV) operates on a region containing many particle images, and attempts to extract the typical displacement within that region by cross-correlating the image intensity field and searching for the most overlap. Placing many of these interrogation regions over the image allows the displacement field over the entire domain to be constructed. By determining the displacement in this manner, as opposed to using PTV, a significant improvement in reliability is obtained, albeit at the expense of spatial resolution. This shortcoming of spatial resolution can, however, be mitigated almost completely through the use of iterative analysis algorithms, and hence PIV has become the method of choice for analysing these images.



(a) (b)

Figure 2.5: Illustration showing (a) ambiguity when pairing particles using PTV. Grey particles at time t, black particles are time $t+\delta t$. Red circles indicate the search region for the particle at $t+\delta t$. Blue arrows represent possible particle pairings, whereas red arrows indicate no candidate pairings found. (b) reduced search region, based on spatially averaged estimate of flow field, can reduce ambiguity, but may lead to missed pairings.

The cross-correlation operator reveals the amount of similarity between two signals, for various amounts of displacement of one signal relative to the other. The continuous cross-correlation for two images $I_a(\mathbf{x})$ and $I_b(\mathbf{x})$, valid for arbitrary size of region, is written using (2.11). The term $R(\mathbf{s})$ is often referred to as the correlation map, where $\mathbf{s}$ represents the coordinate system in this plane.

$$R(\mathbf{s}) = \int I_a(\mathbf{x}) I_b(\mathbf{x}+\mathbf{s}) \tag{2.11}$$

For reasons that will become clear later (see Section 2.10.3), the size of correlation region is typically chosen such that the number of contained particle images, N_I, is approximately 12-15, and $\mathbf{s_i} \approx \mathbf{s}$ for all $i = 1,2,...,N_I$. By substituting the equations for PIV image intensity, (2.8) and (2.9), the mathematical description for the cross-correlation of two regions of PIV images is shown in (2.12).

$$R(\mathbf{s}) = \sum_{i}^{N_p} \sum_{j}^{N_p} I_0(\mathbf{x}_i) I_0(\mathbf{x}'_j) \int \tau(\mathbf{x}-\mathbf{x}_i)\tau(\mathbf{x}-\mathbf{x}_j-\mathbf{d}+\mathbf{s}) \tag{2.12}$$

Contributions to R are made only when $\mathbf{x}_i = \mathbf{x}_j - \mathbf{d} + \mathbf{s}$, which suggests two scenarios;

1. $\mathbf{x}_i \neq \mathbf{x}_j$; In this situation, non-paired particles align randomly due to $\mathbf{x}_i = \mathbf{x}_j - \mathbf{d} + \mathbf{s}$. This contribution to the correlation plane is uniformly random and therefore results in evenly distributed noise over the correlation domain.

2. $\mathbf{x}_i = \mathbf{x}_j$; Contributions to the correlation plane, for $\mathbf{x}_i = \mathbf{x}_j$, can only occur where $\mathbf{d} = \mathbf{s}$. Since this is the same for all particles within the volume, a large peak will therefore exist at $\mathbf{s}$ in the correlation plane, representing the underlying displacement of the particles.

Provided that the correlation peak due to situation 2. is larger than any random peak caused by 1., the location of the largest peak in $R(\mathbf{s})$ will indicate the average displacement of all N_p particles within the interrogation region. However, cross-correlation remains a statistical operator and it is possible for the largest peak to not belong to the underlying signal, even for well-designed experiments [12]. To mitigate this possibility, vector validation schemes are employed which compare each vector to its surrounding neighbours, requiring a minimum degree of similarity. While detecting outliers may be simple, albeit very slow and time-consuming, for a human, the unambiguous detection using an automatic algorithm is challenging; too strict and valid

vectors are replaced, too tolerant and invalid vectors may corrupt the dataset. A more detailed discussion on cross correlation can be found in Section 2.11.

In practice, since the images described by (2.8) and (2.9) are stored and processed digitally, they are discretised onto a number of pixels, where the value of each pixel represents the integral of incident light. This discretisation allows use of the fast Fourier transform (FFT), utilising the convolution theorem (2.13), to greatly increase the efficiency of cross-correlation from O(n^2) to O($n\log n$).

$$R(\mathbf{s}) = \mathscr{F}^{-1}\{\mathscr{F}\{I_a\}\mathscr{F}\{I_b\}\} \tag{2.13}$$

One implication of the discretisation of image intensity values is that the correlation map can only be calculated for integer values of $\mathbf{s}$, resulting in very large relative errors for maximum displacement magnitudes in the region of 8-15px. Nonetheless, it is possible to estimate the sub-pixel correlation values by interpolating in the vicinity of the correlation peak. Section 2.10.2 describes this process in more detail.

2.6 Method classification

There are numerous ways within which PIV can be applied to an experimental situation, each capable of extracting different information from the flow region of interest. Some approaches obtain better spatial resolution, some capture more displacement components, and some obtain better temporal resolution. The broad categories of PIV will be presented here, discussing their suitability for the current research goals, in order to give an overview of the range of application of the forthcoming methods.

Planar PIV

Simplest of the various PIV setups is planar PIV, sometimes referred to as standard PIV or often 2D-2C, relating to the fact that two displacement components are obtained over a two dimensional field of view. A single plane of the fluid continuum is imaged by a single camera perpendicular to the flow (as depicted in Figure 1.1), and analysed to provide a two-dimensional displacement field. Acquisition rates are typically limited in the region of tens to low hundreds of Hz implying that sequential *pairs* of images can be considered as independent samples of the displacement field. The primary benefits

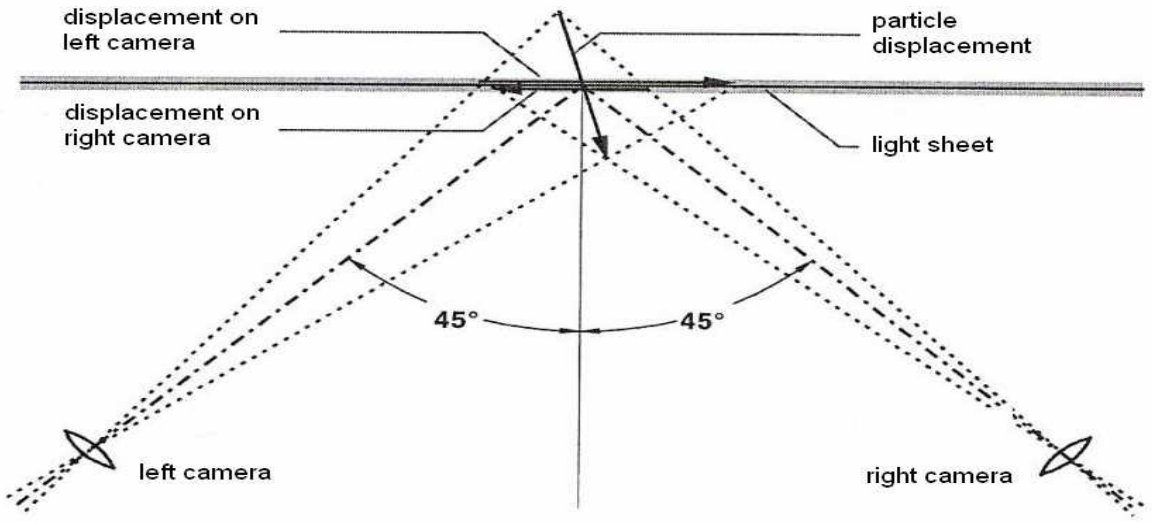


Figure 2.6: Diagram showing the arrangement for stereo PIV. Image from [14]

of standard-PIV over more elaborate approaches, such as stereoscopic and tomographic PIV below, are the relative[1] affordability and simplicity of experimental setup.

Stereoscopic PIV

By using a second camera, offset from the first by a known angle, as depicted in Figure 2.6, the relative displacement at a location can be combined using trigonometry to obtain all three components of the displacement. Accordingly, stereo-PIV is often referred to as 2D-3C, since three displacement components are obtained over a two dimensional field of view. This approach, however, requires careful calibration to allow for accurate reconstruction of the out-of-plane displacement[13], which is further limited by the physical angle between the two cameras. While it is feasible to apply adaptive techniques to reconstructed displacement fields from stereo-PIV, this adds a number of non-trivial challenges which lie beyond the scope of this thesis. This technique is commonly used to obtain some degree of out of plane displacement, required for calculation of some fluid properties, without the necessary jump in cost to tomographic PIV.

Tomographic PIV

Although stereo PIV can extract the third displacement component, within some degree of accuracy, it remains restricted to a relatively thin illumination plane. So-called tomographic PIV (tomo-PIV) overcomes this by illuminating a volume of fluid and imaging with multiple cameras, typically four or more, and is often referred to as 3D-3C,

[1]The term relative should be stressed here. A cheap setup is likely to cost at least tens of thousands of pounds.

since three displacement components are obtained over a three dimensional field of view. Beyond the additional cost of hardware, analysing the data from tomo-PIV requires additional post-processing to reconstruct volumetric images that can be interrogated using three dimensional cross-correlation. By using a larger illumination volume and capturing images from more angles, all three displacement components can be captured with similar accuracy.

Time-resolved

By sufficiently increasing the acquisition rate into the hundreds or thousands of Hz, or by substantially reducing the flow velocity, it is possible to image and track the same particle over several instances, provided it stays within the illumination volume. In doing so, more temporal information about the flow behaviour can be gleaned from experiments, particularly looking at more turbulent flows with lots of mixing. This approach is referred to as time resolved (TR) PIV. Increasing the imaging frequency in this way is not limited to a particular imaging configuration, i.e. Planar-, Stereo-, or Tomo-PIV, however requires significantly more advanced hardware capable of operating at these repetition rates.

2.7 Remarks on experimental background

In the first part of this chapter the technical background of a PIV experiment has been introduced. Broadly speaking, the process contains four components:

1. Seeding: The introduction of tracer particles into the fluid
2. Illumination: Illuminating a small plane of the fluid using a pulsed laser
3. Imaging: Recording the particle images for future digital processing
4. Image analysis: Processing the recorded images to extract the displacement of particles from time t to $t + \delta t$.

Each of these aspects have an impact on the overall ability of an experiment to investigate the behaviour of the fluid. For example, the seeding influences how accurately the particles follow the flow, illumination must ensure sufficient light scattering from the particles so that they can be precisely discerned, and the imaging lens inherently dictates the shape of particle images which, in turn, affects the processing of the images. While each component is important in and of itself to an experiment's success, it is desirable to

separate the experimental configuration from the analysis of images, such that *a priori* information and assumptions are not required[2] to accurately resolve the fluid motion.

2.8 Image analysis overview

Before breaking down each section in detail, it is helpful to establish the key components of a typical algorithm upfront. These components are displayed in Figure 2.7, and the motivations for each are briefly discussed. The key steps in the algorithm described remain general, however, this approach remains synonymous with the method of Scarano and Riethmuller [15] titled 'WIDIM', which will be used throughout this thesis to refer to the canonical, standard, conventional PIV image analysis routine.

The process begins by ensuring the raw images are of adequate quality, otherwise the accuracy or reliability of the results may deteriorate. High quality images are those that have uniformly bright particles on a perfectly dark background, as these enable better cross-correlation behaviour, ease particle detection routines, and allow for smaller cross-correlation windows without diminishing robustness. In addition to the image characteristics, it is often the case that the FOV contains regions that shouldn't be analysed, perhaps due to the presence of an object or an image artefact that is not able

[2] A notable exception is the information required to convert the measured displacement in pixels into a velocity in m/s, requiring the separation time between images and a reference length.

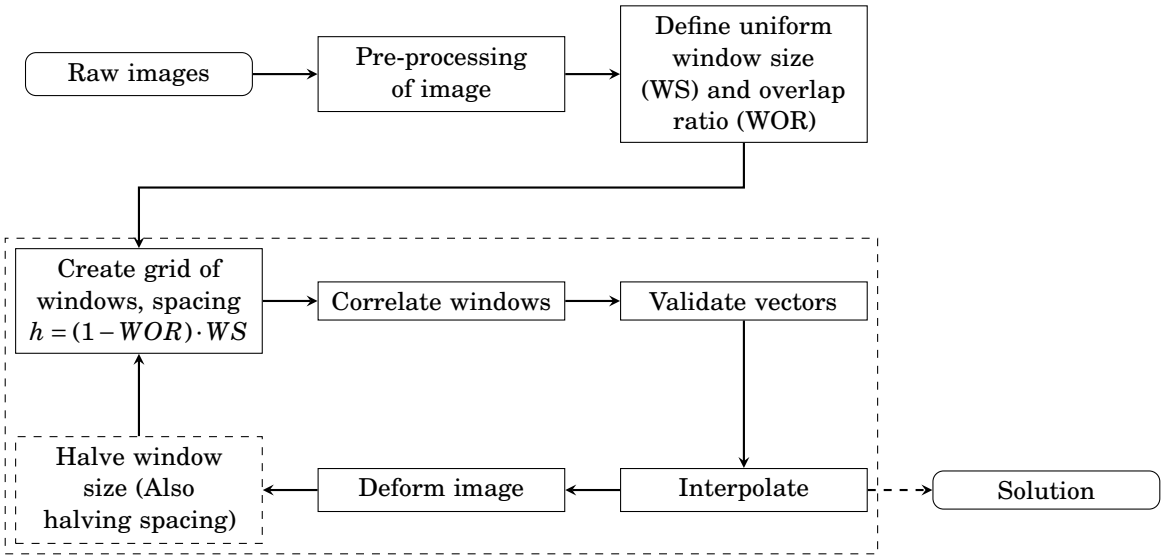


Figure 2.7: Flow chart outlining the main steps in a conventional PIV image analysis routine. The dashed box indicates the main loop of the process which will be performed K times. Note that WS halving may only occur on a subset of the K iterations.

to be removed, e.g. a strong reflection or shadow. These regions can be neglected by analysis routines by use of a mask, although such masks are known to influence the behaviour of sampling near the interface. Image masking is discussed in more detail in Section 2.9.2.

Correlation windows are arranged over the image on a cartesian grid, where the spacing between grid vertices, h, i.e. window centres, is found from (2.14) where WOR is the window overlap ratio and WS is the linear dimension of the correlation window.

$$h = (1 - WOR)WS \tag{2.14}$$

Traditionally, windows are sized, based on user-experience and trial-and-error, uniformly over the image and are gradually reduced in size over several iterations to improve spatial resolution. To aid robustness in the first iteration windows are conservatively large, at the expense of computation and resolution, however, as explored in Section 4.6. Using the cross-correlation operator an estimate for the local particle image displacement can be obtained for each window, building a coarse displacement field. The process of cross-correlation, sub-pixel displacement extraction, and implications for reliability and accuracy, are explored in more detail in Section 2.10.

It is important to acknowledge at this point that the cross-correlation process will *always* return a displacement vector, with no regard for the validity of this vector. Due to the presence of image noise, complex flow behaviour, or poor seeding density, it is not uncommon for the displacement vector obtained from a cross-correlation interrogation to be completely erroneous. In fact, it is argued that an absence of these so-called outliers indicates poor experimental setup [12], although their frequency can be minimised by good experimental setup. Accordingly, the returned displacement field must be checked for congruity by means of a vector validation routine (see Section 2.11), comparing each displacement vector with its neighbours. These methods must strike a balance between over-detection, i.e. incorrectly identifying valid vectors as invalid, and under-detection, i.e. failing to detect erroneous vectors, with limited available information.

To facilitate better spatial resolution, PIV image analysis routines typically implement an iterative approach wherein the raw images can be deformed in the opposite sense to the observed displacement, reducing the residual displacement for subsequent iterations. This approach is known to be very beneficial to both the accuracy, and reliability, of PIV image analysis routines (see Section 2.12). However, for an image to be deformed, an estimate of the displacement is required at every pixel, and hence interpolation is required to obtain the so-called dense predictor. Obtaining the dense predictor efficiently is a challenging topic that will recur throughout this thesis.

Finally, due to the reduced residual displacement, the window size, and hence grid spacing, can be reduced gradually to enable better spatial resolution and allow the residual displacement to be minimised. Typically, this reduction takes place over two to four main iterations, after which a couple of convergence iterations, wherein the WS remains fixed, may be performed to refine the solution. For this reason, the 'halve window size' step in Figure 2.7 is optional and may be omitted from some iterations.

2.9 Image Preparation

The ideal PIV image contains uniformly bright, uniformly sized, sharp, particles on a perfectly dark background. In this scenario, the correlation map contains only contributions due to correlation of particles, although as mentioned in As pointed out in Section 2.5.2, these contributions may be from random correlation between non-pairs or displacement-based correlation of the same particle. Provided there are sufficient particles in the interrogation region, it is therefore most probable that the largest peak in the correlation map corresponds to the underlying displacement.

In practice, PIV images are rarely perfect and likely contain artefacts which may hinder the estimation of fluid displacement. Oftentimes, images contain stationary objects, e.g. an aerofoil or surface, and reflections thereof, or imperfections due to the imaging apparatus, e.g. thermal noise. Unfortunately, the cross-correlation operator remains purely mathematical and will indiscriminately return a maximum value, and hence displacement, associated with the underlying image intensities. This can cause spurious behaviour in regions where such a displacement is infeasible, for example, returning zero displacement in the centre of the flow field due to a reflection. Conversely, non-stationary artefacts, e.g. object vibration, are still evaluated as local displacement, yet can not be trivially distinguished from true particle displacement.

PIV images are, therefore, often pre-processed to mitigate some of the adverse effects. For example, an image mask can be defined indicating regions where no displacement information is expected, e.g. due to the presence of a stationary object or in regions of very poor image quality. Analysis algorithms can disregard these regions, safeguarding from spurious data and improving computational efficiency. Alternatively, image quality can be improved by attempting to remove such artefacts or normalise particle brightness, however, care must be taken here to avoid loss of information stored in the image.

Pre-processing steps which alter the image, in order to facilitate better data recovery rates, are implicitly modifying that encoded information, potentially obscuring the

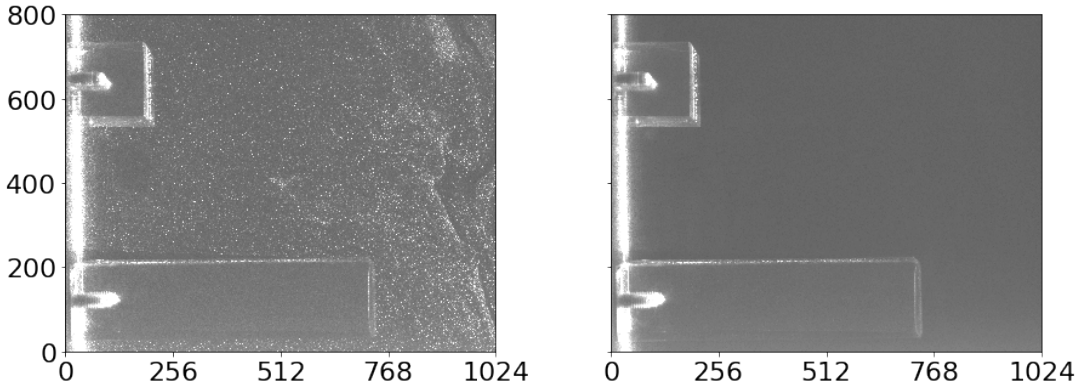


Figure 2.8: Background image generation using an ensemble minimum filter. (a) Exemplery raw image (b) ensemble minimum background image.

truth in the process. Rather, it seems prudent to develop algorithms to make smart decisions in the face of confusing data, rather than contaminating such data with various assumptions about what it should be. For this reason, while it is acknowledged that pre-processing techniques are extremely beneficial in the analysis of PIV images, care must be taken when applying these steps.

2.9.1 Image pre-processing

A common artefact of PIV images is background noise, typically caused by ambient light or thermal noise in the camera. This noise can significantly degrade the analysis performance, yet is easily removed through using a background subtraction method [16]. The background image can either be obtained during the experimental setup, without seeding, or calculated from the ensemble of seeded images via an ensemble average or minimum [17]. Figure 2.8 shows the background image generated by searching for the ensemble minimum, per pixel, over 150 images.

The effects of background subtraction are demonstrated in Figure 2.9, for an arbitrary, synthetically generated arrangement of particles, where the mean noise level has been subtracted, and the influence on the correlation map shown. Despite the magnitude of the correlation peak decreasing, the contrast of the peak with respect to the background correlation values is much greater, facilitating more reliable peak detection.

Another issue that affects cross-correlation, is non-uniform particle image intensity. Since cross-correlation is based upon intensity values, particle images whose intensity is significantly greater than the mean can dominate the cross-correlation, in doing so

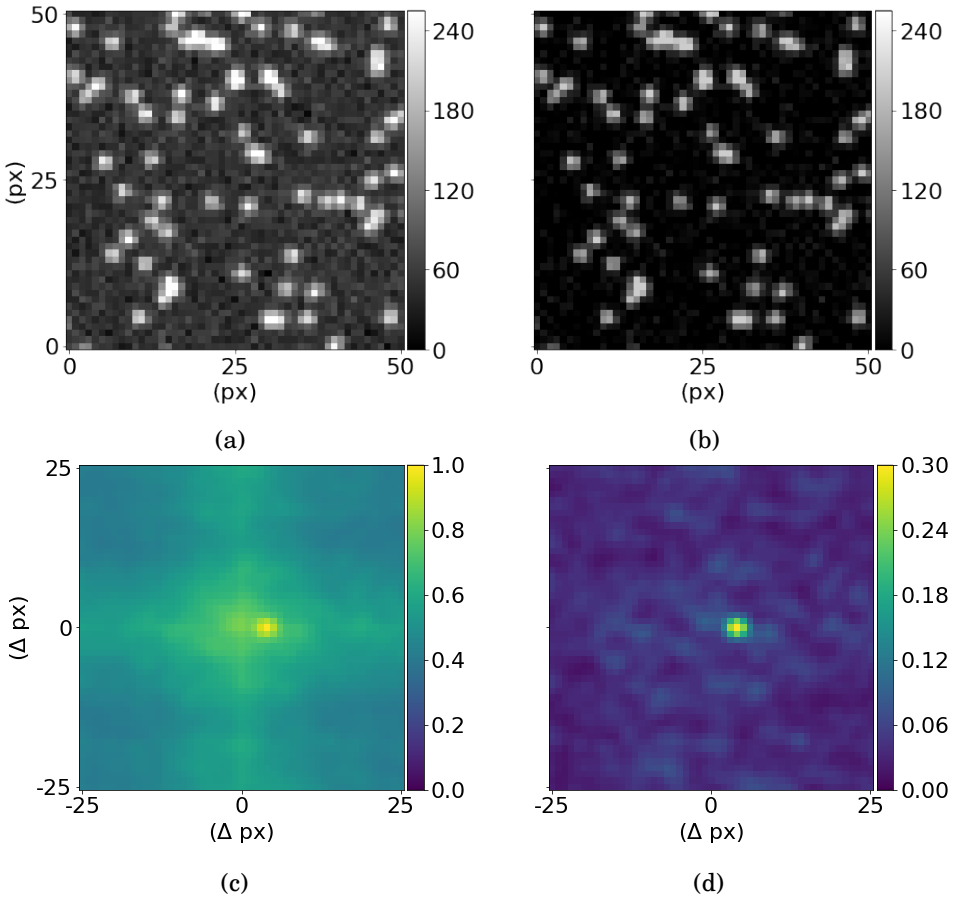


Figure 2.9: Demonstration of the importance of background subtraction in cross-correlation. PIV images (a) with background noise (b) with mean background intensity subtracted. Resulting correlation map for (c) with background noise (d) with background noise subtracted. Expected displacement is 4px to the right. Note the difference in correlation magnitude, yet the increase in contrast of the displacement peak.

effectively muting the contribution from dimmer particles. Occasionally, successive bright particle images don't belong to the same particle, causing the measurement to be biased to some random displacement, increasing the likelihood of outliers (see Section 2.11). More commonly, however, the effect is that the randomly brighter particle images will contribute more to the correlation map, biasing the measured displacement to these particles' in an unpredictable manner.

To counter both of the above issues, particle brightness can be normalised. One method to achieve this is to cap image intensity to $I_{median} + n\sigma_I$, where I is the pixelwise image intensity and n is some user-specified constant, such that the majority of particle images have approximately similar brightness values [18]. Alternatively, a min/max

filter can be applied wherein, over a small region of typically 7x7 or 15x15 pixels, intensity values are mapped to values between zero and unity, effectively removing spatial variations with a length scale greater than the filter size [19].

While there are many more image pre-processing techniques that can increase the amount of data yield, they must be used with caution. Embedded in PIV images is information about the underlying flowfield being examined, yet this is combined with undesirable noise and other artefacts. The goal of the aforementioned image pre-processing techniques is to remove the undesirable side-effects, without affecting the information pertaining to the flow, yet this is seldom achieved in practice. The reality is that the flow information encoded in the particle images is influenced by pre-processing techniques, leading to increased measurement uncertainty. The increase in uncertainty must be weighed against the increase in data yield, and image pre-processing should not be used without understanding of their influence on the final solution.

2.9.2 Image masking

As the experimentalist is frequently interested in fluid interact action with some structure or object, e.g. flow around a cylinder, aerofoil, or step, PIV images will often contain object boundaries or surfaces. The presence of these surfaces will cause a peak in the correlation map near the origin, since they do not move between successive images. It is common practice, therefore, to omit the pixels which are known to belong to stationary objects from the cross-correlation procedure by using an image mask. An example of such is shown in Figure 2.10, based on the same image as Figure 2.8, along with the resulting ready-to-be-analysed image, which has also had the background subtracted.

The most basic approach to generating logical masks is to manually draw one using computer drawing software, although this approach is time consuming, complex, and is impractical for any moderately complex shape. Reflections from objects add further difficulty in accurately identifying which pixels truly correspond to the object and which correspond to the flow, leading to an element of guesswork by the user. Automated, or at least semi-automated, mask generation techniques alleviate the situation and allow for rapid masking, which typically makes use of image gradients to identify flow-object interfaces. For example, Dussol *et al*. [20] use the edge detection of Canny [21] and morphological smoothing to identify the edge contours of a hull, in order to create a mask for the first PIV image in the sequence. The known oscillatory sinusoidal motion of the hull can then be used to automatically update the mask for the subsequent frames of the image set. Similarly, Sanchis and Jensen *et al*. [22] capitalised on image intensity

gradients to extract the air-water free surface interface. Their method identifies the local position and orientation of the interface at a series of locations, which was then interpolated using a cubic interpolant to obtain the mask over the whole image.

In a slightly less conventional approach, Masullo and Theunissen [23] use the pixel intensity statistics from the entire ensemble of images to extract the image mask. The underlying principle is that regions which contain flow information are characterised by a bimodal intensity distribution - mostly dark background, with the occasional very bright particle. Conversely, regions without flow information are characterised by a normal distribution of intensity, i.e. small fluctuations about a single mean value. Such pixels can be identified using tests for normality with as few as 40 images. This approach is highly desirable, since parameters can be semi-universally set, allowing for autonomous mask generation in line with one of the primary motivations for this research - user independence.

2.10 Estimation of particle displacement

As introduced in Section 2.5.2, cross-correlation can be used to calculate the displacement between two images separated by a small time delay δt. The following section digs deeper into the use of cross-correlation for PIV displacement estimation, discussing implementation details and guidelines for improved performance. Such guidelines will be frequently referenced in subsequent chapters, notably Chapter 4, to maximise the performance of displacement estimation by automatically selecting parameters according

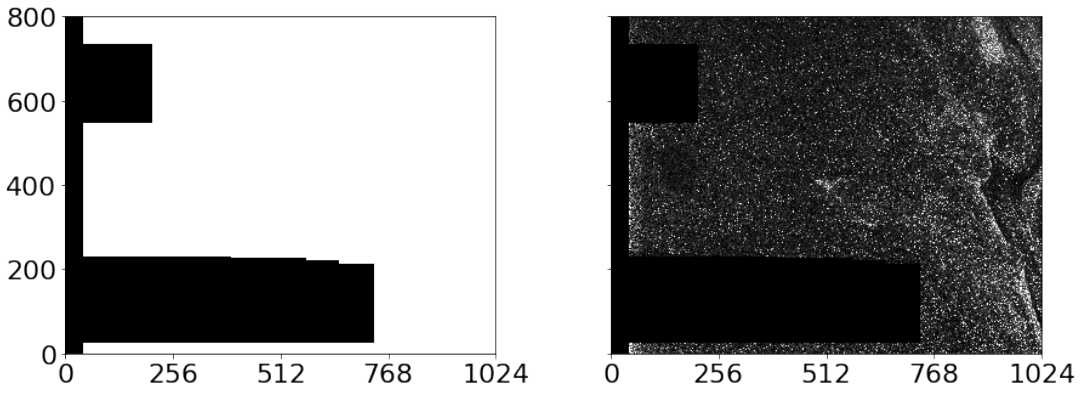


Figure 2.10: (a) Manually drawn mask (b) Final pre-processed image with background subtracted and mask imposed. Brightness increased for clarity.

to these guidelines and the observed conditions.

2.10.1 Cross-correlation

To begin, we will first recap on the concepts introduced previously, covering the fundamental mathematics, noting that more detailed derivation of such maths can be found in the works of Raffell *et al*.[1], Westerweel *et al*.[24, 25], and Keane *et al*.[26]. Given N_p particles in the object plane, with locations that map to $\mathbf{x_i}$ in the image plane, for $i = 1...N_p$, at time t, and subsequently at locations $\mathbf{x'_i} = \mathbf{x_i} + \mathbf{d_i}$ at time $t + \delta t$, the resulting continuous images I_a and I_b, respectively, are defined using (2.8) and (2.9).

$$I_a(\mathbf{x}) = \sum_{i}^{N_p} I_0(\mathbf{x}_i)\tau(\mathbf{x} - \mathbf{x}_i) \qquad \text{(2.8 revisited)}$$

$$I_b(\mathbf{x}) = \sum_{i}^{N_p} I_0(\mathbf{x}'_i)\tau(\mathbf{x} - \mathbf{x}_i - \mathbf{d_i}) \qquad \text{(2.9 revisited)}$$

Where $\tau(\mathbf{x} - \mathbf{x_i})$ is the point spread function of the lens, i.e. the shape of the particle in the image plane, which can be approximated by a Gaussian. The continuous cross-correlation equation is then written as in (2.12), where $\mathbf{s}$ is the displacement in the correlation plane.

$$R(\mathbf{s}) = \sum_{i}^{N_p}\sum_{j}^{N_p} I_0(\mathbf{x}_i)I_0(\mathbf{x}'_j)\int \tau(\mathbf{x} - \mathbf{x}_i)\tau(\mathbf{x} - \mathbf{x}_j - \mathbf{d} + \mathbf{s})d\mathbf{x} \qquad \text{(2.12 revisited)}$$

As per Keane and Adrian [26], contributions to the correlation plane, $R(\mathbf{s})$ can be decomposed into three components; the contributions from the correlation of mean intensities, fluctuating intensities, and the displacement of particles, represented by R_C, R_F, and R_D, in (2.15), respectively.

$$R(\mathbf{s}) = R_C(\mathbf{s}) + R_F(\mathbf{s}) + R_D(\mathbf{s}) \qquad (2.15)$$

The correlation term $R_D(\mathbf{s})$ represents contributions to the correlation map for all $\mathbf{x_i} = \mathbf{x_j}$, c.f. Section 2.5.2, and can be shown to be represented by the following equation [1]

$$R_D(\mathbf{s}) = R_\tau(\mathbf{s} - \mathbf{d})\sum_{i}^{N_p} I_0(\mathbf{x_i})I_0(\mathbf{x'_i}) \qquad (2.16)$$

Where R_τ is the autocorrelation of particle images, which represents their intensity profiles or shapes, and obtains a maximum for $\mathbf{s} = \mathbf{d}$.

In practice, images are discretised onto a pixel array wherein the intensity value for each pixel is proportional to the total incident light over the pixel's area, as described by (2.17), where m and n represent the horizontal and vertical mid-points of a specific pixel, and I_A represents the discretised image. A similar equation can be constructed for I_B.

$$I_A(m,n) = \int_{x=m-\frac{1}{2}}^{x=m+\frac{1}{2}} \int_{y=n-\frac{1}{2}}^{y=n+\frac{1}{2}} I_a(x,y)dxdy \tag{2.17}$$

The discrete form of cross-correlation is, therefore, shown by (2.18) where M and N represent the dimensions of the region to be correlated.

$$R(i,j) = \sum_{m=1}^{M} \sum_{n=1}^{N} I_A(m,n) I_B(m+i,n+j) \tag{2.18}$$

As mentioned in Section 2.5.2, the convolution theorem, shown in (2.19), can be used to significantly accelerate the cross-correlation process, due to the efficiency of the FFT. Using the FFT algorithm in this way reduces computational complexity from $O[N^4]$ to $O[N^2 \log_2 N]$, where N is the spatial dimension of the correlation window.

$$R(i,j) = \mathscr{F}^{-1}\{\mathscr{F}\{I_A\}\mathscr{F}\{I_B\}\} \tag{2.19}$$

Finally, to mitigate intensity variation between exposures it is important to properly normalise the input intensities as per [8], shown in (2.20) which now represents the discrete normalised equation for cross-correlation.

$$R(i,j) = \sum_{m,n} \frac{(I_A(m,n) - \overline{I_A})(I_B(m+i,n+j) - \overline{I_B}(m,n))}{\sigma_{I_A}\sigma_{I_B}} \tag{2.20}$$

2.10.2 Sub-pixel displacement

Once the discrete correlation has been calculated, a lookup for the location (i, j) of the largest value reveals the particle displacement[3] to the nearest integer value. Considering the typical maximum displacements to be approximately 10 to 15px, measured displacements limited to integer values result in unacceptably large relative errors. Therefore, the precise location of the peak must be resolved with sub-pixel accuracy to maximise information extraction.

One straightforward approach is to obtain the centroid of the correlation peak from the ratio of first- to zeroth-order moments, as in (2.21). However, this approach requires

[3]For the sake of the current discussion, we assume that the largest peak does indeed correspond to the particle displacement, and is not due to random correlation.

segmentation of the signal peak from background noise, which is often ambiguous in practice [1].

$$x_0 = \frac{(i-1)R_{i-1,j} + iR_{i,j} + (i+1)R_{i+1,j}}{R_{i-1,j} + R_{i,j} + R_{i+1,j}} \tag{2.21a}$$

$$y_0 = \frac{(j-1)R_{i,j-1} + jR_{i,j} + (j+1)R_{i,j+1}}{R_{i,j-1} + R_{i,j} + R_{i,j+1}} \tag{2.21b}$$

A more robust approach involves fitting the peak correlation values to a known function, from which the sub-pixel peak location can be inferred. Two successive one dimensional fits can be used to obtain the x and y coordinates of the peak. This fit is typically only applied to the immediately neighbouring correlation map values, leading to these approaches being referred to as three-point estimators. For the same reason, these methods tend to work best when the particle image diameter is approximately 2-3 pixels in diameter. A parabolic fit may be adopted, using (2.22), however, this leads to significant peak-locking due to biasing of the result to integer displacements.

$$x_0 = i + \frac{R_{i-1,j} - R_{i+1,j}}{2R_{i-1,j} - 4R_{i,j} + 2R_{i+1,j}} \tag{2.22a}$$

$$y_0 = j + \frac{R_{i,j-1} - R_{i,j+1}}{2R_{i,j-1} - 4R_{i,j} + 2R_{i,j+1}} \tag{2.22b}$$

As explained by [27], this is because the shape of the correlation peak is poorly approximated by a parabola, and instead is better approximated by a Gaussian, as explained in Section 2.4. Applying a Gaussian fit to the correlation data, using (2.23), significantly improves the ability to accurately extract sub-pixel displacements [27, 8].

$$x_0 = i + \frac{\ln R_{i-1,j} - \ln R_{i+1,j}}{2\ln R_{i-1,j} - 4\ln R_{i,j} + 2\ln R_{i+1,j}} \tag{2.23a}$$

$$y_0 = j + \frac{\ln R_{i,j-1} - \ln R_{i,j+1}}{2\ln R_{i,j-1} - 4\ln R_{i,j} + 2\ln R_{i,j+1}} \tag{2.23b}$$

Three point Gaussian curve fitting estimators represent a cost-effective, accurate, means of extracting the sub-pixel displacement from the correlation map, however, there exist more accurate methods that can be used at the expense of computational cost. Although three-point estimators are used throughout the works in this thesis, a brief discussion of two of the higher fidelity methods is given below for context.

Onesuch method is introduced by Lourenco and Krothapalli [27], and further discussed by Roesgen [28], which uses cardinal interpolation for signal reconstruction, also known as Whittaker reconstruction, to iteratively refine the location of the maximum

peak value by using a 5x5 stencil to reconstruct the signal for each iteration. Typically six iterations are required to obtain the peak location to the desired level of accuracy of 1/64th of a pixel. Although the method boasts improved accuracy over the three-point Gaussian fit, it is significantly more computationally expensive and therefore not commonly used.

Another approach to for sub-pixel displacement estimation is to fit a single two dimensional Gaussian model, rather than two successive one dimensional three-point curve fits [29]. The coefficients for the two dimensional equation are found using a regression analysis of the peak intensities, which then allows for sub-pixel peak location from simple analytical functions. By fitting a single two dimensional model in this way, the approach is less sensitive to small variations in peak shape and results in notably less peak-locking effects than using three-point estimators. However, the approach relies on the application of a Gaussian filter, introducing an additional filtering effect which may influence the data recovery in the image, to minimise noise, without which the accuracy of the method is largely comparable to two three-point estimators. Furthermore, it performs poorly when there is a constant offset in the intensity value, i.e. in the presence of background noise, and therefore accurate removal of background gray noise is essential with this method.

2.10.3 Window requirements

A correlation window is capable of returning a single displacement vector, representing the most likely displacement of particles within the window. Appropriate choice of WS is crucial to maximising the accuracy and reliability of an image analysis algorithm, and is influenced by many factors. Generally, the ideal correlation window contains negligible noise with uniformly sized and shaped particles each with a small uniform displacement. A few of the most prominent factors are, therefore:

- Number of particle image pairs
- Particle image displacement magnitude
- Velocity gradients
- Particle image size and shape
- Background noise

Perhaps the most important factor is the number of particle images contained within a correlation window; as the number of particles increases, so too does the signal peak

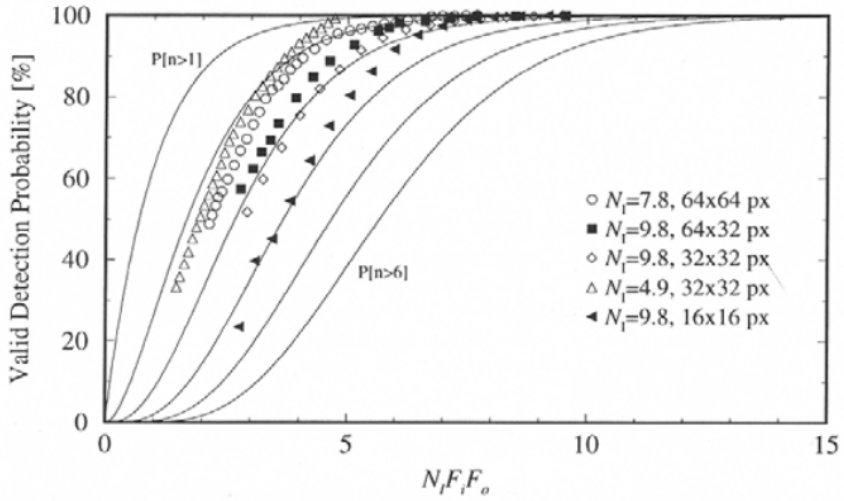


Figure 2.11: Effect of effective particle density (NIFIFO) on the valid detection rates of cross-correlation. Image from [1].

and detectability. The detectability of a signal peak is often characterised by the signal-to-noise ratio (SNR), defined as the ratio between the largest peak and second-largest peak, in the correlation map. Although in practice the largest peak is not guaranteed to be signal, a large SNR indicates confidence that the largest peak actually *is* signal. In addition to the number of particles, the local displacement magnitude also affects detectability; large in-plane displacements, i.e. (u,v), or out-of-plane displacements, i.e. w, may result in particles which are present in W_A, but not W_B, and vice versa, where W_A and W_B represent the pixel intensity values from I_A and I_B, respectively, within the region of the correlation window. These losses are quantified using F_I and F_O for in- and out-of-plane loss of pairs, respectively, varying from zero for total loss of pairs to one for no loss. The number of effective particle image pairs can therefore be described by the term $N_I F_I F_O$, where N_I represents the number of observed particles in the region. In order to achieve 95% valid detection probability, $N_I F_I F_O$ is recommended to be approximately eight [30]. Figure 2.11 shows how the valid detection rate varies with $N_I F_I F_O$

A regularly used threshold on particle displacement magnitude is $WS/4$, commonly referred to as the *one quarter rule*[30], to ensure adequate valid detection rates. Of course, without *a priori* knowledge of the displacement field, it is not possible to know such a size. Fortunately, most PIV experiments are carefully designed to have maximum displacements in the region of 8-12px, driven by acceleration based errors described in Section 2.3, and thus first pass correlation windows of at least 32-48px will generally satisfy the *one quarter rule*. More on initial window sizing can be found in Section 4.6.

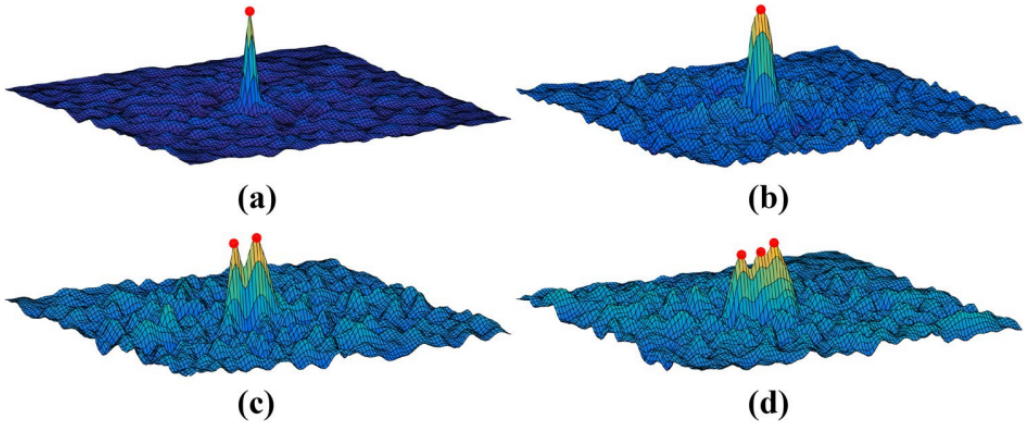


Figure 2.12: Effect of gradients on the correlation map. Image from [31].

In addition to limits on displacement magnitude, displacement gradients within the windowed region should also be minimised for two key reasons. Firstly, the correlation approach is only able to extract a single displacement per window, hence displacement gradient information will not be captured from a single window. Additionally, under large gradients a smearing of the correlation peak will occur, which can reduce the accuracy of sub-pixel fitting, as well as reduce the height of the peak, shown in Figure 2.12, reducing detectability and possibly leading to more outliers (see Section 2.11).

The effects of displacement gradients on the uncertainty of measurements are quantified by monte-carlo simulations, from [1], and shown in Figure 2.13, wherein the considerable increase in RMS for increasing gradients can be observed. Furthermore, it shows that smaller correlation windows better tolerate large displacement gradients due to the fact that the aforementioned peak smearing behaves as a linear function of window size. Keane and Adrian, therefore, suggest a general rule that displacement gradients should be restricted to <5% of the interrogation size[32].

The size of the particle image also influences the accuracy of PIV measurement accuracy. As discussed in the previous section, the process of calculating the sub-pixel peak loaction is particularly sensitive to noise in the pixel intensities. As such, when the particle size is small (d_τ <= 1.5px), neighbouring intensities are dominated by noise, and not by correlation signal, leading to larger uncertainty as seen in Figure 2.14. Conversely, when the particle becomes large, variations in correlation value between neighbouring pixels reduce while background noise remains the same. Therefore, the relative noise effectively increases again, thus reducing the ability to identify the true peak accurately, also shown in Figure 2.14. Furthermore, a bias towards integer displacement values occurs when the particle image is too small, due again to the failure

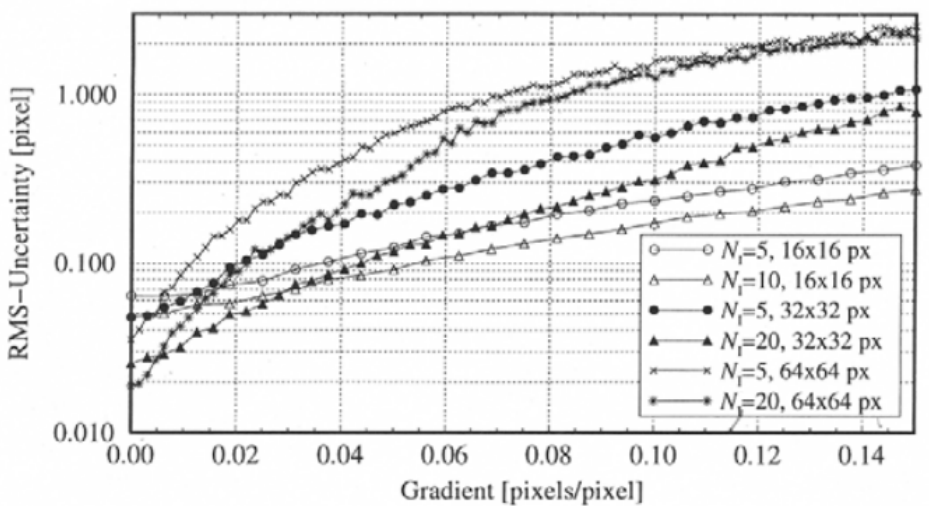


Figure 2.13: Effect of displacement gradients on the RMS uncertainty of cross-correlation. Image from [1].

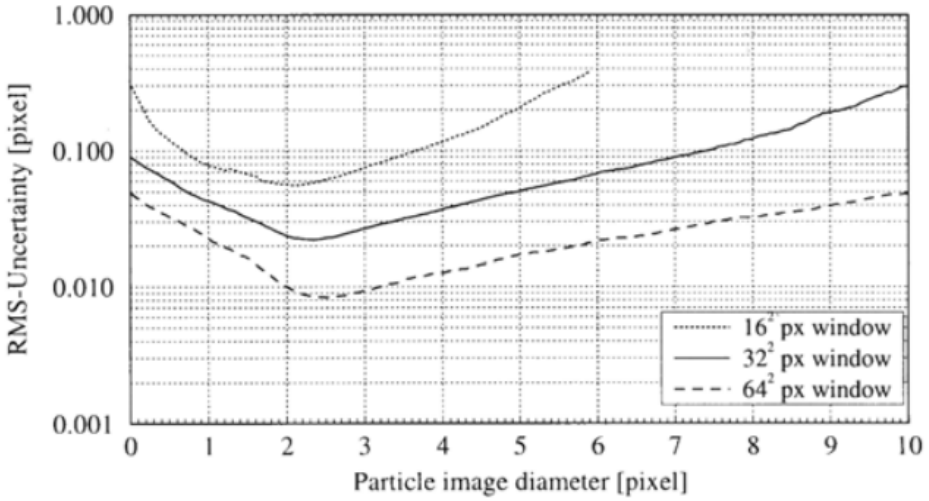
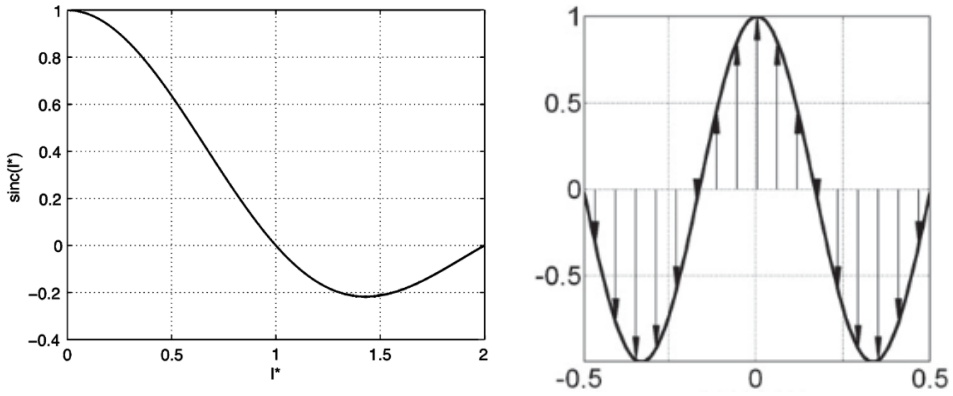


Figure 2.14: Effect of particle image diameter on the RMS uncertainty. Image from [1].

of sub-pixel estimators, in a phenomenon referred to as "peak-locking". Because of these effects, the optimum particle image diameter lies in the region of 2-3px. A final observation is that the larger the correlation window the more particles contribute to the signal peak, reducing the relative noise level and therefore reducing uncertainty.

2.10.4 Spatial response

Due to the spatial averaging effect of cross-correlation, an inherent low-pass filtering effect is present in PIV limiting the smallest detectable wavelength to the correlation window size, unless more elaborate frequency domain analysis is used [33]. As such, the cross-correlation operator is commonly assumed to act as a moving average (MA) filter,



(a) (b)

Figure 2.15: (a) The theoretical moving average (linear) frequency response of cross-correlation for normalised window size, $l*$. (b) Example one-dimensional displacement field for $l* = 1.5$. Image from [34]

for which the frequency response is described by the cardinal sin function, as shown in Figure 2.15(a), where $l*$ is the window size normalised by the wavelength. Importantly, it is noted that the response is negative for certain values of $l*$, e.g. for $l* = 1.5$ in Figure 2.15(b).

Contrary to this common assumption, the cross-correlation operator as used in PIV is *non-linear* and is not generally described by a MA filter, except for in specific conditions. To behave as a MA filter, the gradients, out-of-plane displacement, and ratio of displacement amplitude to particle image diameter must all be small [34, 35], which is seldom the case in non-trivial flows. Figure 2.16 shows the actual extracted displacement from cross-correlation for a sinusoidal displacement for a range of normalised window sizes. Firstly, for small WS/λ there exists a bias towards regions of higher curvature, i.e. the peaks of the sinusoid. Secondly, the response for $WS/\lambda > 1$ is far more significant than a MA filter would suggest, since PIV selects the *most probable*, i.e. the modal, displacement. Consider Figure 2.15(b) again; assuming a uniform distribution of particles, there will be twice as many particles with negative displacement as posistive displacement. There will exist two prominent peaks in the correlation map at negative and positive one, however, the negative peak will be approximately twice as large and will therefore be detected as the actual displacement.

This non-linear behaviour implies that it is crucial to obtain correctly sized windows from the offset, as otherwise the residual displacement may *increase*. Increasing residual displacement should be avoided since this leads to increased loss of pairs, F_I, which drastically reduces $N_I F_I F_O$ and subsequently valid detection rates. Furthermore, this

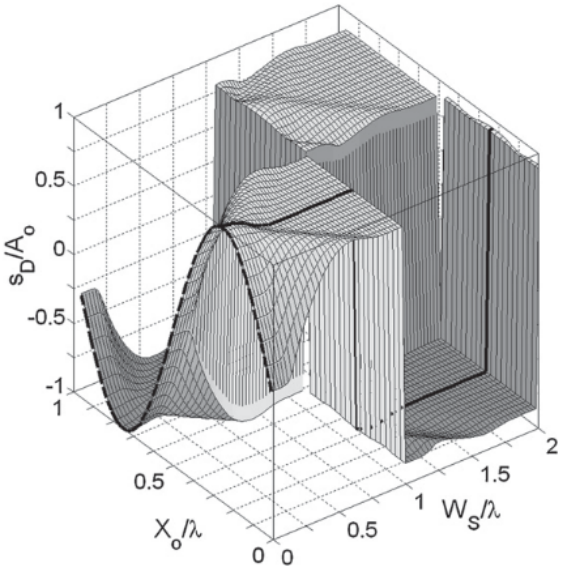


Figure 2.16: The actual cross correlation response for varying normalised WS. Image from [34]

can lead to irrecoverable distortion of image via image deformation (Section 2.12). In order to mitigate this, a completely new method will be presented in Chapter 4, which automatically determines an optimal initial window size.

2.11 Vector validation

When interrogation windows are correlated, it is anticipated that the largest peak in the correlation map corresponds to the true particle image displacement. Yet, random correlation between un-paired particle images can lead to a larger peak than the true signal, leading to a random, erroneous, displacement estimate (outlier). Figure 2.17 shows an example of how this may occur. The likelihood of this occurring can be reduced by increasing the correlation window size, however, this is in direct conflict with the desire for maximising spatial resolution. As such, outliers are inevitable even for optimally designed experiments [12] and, therefore, require identifying and correcting where possible to safeguard the accuracy of the measure displacement field. This is of particular importance for adaptive methods which rely on the displacement field to guide further sampling efforts; if the displacement field is incorrect then the performance of adaptive methods can be significantly hindered.

One of the earliest and most basic approaches to outlier identification was by Keane and Adrian [32] who define a signal to noise ratio (SNR), which is the ratio of the largest peak in the correlation map to the second largest peak. Outliers may then be identified as

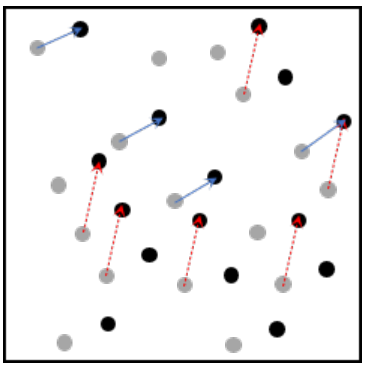


Figure 2.17: Demonstration that random arrangements of particles may cause alternative viable displacement pairings. Grey dots are particles at time t, black dots are particles at time $t+\delta t$. Blue arrows indicate the actual displacement, whereas the red arrows represent an alternative pairing and the corresponding correlation peak may dwarf that of the blue arrows..

those with a SNR less than some threshold, SNR_{thr}. A large SNR indicates confidence that the largest peak corresponds to signal and not to random noise, though provides no guarantee that this is the case.

Another method based upon correlation properties is "correlation based correction" by Hart [36]. The principle is that the displacement signal will be approximately the same in nearby windows, whereas noise is randomly distributed and will be different in the two windows. By multiplying the correlation maps of adjacent interrogation regions, peaks in similar locations will be amplified whereas peaks appearing in only one map will be attenuated. Unfortunately, this approach has conflicting desires for window spacing, which needs to be close enough for the displacement to be sufficiently similar, yet far enough apart for the noise characteristics to differ. Despite good performance in simple cases, the resulting displacement fields still contain outliers which need dealing with and hence further detection algorithms are required.

Considering that outliers are typically significantly different from their neighbours, Westerweel [37] developed a criterion which calculates the residual of each vector to the median of the neighbouring eight (3x3) vectors, declaring the vector an outlier if the residual is above some threshold. One of the downsides to this method is that the threshold is uniform across the image and therefore does not account for the flow being more or less turbulent. While Shinneeb *et al.* [38] proposed a variable threshold approach with this method, it still heavily depended on the original filter of Westerweel.

Westerweel and Scarano [39] subsequently proposed a modification to their earlier work making it universally applicable, irrespective of the underlying displacement field.

In this improvement, the residual is normalised by the median of the residuals of the neighbouring vectors, as per (2.24). In doing so, the authors found that histograms of residuals collapse onto a single curve, regardless of the flows turbulence level, allowing a universal threshold to be defined. In regions of almost perfectly uniform flow, the residual of the neighbours may be extremely small, leading to artificially large values of r_0^* and thus over-detection in these regions of uniformity. A constant offset, ϵ, is added, therefore, to represent the background noise which is generally accepted to be about 0.1px. This method is the most widely used outlier detection method within PIV algorithms, due to its simplicity, universality, and computational efficiency.

$$r_0^* = \frac{|U_0 - U_m|}{r_m + \epsilon} = \frac{|U_0 - \text{median}(U_i)|}{\text{median}\,[U_i - \text{median}(U_i)] + \epsilon} \tag{2.24}$$

The above method, however, implicitly relies on neighbouring vectors being equispaced, which is not necessarily the case for unstructured adaptive sampling or PTV leading to unequal weighting being ascribed to different displacement vectors. Duncan *et al*. [40] introduce an inverse distance weighting to account for this, shown in equation (2.25). Although the distance weighting does not add much computational complexity of itself, determining the inter-sample distances, often through a Delaunay triangulation, is a computationally intensive process. The authors state an increase in computation of approximately 3x relative to the approach of Westerweel and Scarano, although this is largely due to the triangulation process.

$$r_0^* = \frac{\left| \frac{U_0}{\text{median}(d_i + \epsilon_a)} - \text{median}\left(\frac{U_i}{d_i + \epsilon_a}\right) \right|}{\text{median}\left[\frac{U_i}{d_i + \epsilon_a} - \text{median}\left(\frac{U_i}{d_i + \epsilon_a}\right) \right] + \epsilon_a} \tag{2.25}$$

Vector validation methods are crucial to an iterative PIV image analysis algorithm, as they minimise the extent of invalid vector corruption of the displacement field estimate. Due to the statistical nature of the task, the false detection of outliers, either flagging a valid vector or missing an outlier, remains inevitable, particularly in regions where the natural level of fluctuations are already considerable. Despite traditional validation routines being based on data forming a structured grid, methods such as that of Duncan *et al*. or Masullo *et al*. [41], facilitate the validation of unstructured vector fields. As will be explored in Chapter 5, the local uniformity of vector spacing, analogous to that of a structured grid, significantly influences the performance of unstructured validation routines and, therefore, remains a desirable characteristic.

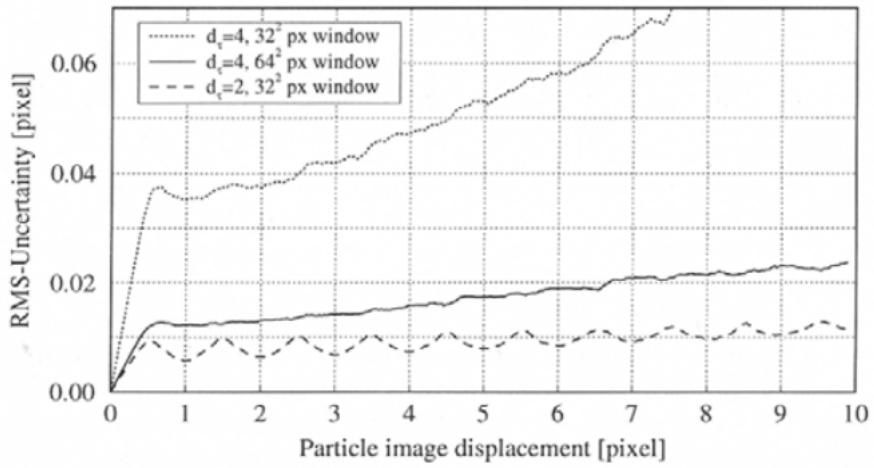


Figure 2.18: Effect of particle image displacement on the RMS uncertainty. Image from [1].

2.12 Iterative methods

When analysing a PIV image, it is beneficial to gradually refine the estimate of the displacement field using iterative algorithms, allowing more appropriate analysis properties at each stage of the process. In the early iterations, robustness is prioritised, necessitating larger windows, to avoid the issues detailed in the previous two sections, i.e. erroneous vectors. Conversely, smaller windows are used for subsequent iterations for a more localised estimation of the displacement. Concomitantly, the number of windows starts small and is increased, so that computational resources are reserved for later iterations where they can yield better estimates.

2.12.1 Motivation

As can be explained by theory [19], and obvserved by experiment [8, 25], the RMS error for cross-correlation significantly drops when the displacement, Δx, is less than half a pixel. This is demonstrated in Figure 2.18 from [1]. When the displacement is greater than half a pixel, the error continues to increase, albeit at a much lower rate. Furthermore, the effect of in-plane loss of pairs, F_I, is much reduced when the displacement is small, and therefore valid detection rates are also expected to be higher, as per Figure 2.11.

Keane and Adrian [26] are generally recognised as being the first to suggest using window offsetting to improve correlation performance. In the method, the authors achieve an effective window offset by allowing the correlation region in the second

image to be significantly larger than in the first image, thereby compensating for the in-plane displacement and loss of pairs. Similar approaches are reported in the literature, e.g. by Fincham *et al*. [42]. This approach, however, results in considerably greater computational cost, which can be mitigated by simply offsetting an equisized correlation region in the second image by an amount equal to the integer part of the displacement estimate [26].

The performance of discrete window offset was later investigated by Westerweel *et al*. [25], by means of an analytical model and experimental validation. In the paper, the noise was shown to be reduced for all values of u and $u\prime$ when window offsetting is adopted. The key argument is shown below, starting by defining the signal power, S, as the variance of the fluctuating displacement:

$$S = \mathrm{var}\{u\} = u'^2 \tag{2.26}$$

and subsequently define the total noise as

$$N \equiv \int \mathrm{var}\{\varepsilon|u\} f(u) \mathrm{d}u \tag{2.27}$$

where $\mathrm{var}\{\varepsilon|u\}$ is the variance of the measurement error, ε, for a given displacement, u, and $f(u)$ is the probability density function (PDF) of the displacement field. Based on the previous findings of Westerweel [19] and Willert [8], c.f. the behaviour of the RMS error for varying displacement from Figure 2.18, the variance of the measurement error can be described by (2.28).

$$\mathrm{var}\{\varepsilon|u\} = \begin{cases} 4c^2u^2 & \text{for } |u| < \frac{1}{2} \\ c^2 & \text{elsewhere} \end{cases} \tag{2.28}$$

where c is a constant which describes the general performance of the cross-correlation analysis procedure. Thus for small turbulence, i.e. small u', the reduction in noise, NR, defined by the ratio of SNRs without and with offset is found to be

$$\mathrm{NR} = \frac{1}{4u'^2} \tag{2.29}$$

Whereas for more turbulent flows the noise reduction tends to a more uniform value of approximately three. Since the variation of error (2.28) depends on the size of the interrogation window, via the influence of c, it is therefore argued that, for a given SNR, the spatial resolution can be improved by the use of smaller windows when using window offsetting.

2.12.2 Discrete offset

One of the early presented implementations of iterative image processing was that of Scarano and Riethmuller [43], and similarly by Westerweel *et al*.[25]. Following an initial estimate of the displacement field, called the predictor, WS can be reduced and windows offset in the second image by the integer fraction of the predictor. The authors showed that a decoupling of the dynamic range is achievable, due to the ability to maintain sufficient particle images within smaller windows.

When adopting a multigrid implementation, with increasing spatial resolution (smaller windows), two implications become apparent. Firstly, when reducing the sample spacing, predictor information is required in locations without previous data, requiring vector interpolation, discussed in Section 2.12.4. Secondly, the need to autonomously validate data between iterations, to avoid the propagation of erroneous vectors, becomes even more critical[4]. In this early method, vector validity was determined by comparing the local divergence of the vector field with a user-defined threshold, replacing invalid vectors using simple bilinear interpolation.

2.12.3 Continuous offset

Further reductions to the measurement uncertainty can be realised the closer the residual displacement is to zero, as from (2.28). Fractional window offset can achieve this, yet requires image intensity values be known at locations between pixels, necessitating image interpolation and increased computational cost. Since the images being analysed inherently contain a lot of high-frequency information, i.e. bright particles on dark backgrounds, higher-order image interpolation schemes are typically favoured to prevent loss of information at the expense of greater computational cost (see Section 2.12.5).

Early implementation of continuous window shifting can be attributed to Huang *et al*. [44, 45] using the particle image distortion (PID) technique. In this approach, the correlation region, called the particle image pattern (PIP) here, is distorted according to the measured gradients. However, the method used only a linear interpolation for image reconstruction and used a simple centroid scheme to find the sub-pixel displacement peak, thus the method's uncertainty was quite large. A similar approach was also proposed by Jambunathan [46]. In their work, more emphasis was placed on the algorithmic approach of their method, yet the authors also used a bilinear image reconstruction

[4]Such schemes already existed prior to the emergence of iterative methods, but solely as a post-processing step.

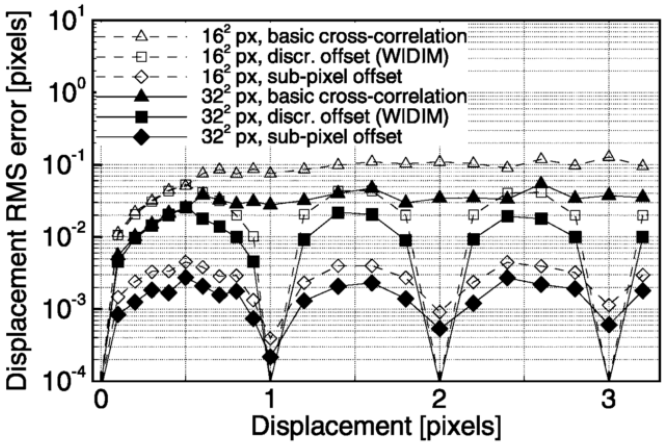


Figure 2.19: RMS uncertainty as a function of the displacement for basic cross-correlation, discrete window offset, and continuous offset. Image from [15]

scheme, and so levels of uncertainty were large. On a similar topic, Gui and Werely [47] investigated the use of continuous window shifting, using bilinear image reconstruction, specifically on the ability to suppress the peak-locking phenomenon (see Section 2.10.2).

Following on from these techniques, Scarano and Riethmuller [15] improved their previously proposed iterative discrete-offset algorithm, WIDIM, by implementing continuous window shifting. Critically, the proposed method used a sinc interpolation scheme which boasted low uncertainty levels for the reconstruction of PIV images. Figure 2.19 shows the significant effect of both discrete window offset and of continuous offset. Note that for basic cross-correlation, i.e. single pass, the uncertainty largely agrees with 2.18, whereas for both multi-pass approaches a significant decrease in RMS for integer displacements is observed, and that for continuous offset the overall uncertainty is lower again.

2.12.4 Vector interpolation

Interpolation is an important part of any modern PIV image analysis algorithm and it appears in three key locations; identifying the sub-pixel location of a correlation peak, as in Section 2.10.2, deforming images according to the underlying displacement field for continuous offsetting, as in Section 2.12.5, and for the interpolation of coarse displacement data onto a dense displacement field for post-processing or image deformation. Each step has its own requirements, sub-pixel fitting places restrictions on the shape of the interpolant, whereas image interpolation requires interpolation of high-

frequency content, demanding high-order interpolants at the expense of computational cost. Vector interpolation, however, requires good computational efficiency *sufficient* smoothness, without introducing instabilities or overfitting.

Due to the range of flow scales present within the displacement field, obtaining the best interpolator for a given vector spacing is not necessarily simple. Traditionally, a bi-linear interpolation scheme was adopted, which boasts simple implementation and rapid evaluation. While this approach is sufficient when the wavelength to sample spacing ratio is high, significant modulation of the input signal is observed as the spacing increases or the signal frequency decreases, as per Figure 8 in [48].

Astarita *et al*. also investigated a shifted linear interpolation scheme, developed by Blu *et al*.[49], where the sampling knots are shifted by a fifth of the spacing, before being interpolated. This method represents very little additional algorithmic or computational complexity, and is confirmed by Astarita to improve the modulation transfer function when the sample spacing relative to the wavelength is large. On the other hand, performance, where the relative spacing is small, is degraded slightly compared to the standard method.

In addition to the two linear-based interpolation schemes, Astarita investigated a cubic B-spline interpolation and found it to significantly outperform the aforementioned linear approaches, at the expense of computational cost. The cost can be offset by the improved spatial response, though, since fewer vectors are needed to obtain an equivalent description of the underlying flow, in fact this interpolation method is quoted to reduce the overall computation time by as much as an order of magnitude [48]. Furthermore, by adopting such a C2 continuous scheme, well-behaved second derivatives can be obtained. A bicubic interpolation scheme is, therefore, the recommended approach for the majority of applications.

2.12.5 Image interpolation

The efficient way to implement continuous window shifting is to instead deform the underlying image, and perform the window deformation globally. Using the previously described dense predictor, an image can be reconstructed by, for each pixel, looking to where the current pixel will be in the second image. For example: The horizontal displacement at a particular pixel (i, j), from I_A to I_B is estimated to be 4.5px using cross-correlation. Therefore, the *deformed* second image, $I'_B(i, j)$, adopts the value of $I_B(i + 4.5, j)$. This procedure is performed for all pixels to obtain the deformed image.

The cross-correlation between I_A and I'_B is then performed to update the estimate of the displacement field.

This process can be further improved by noting that a central difference scheme is second-order accurate in time, whereas a forward difference scheme is only first-order [50]. Using a central difference scheme, I'_A and I'_B are calculated using minus- and plus-half the displacement, respectively. The displacement field is then updated by correlating the two deformed images.

The choice of signal reconstruction scheme, has an inherent influence on the performance of iterative image analysis routines. Astarita and Cardone [51] performed a thorough investigation into various interpolation schemes, including bilinear, biquadratic, bicubic, and simplex, as well as cardinal sine (sinc), FFT, and B-Spline schemes with varying kernel size. The reader is referred to their article for a thorough analysis of the various methods, with particular attention being drawn to Figure 9. The key conclusions were that if the computational cost is the primary driver, then the simplex interpolation scheme should be adopted, whereas for the best reconstruction accuracy a sinc, FFT, or B-Spline scheme should be used. While the latter may result in a significant increase in cost for the deformation stage, the authors note that for high overlap ratios the time required to correlate all windows may significantly exceed the time required to deform an image, even with one of the high accuracy methods. These results, as well as their efficient implementations [52, 53], have led to B-Spline interpolation becoming the typical approach to reconstructing PIV images.

2.12.6 Predictor corrector filtering

The result of the first pass of cross-correlation is a displacement field which is subsequently used to deform the PIV images. In doing so, the residual displacement should be reduced allowing for better measurement of the displacement between images in the second iteration. The measurement of the residual displacement can then be combined with the previously obtained field to obtain the improved solution. In this context, the displacement field used to deform the images is known as the predictor, whereas the measured displacement of the residual is called the corrector. Following each iteration the corrector updates the predictor, ideally reducing the residual with each pass.

$$f_{2\mathrm{D}}(\xi,\eta) = \begin{cases} C_2\left[\left(\frac{l}{2}\right)^z - |\xi|^z\right] & \text{for } |\xi| \geq |\eta| \\ C_2\left[\left(\frac{1}{2}\right)^z - |\eta|^z\right] & \text{for } |\xi| < |\eta| \end{cases} \tag{2.30}$$

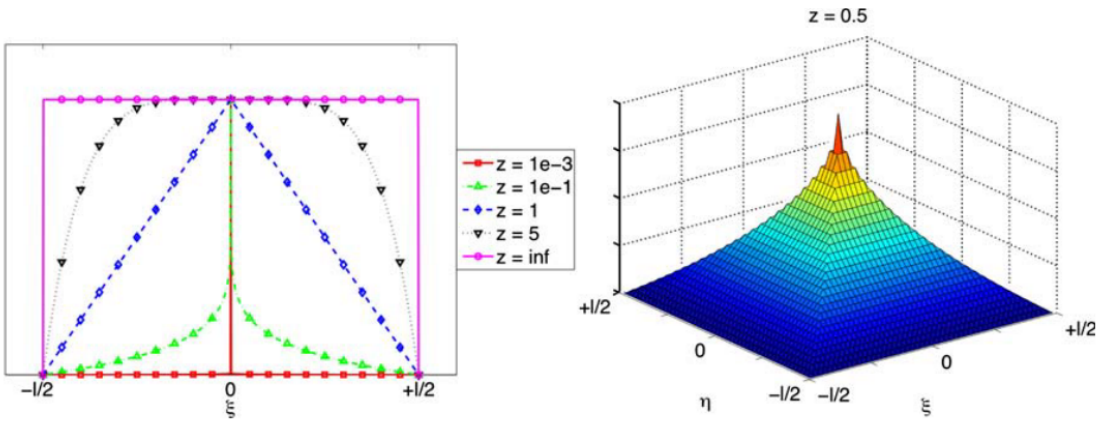
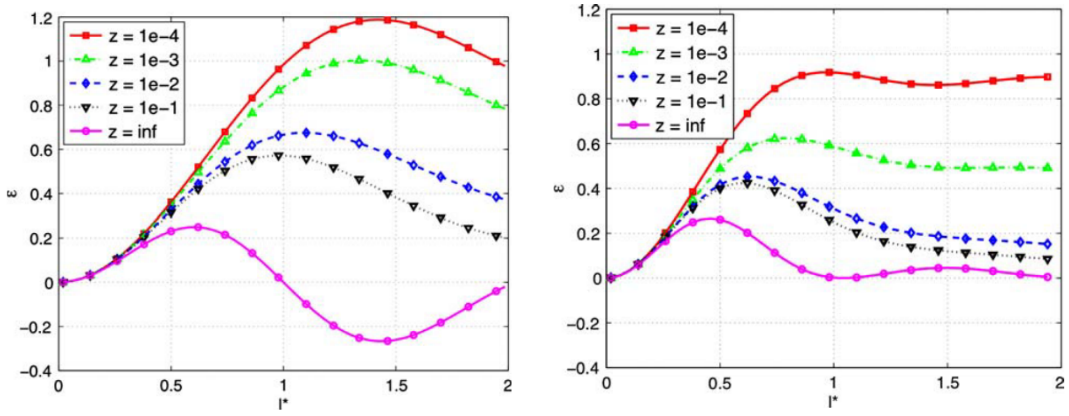


Figure 2.20: Illustration of the family of filters proposed by Schrijer and Scarano. Image from [54]

However, as investigated by Schrijer and Scarano [54], this can lead to instabilities in the solution for certain wavelengths where a negative response is observed, leading to an *increase* in the residual displacement. This instability can be resolved by the use of a spatial filter, at the expense of spatial resolution, for which the authors proposed the family defined by (2.30) wherein larger values of z relate to more spatial filtering with $z = \infty$ being equivalent to a top-hat filter. An illustration of this filter behaviour is shown in Figure 2.20

$$\epsilon = r_f(1 - r_c) \tag{2.31}$$

While this spatial filtering can be applied at either the predictor or corrector stage, it suffices to note that the corrector filtering performed entirely worse than the predictor filtering approach, and so only the latter case is discussed herein. Defining the stability criterion (2.31) as a function of the spatial response of the imposed filter, r_f, and the image weighting filter, r_c, which is normally a top-hat and hence r_c represents $sinc(l*)$, the stability of a range of wavelengths for different filter shapes can be determined. Figures 2.21(a) and 2.21(b) show the stability criterion for a one- and two- dimensionally varying flow field, respectively. Note that stability is ensured for all $-1 <= \epsilon <= 1$, however, for negative ϵ a fluctuating sign is observed. From this, it can be seen that if the filter size is too small, i.e. approaching a Dirac delta, then the process is unstable for one-dimensionally varying flow fields. Using a large spatial filter can, therefore, make the process stable, at the expense of spatial resolution.



(a) (b)

Figure 2.21: The stability coefficient as a function of normalised window size, $l*$, for (a) one dimension, and (b) two dimensions. Image from [54]

2.13 Summary

Once the experimentalist has conducted the experiment, the next, non-trivial, task is to extract the vast amount of information contained in the recorded images using the methods, i.e. cross-correlation, explored within this chapter. Amongst these images, however, sits a combination of useful information and undesirable noise which makes the task of information extraction more difficult.

User-controlled parameters, for example window size, overlap ratio, or number of iterations, strongly influence the ability to separate such useful information from the noise (as per Sections 2.10.3, 2.11, and 2.12.6), yet these parameters are set uniformly over images with spatially varying content, leading to less accurate or, as depicted by Figure 2.16, completely erroneous measurements. While iterative methods, as described in Section 2.12, may help alleviate some of these restrictions, they still heavily depend on uniform, user-defined, parameters.

The motivation for this research is to maximise information extraction regardless of variations in image or flow properties. This can be sought in broadly two ways; via brute force, e.g. more hardware or evolutionary optimsation of existing methods, or through intelligent solutions, and neither should be discounted. Along the first approach, by improving the amount of information extracted per unit resource, it may be possible to improve the total information gained for a given amount of resource. Accordingly, Chapter 3 presents a new technique which can significantly optimise the existing conventional approach to image analysis by reducing the amount of computation

required for a specified level of solution uncertainty.

This brings significant improvements in computational efficiency, yet does not consider adapting interrogation parameters according to local conditions. An intelligent (adaptive) approach to image analysis is, therefore, considered in order to improve the flexibility of the analysis process, such that information may be more optimally extracted. Onesuch challenge of implementing adaptive approaches lies in de-coupling the parameters of the conventional method; most prominently that the vector spacing is governed by the window size and number of iterations, as per (2.14). Chapter 4, therefore, explores the literature on adaptive PIV methods, and attempts to decouple such parameters. Alongside this decoupling, a method for automatically determining *initial* window size, according to the constraints defined in 2.10.3, is proposed which allows for locally optimal correlation window size, while significantly reducing user-dependence and computational cost.

Another downfall of conventional, structured, approaches is that the sampling rate remains uniform, despite varying flow scales and image characteristics. Accordingly, fully unstructured distribution methods are explored in Chapter 5, comparing existing techniques with a novel distribution method, showing the novel method to produce high quality results while remaining more computationally efficient.

Building on these findings, a completely novel approach to PIV image analysis is explored in Chapter 6; semi-structured adaptive sampling. The proposed methodology retains the flexibility offered by fully unstructured approaches, as well as the simplicity and efficiency offered by structured approaches. Chapter 6 investigates the implementation details of such an approach, with a focus on reducing dependency on user-defined parameters.

CHAPTER 3

SELECTIVE SPATIAL ANALYSIS

Data-dependent, adaptive sampling approaches typically involve the spatial adjustment of sampling locations and parameters over the field of view. Such methods already exist within PIV and are explored in significant detail in Chapter 4, and will subsequently be at the core of the remaining chapters of this thesis. However, another approach to improve the information extraction from PIV images, is to reduce the computational cost, such that for a given amount of computational resource, more intensive parameters may be adopted, thereby improving the equivalent data yield. By reducing the extent of the domain which requires analysing, based on statistical definitions of convergence, it is possible to significantly reduce the computational requirements for the analysis of an ensemble of images. In this context, the confidence interval, i.e. statistis, can be used to suggest the amount of available information for each location in the image, allowing regions which have been sufficiently resolved to be neglected by means of a simple image mask. Due to spatial coupling, care must be taken such that the decision to stop analysing a particular region does not negatively affect the surrounding neighbourhood - which still requires analysis. The extent of spatial coupling, and the effect that various analysis parameters have on such coupling is, therefore, investigated in this chapter. The feasibility and benefits of this approach are demonstrated, revealing that a 44% reduction in the number of correlation windows can be achieved for a typical PIV image ensemble.

3.1 Introduction

It has been established that particle image velocimetry (PIV) is used extensively to experimentally investigate the characteristics of fluid dynamics by analysing a pair of images of tracer particles separated by a small time delay, δt. While such instantaneous displacement measurements may be of interest by themselves, PIV is also extensively used to acquire time-averaged solutions of an ensemble of image pairs.

Typically, this time-averaging is applied to the many instantaneous displacement fields, providing the mean solution as well as other statistics about the flow, for example, the standard deviation or kurtosis. Alternatively, ensemble cross-correlation can be utilised wherein the cross-correlation map for each location is averaged over all images, and a single displacement extracted representative of the mean [55]. This approach is frequently used in micro-PIV or when experimental conditions, such as seeding, are poor and would result in instantaneous cross-correlation failing [17]. While both approaches have their advantages and disadvantages, the primary benefit to the former is the ability to obtain statistical quantities about the flow, whereas ensemble-correlation is limited to the mean displacement field.

Such statistics may be of interest to the experimentalist but must be reliable, i.e. converged to the user's requirements. Satisfactory convergence of the mean displacement field may be obtained using relatively few, $N < 100$, image pairs for reasonably steady or straightforward flows, although the number of images may be several times larger for more unsteady flows. Conversely, Ullum *et al*. [56] note that reliable convergence of higher-order statistics may require a dramatic increase in the required number of image pairs towards $N = 20,000$, posing significant demands on computational resources.

While the computational cost to obtain a single mean displacement field may not be significant of itself, it is often desirable to test many parameter configurations, which can rapidly increase the number of mean displacements to be analysed. Meanwhile, access to wind tunnels (WT) is often restricted due to their running costs, availability, or regulatory bodies, as is the case in Formula 1[1] [57]. As such, the experimentalist must often decide which parameter configurations are likely to be the most informative in advance, in order to maximise WT utilisation. Reductions in computational requirements to obtain the desired results may, conversely, enable the user to adjust model-parameter configurations on the fly, based upon results obtained concurrently with WT run-time,

[1]Formula 1 teams are restricted to a certain combined total hours WT run-time or equivalent CPU time for CFD. It is thus the prerogative of the team to decide how to split their allocated hours between experimental and numerical.

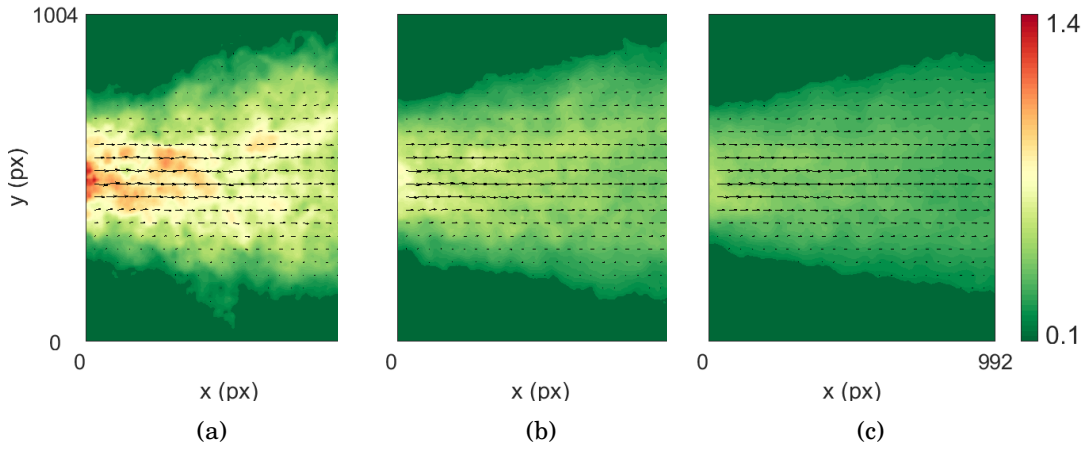


Figure 3.1: Evolution of the confidence interval (95% confidence level) of the mean displacement for an axisymmetric turbulent jet, provided for the second PIV challenge [2], using (a) 20 image pairs, (b) 50 image pairs, (c) 100 image pairs. The confidence interval lower limit is capped at 0.1 for the purpose of illustration. Units in px.

thus offering a considerable benefit towards maximising the efficiency of WT usage.

When calculating ensemble-averaged solutions, the underlying flow should ideally be quasi-stationary, though some temporal fluctuations are unavoidable and the magnitude of such will naturally vary over the domain due to the underlying flow behaviour. Furthermore, flow features typically vary spatially, presenting a range of displacement gradients, which are known to be detrimental to the accuracy of the correlation outcome [58], adding increased uncertainty in the form of artificial displacement fluctuations. Additionally, due to the non-linearity of the cross-correlation response, c.f. Section 2.10.4, the displacement will be biased towards regions of curvature [35]. This produces a constant bias to the mean and also influences the measured temporal variance, since the amount of bias per-image basis will vary depending upon the realisation of particle images within the correlation window. Due to these factors, the number of image pairs required for local statistical convergence will vary over the field of view (FOV); regions with little temporal variance (e.g. laminar flow), will require fewer samples for statistical convergence than more turbulent, temporally varying, regions.

Spatially adaptive sampling strategies, that will be explored in later chapters, may ameliorate the situation by locally reducing the number of correlation windows [59, 60], although don't consider local convergence and will continue to place windows, albeit fewer of them, in regions which may already be satisfactorily converged.

Confidence intervals provide an insight into the convergence of a statistic, by indicating the bounds within which there is certain confidence that the true statistic

value lies. If the width of the interval is smaller than some user-defined threshold value, then the solution at that location can be considered sufficiently accurate for the user's needs. Figure 3.1 shows the magnitude of the 95% confidence interval (CI) for a turbulent jet flow, as provided with the Second International PIV Challenge [2], as the number of images contributing to the ensemble increases. Regions in which confidence in the solution is sufficient, i.e. regions in Figure 3.1 where the CI <= 0.1, can be considered to have reached statistical convergence.

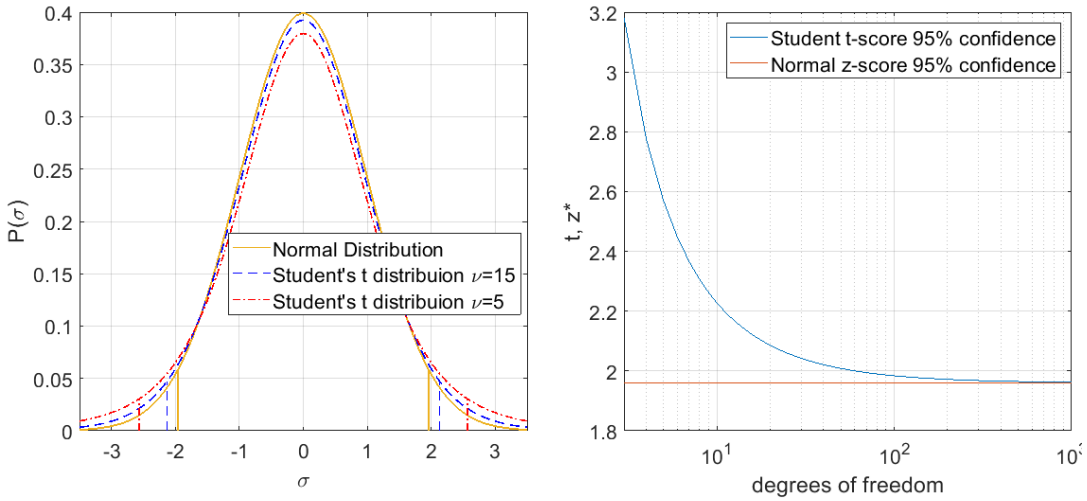
The confidence intervals can be computed pixelwise[2] to identify preceisely regions still requiring further sampling, e.g. the central jet flow in Figure 3.1, which is herein referred to as the region of interest (ROI). In theory, the image outside of the ROI need not be sampled any further, which could be achieved by imposing a sampling mask in a similar methodology as image masks are applied. Applying a mask immediately surrounding the ROI, however, would artificially truncate information along the flow-mask boundary, something which has repeatedly been found to influence correlation [61, 62, 63]. To avoid such truncation, the sampling mask should be dilated until these effects are no longer observed, and such a region shall herein be referred to as the analysis support radius (ASR).

This chapter first explores how confidence intervals can be calculated, and subsequently how they allow automatic detection of the ROI within the image. Afterwards, a novel methodology is introduced which allows calculation of the ASR for arbitrary analysis configurations. The methodology is then proven on a set of experimental images of the flow over a blunt trailing edge, showing the potential computational savings. While remaining simple, this method is able to significantly reduce the total number of correlations required for the analysis of a typical image ensemble.

3.2 Background Theory

Broadly speaking, the CI represents the range around the observed statistic mean within which there is a certain probability, α, of the true statistic mean occurring. More specifically, the confidence level indicates the proportion of calculated confidence intervals which would contain the true value, if infinitely many intervals were calculated from the same underlying distribution [64]. Increasing the required confidence level will increase the width of the interval - thus giving more chance of the true value being within the

[2]Discussion relating to the need to calculate pixelwise is provided in Section 3.3.4



(a) (b)

Figure 3.2: Comparison of the Normal- and Student's t- distributions, and the effect of the number of degrees of freedom, ν, on calculating the confidence interval

calculated bands; a 100% confidence level, $\alpha = 1$, although not realistic, would result in an infinitely large interval to be certain of containing the true value.

The equation to calculate the CI varies depending on whether the standard deviation of the population is known or simply estimated from a sub-sample of the population. The equations for each situation are shown in (3.1a) and (3.1b), respectively, where N is the number of samples, σ_u is the true population standard deviation, S_u is the observed sample standard deviation. The critical values $z^* = f(\alpha)$ and $t = f(\alpha, N)$ are derived from the standard normal distribution or the Student's t-distribution, respectively, and will be covered in more detail in the remainder of this section. These equations would produce a CI $\overline{u} \pm CI_{\overline{u}}$ where $\overline{u}$ is the observed mean of the data. Equation (3.1b) should also be used when the sample size, N, is small to more correctly reflect the statistical probabilities.

$$CI_{\overline{u}} = \begin{cases} z^* \dfrac{\sigma_u}{\sqrt{N}}, & \text{if } \sigma_u \text{ is known} \quad (3.1a) \\ t \dfrac{S_u}{\sqrt{N}} & \text{if } \sigma_u \text{ is estimated with } S_u \quad (3.1b) \end{cases}$$

The aforementioned critical values represent the width, in standard deviations, either side of the mean such that the integral of the probability density function (PDF) in this region is equal to the imposed confidence level, α, see Figure 3.2(a). For the standard normal distribution, with $\mu = 0$ and $\sigma = 1$, depicted in Figure 3.2(a), $z^* = f(\alpha)$, where

common values of α are 0.68, 0.95, and 0.997 which correspond to z^* values of 1, 1.96, and 3 respectively. Alternative values can be found in lookup tables, or by numerically solving the cumulative density function (CDF) of the standard normal distribution for some target α, for which functions are readily available in most public statistics libraries.

For the Student's t distribution, on the other hand, $t = f(\alpha, \nu)$ where $\nu = N - 1$ and represents the number of degrees of freedom. When there are few degrees of freedom, there is more uncertainty in the estimate of the standard deviation; the t-distribution accounts for this uncertainty by having larger tails, allowing for more samples further from the mean. This can be seen in Figure 3.2(a) by the increased probability for large σ, and decreased probability for low σ. Consequently, the critical value t must be larger, though $t \to z^*$ as $N \to \infty$, and this can also be seen in Figure 3.2(b). Similarly to above, values of t for various degrees of freedom can be found in lookup tables or calculated with functions from readily available libraries.

It is, of course, possible to calculate the standard deviation using the simple equation shown in (3.2). However, this requires storing all the displacement fields, $u_i(x, y)$, where the notation (x, y) is dropped for clarity, and re-calculating for every additional displacement field.

$$S_{u,N} = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(u_i - \overline{u})^2} \tag{3.2}$$

Efficient and stable updating algorithms exist for both the mean and variance, which can calculate these values without the need to store all contributing data [65]. In (3.3), u_N signifies a component of the N^{th} displacement field, $\overline{u}_N$ is the mean of all N samples (u_i with $i = 1 \ldots N$) and ${S^*}^2_{u,N}$ is the sum of squared differences to the mean, which estimates the variance, $S^2_{u,N}$, via the relationship $S^2_{u,N} = {S^*}^2_{u,N}/(N-1)$.

$$\overline{u}_N = \overline{u}_{N-1} + (u_N - \overline{u}_{N-1})/N \tag{3.3a}$$

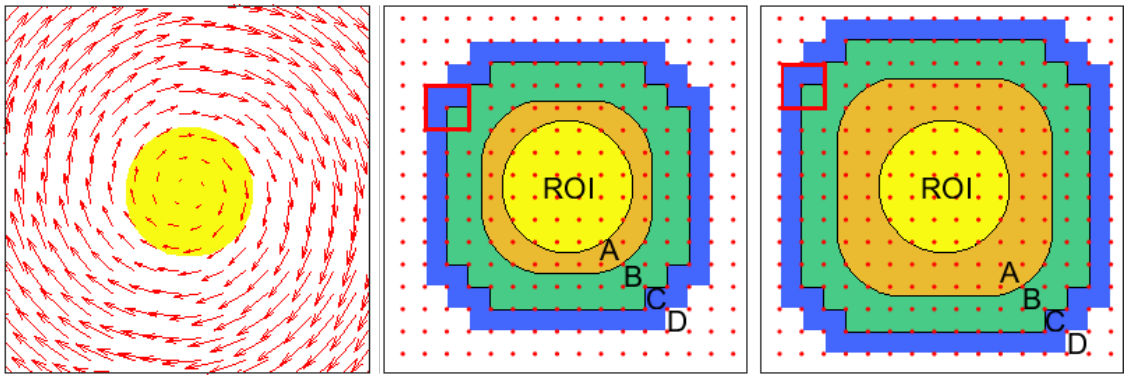
$${S^*}^2_{u,N} = {S^*}^2_{u,N-1} + (u_N - \overline{u}_{N-1})(u_N - \overline{u}_N) \tag{3.3b}$$

Having estimated the sample standard deviation, S_u, the confidence interval can then be calculated for each vector location using (3.1b), as in Figure 3.1. Comparison of this confidence interval to some threshold value is then used to indicate convergence. For PIV applications, a suitable threshold value of the mean could be 0.1px, in line with the commonly assumed uncertainty of PIV [39]. Although this uncertainty level depends on many parameters, it serves as a reasonable threshold to demonstrate the presented methodology and is generally appropriate for uncertainty in the mean displacement.

It should be emphasised, however, that this parameter should be selected by the user depending on the particular conditions of the experiment, analysis procedure, and on the statistic of interest. It is worth noting that the presented methodology can be extended to essentially arbitrary statistic properties. Doing so requires calculating the standard deviation of the statistic in question, which can be obtained formally using standard expressions of variance [66], or may be approximated using techniques such as bootstrapping [67, 68].

In the extensive paper by Sciacchitano and Wieneke [69], the authors discuss how instantaneous uncertainty propagates to ensemble statistical uncertainty. In their paper, the uncertainty of the mean is shown to decrease with increasing effective number of independent samples, N_{eff}, and increase as the local, spatial, standard deviation[3] relative to the displacement magnitude increases, i.e. as the instantaneous uncertainty increases. In this context, $N_{eff} \leq N$ and accounts for any time-dependency between successive snapshots, such as in time-resolved data sets. Understanding the increase in temporal uncertainty of the mean due to instantaneous uncertainty can, and should, be used to aid selection of the user-imposed threshold in the currently proposed methodology. Since the proposed methodology considers only the measured temporal fluctuations, which contains both the actual flow fluctuations and measurement error, the imposed threshold should be appropriately reduced if a certain confidence in just

[3]This refers to the amount of displacement fluctuation within a region of space, as opposed to the level of fluctuation over time for a particular pixel



(a) (b) (c)

Figure 3.3: Depiction of the analysis support radius for an arbitrary region of interest (yellow), showing; the interpolation kernel A, the vector validation kernel B, image deformation kernel C, and finally the region with no influence over the ROI, D, for both bi-linear, (b), and bi-cubic, (c), interpolation kernels. Red dots represent sample locations, for a 50% overlap ratio, while the red square depicts an individual correlation window.

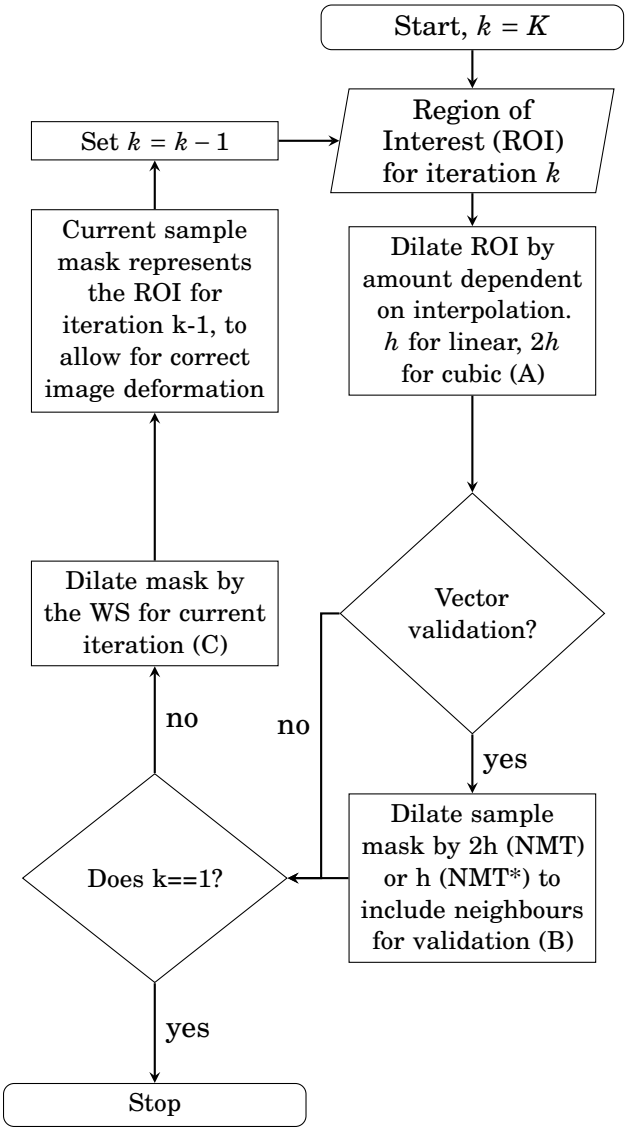


Figure 3.4: Flow chart showing masking process. Note that k corresponds to the iteration number in the WIDIM analysis, and that masks are created working backwards from the final WIDIM iteration, due to the forward dependency of iterations. Letters in parentheseses reflect the regions depicted in Figure 3.3

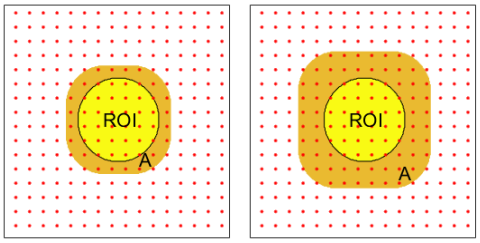
the underlying flow fluctuations is desired. The amount to reduce the threshold can be approximated by the uncertainty mean square, which can be found using accurate uncertainty quantification methods such as those in [70].

3.3 Methodology

Using the theory introduced previously, the ROI can be determined by comparing the CI, for each vector location, to the user-defined threshold. The subsequent goal is to minimise the computation required to resolve the displacement at these locations, without artificially introducing numerical artefacts associated with image masks. Each sample, i.e. correlation window, will have a support radius, termed the analysis support radius, within which changes to the sampled value will influence the final solution in some way, yet beyond such distance, there will be no effect on the solution. A typical analysis algorithm, e.g. WIDIM [15], can be roughly divided into 4 steps 1) interrogating the image at pre-defined locations, 2) validating the results of such interrogation by comparing each vector to its neighbours, 3) interpolating the validated vectors onto a pixelwise grid, called the dense predictor, 4) deforming the raw images according to the dense predictor. Steps 1-4 are then repeated several times to improve the spatial resolution, by reducing the interrogation window sizes, and/or inter-sample spacing, i.e. vector-pitch. The methodology presented herein aims to minimise the number of correlation locations in step 1, while ensuring that the solution at step 3, within some arbitrary region, is unaffected. Note that although the initial region of interest is defined only at vector locations, intermediate steps in the algorithm must be able to handle a pixel-wise defined ROI, as will be explained in the remainder of this section. Accordingly, an overview of the methodology, for an arbitrary circular ROI, is depicted in Figure 3.3, with each component being further broken down and discussed in isolation. Due to the forward dependency of PIV analysis algorithms, i.e. that later iterations in the analysis rely on the output of previous iterations, the proposed method obtains sampling masks in reverse order. In other words, the minimum region that must be sampled in the first analysis iteration can not be determined until it is known which regions are required in the second iteration, which in turn relies on the third iteration, and so on. To avoid confusion, that the *first* sampling mask obtained is to be used in the *last* analysis iteration, any references to a particular iteration refer exclusively to the analysis iteration in which the mask is applied.

3.3.1 Interpolation

The methodology proceeds by working backwards, step-by-step, from the desired final solution to the very first correlation. Referring to the previously described steps 1-4, it is noted that in the final iteration of a typical PIV algorithm steps 3 and 4 may



(a) (b)

Figure 3.5: Intermediate sample mask when the ROI is dilated for vector interpolation for both bi-linear, (a), and bi-cubic, (b)

often be neglected, provided the interpolated displacement field is not required, e.g. for post-processing, nor the final deformed image required, e.g. for particular uncertainty quantification [71]. While dilation of the sampling mask to account for interpolation may be omitted if the displacements are only required at vector locations, in the case that the ROI is defined per pixel, as it is for intermediate iterations (see Section 3.3.3), the sampling mask must be dilated to consider the spatial requirements of interpolation. Typically, bi-cubic interpolation is used to obtain the dense predictor from vector information, balancing speed and accuracy [48], as well as being simple to implement. More accurate schemes, such as cardinal sin or FFT interpolation, can be used at the expense of computation, whereas, bi-linear interpolation may be used to accelerate analysis at the expense of accuracy. While the choice of scheme is ultimately left to the user, the principles described herein can be applied to any scheme, provided that the interpolation kernel can be determined a-priori, i.e. is not data-dependent, and is finite in nature. Due to their popularity, this research shall focus on bi-linear and bi-cubic interpolants.

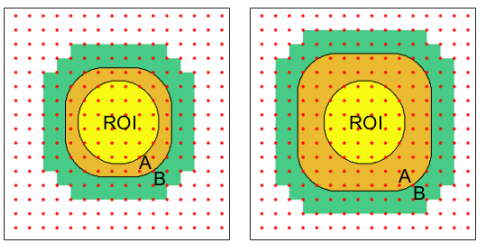
The first step in finding the ASR is, therefore, to find the vectors whose interpolation kernel overlaps with the ROI. For bi-linear interpolation, each pixel is only influenced by the four immediate neighbouring vectors, which form a 2x2 stencil. Dilating the ROI by the inter-sample distance, h_k, where k is the iteration number, in both x and y, therefore ensures that all vectors which would influence the interpolation within the ROI are contained within the dilated region. This region is depicted by region A in Figure 3.5(a). In contrast, bi-cubic interpolation requires a 4x4 stencil and hence the ROI must be dilated by $2h_k$ to ensure all influencing vectors are contained within the new region. This is depicted by region A in Figure 3.5(b). These regions indicate all the samples which are needed to obtain the correct interpolation within the ROI.

3.3.2 Vector Validation

Prior to interpolation, though, vectors are typically subject to validation, using schemes such as the normalised median threshold test (NMT) [39] to detect outliers. Since validation may influence the values of samples within the region A, all of the information required for this step must be collected. Vector validation can be broken down into two steps; invalid vector detection, and vector replacement. The NMT compares each vector to its eight surrounding neighbours from a 3x3 grid, if the vector disagrees with its neighbours beyond some pre-defined threshold, then it is replaced by the analysis routine. The replacement vector is typically calculated as the mean of the valid eight neighbours, or zero if all are invalid. Therefore, to correctly *detect* outliers within region A, the sampling mask must be extended to include the immediately neighbouring vectors, i.e. dilate region A by a further h_k. This intermediate region shall be denoted as region B'. However, if a vector at the edge of region A is determined to be an outlier, it must be replaced by the average of its neighbours, *excluding any outliers*. This, of course, is logical; values that have been identified as being erroneous shouldn't be considered. Yet this requires validation of vectors at the limit of region B', which itself requires the neighbouring eight vectors to be present. As such, for uninhibited validation, region A must, in fact, be dilated by $2h_k$. The resulting region, region B, can be seen in Figures 3.6(a) and 3.6(b)[4].

While dilation by $2h_k$, to obtain region B, is required to guarantee numerical equivalence to an equivalent conventional solution, dilating by h_k is generally sufficient for approximate results. This is for two reasons; firstly, the final solution for instantaneous image analysis is only affected if a vector at the perimeter of region

[4]It is sometimes the case that invalid vectors in the final iteration are not replaced with interpolation, but simply omitted from statistics. In such a case, dilation by h_k, to obtain region B', would be sufficient in the final iteration.



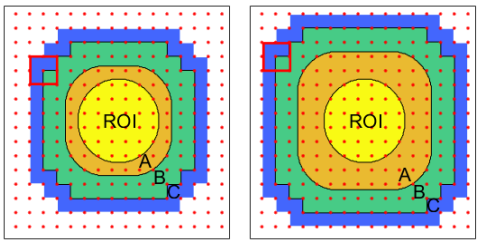
(a) (b)

Figure 3.6: Intermediate sample mask when the ROI is dilated for vector validation for both bi-linear, (a), and bi-cubic, (b), vector interpolation

A is identified as an outlier, which represents only a small proportion of the samples. Samples not on the edge of region A would have access to vectors $2h_k$ away and would be unaffected. Secondly, vectors along the perimeter of region B are still guaranteed to have at least three immediate neighbours, and typically at least five, which can be used for an approximate vector validation attempt. Furthermore, the influence of outliers on the solution is likely to be corrected, at least partially, by subsequent iterations, and therefore only outliers on the perimeter of region A in the final iteration are likely to cause any significant disturbance to the ensemble solution. Herein, dilation by $2h_k$ for vector validation, ensuring numerical equivalence, will be referred to using NMT, whereas dilation by h_k, accepting some numerical differences, will be referred to as NMT*. Differences of the ensemble mean values using NMT compared to NMT* are typically $< 1\%$, i.e. at least two orders of magnitude smaller than the imposed acceptable threshold, yet, require fewer correlations to obtain.

3.3.3 Image deformation

Considering that many PIV analyses make use of iterative routines to obtain improved spatial resolution [72], the implications of image deformation must also be considered. or each of the locations which must be correlated in iteration k, i.e. region B, the correlation response must be identical to an equivalent complete analysis routine. Accordingly, each pixel within each correlation window in region B, iteration k, must be appropriately deformed. This region can be found by convoluting each sample location in region B with the current window size (WS), as indicated by region C in Figures 3.3(b) and 3.3(c). Since image deformation, following the methodology of [52], requires the predictor for iteration $k-1$ at each of these pixels to be correct, region C effectively serves as the ROI for iteration $k-1$. The process is repeated until the first iteration is reached and a



(a) (b)

Figure 3.7: Intermediate sample mask when the ROI is dilated for image deformation for both bi-linear, (a), and bi-cubic, (b), vector interpolation

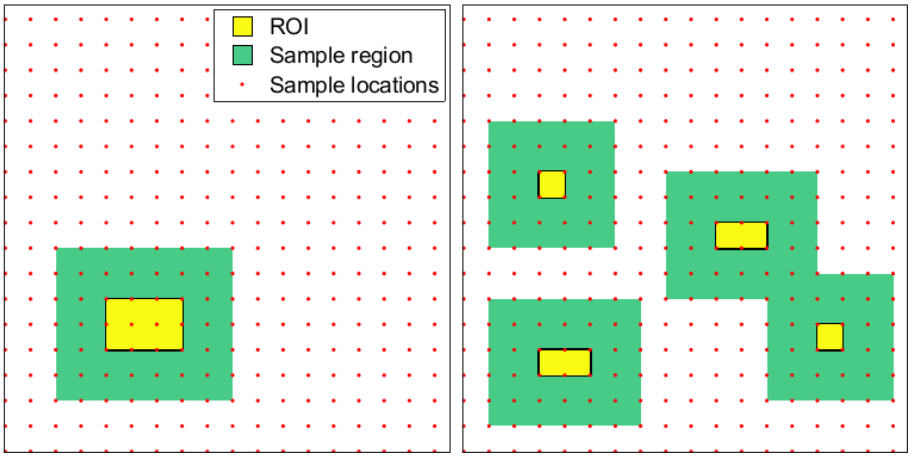


Figure 3.8: Influence of the composition of the ROI on the number of samples required for (a) a single contiguous ROI, requiring 56 samples (red dots) using the proposed method, and (b) several distributed ROIs, requiring 150 samples, for an equivalent cumulative extent of ROI. Note that colours are the same as in Figure 3.3. To simplify the illustration, a linear interpolation is imposed, which results in the interpolation kernel (which would be orange) being identical to the ROI in this case, since to the linear kernel does not extend beyond sample locations.

sample mask is obtained for each iteration. The described approach is presented as a flowchart in Figure 3.4, and with each specific sampling region graphically highlighted in Figure 3.3 for clarity. A standard image analysis routine can then be modified to correlate only sample locations which lie within this region and ignore everything else, i.e. region D, with no negative effects on the solution in the ROI.

Implementation

In its simplest form, the proposed sampling masks may be imposed in the same way that a conventional image mask is applied, i.e. skipping correlations where the window is in the inactive region, requiring minimal code modifications. Yet, further modifications to various components of the analysis process, e.g. interpolation and image deformation, which continue to operate on the entire image, can yield further optimisations. It is relatively straightforward to modify vector validation and structured interpolation codes to consider only the active region. In doing so, the computational cost for these processes is proportional to the number of vector locations, i.e. the number of cross-correlation operations. Image deformation requires interpolating the image based on the predictor values over the image. This image interpolation operates pixel-wise and hence the

code can be easily modified to exclude pixels outside of the sampling mask from the deformation process, resulting in a computational cost scaling with the number of pixels in the ROI. When using B-Spline image interpolation, as recommended by Astarita for the best accuracy [51], efficient implementation relies on a finite-impulse response (FIR) filter, which is applied to the entire raw image concurrently [53]. Such filtering is not be easily reduced to arbitrary sub-regions, however, this step is only required once per image and yields a marginal computational overhead. Finally, calculation of the sampling mask can be obtained through a series of binary image dilations, wherein the ROI or sampling region is denoted by True (1) and False (0) elsewhere, which can be optimised using bitwise logical operators and are computationally cheap [73]. Hence the methods computational savings scales roughly linearly with decreasing ROI extent.

Since the calculation of the standard deviation requires at least two data points, this methodology can, in theory, be implemented from the third image pair onwards. In practice, however, confidence intervals calculated with such few samples are likely to be large, and would accordingly necessitate ubiquitous sampling. To mitigate unnecessary computation, the first $N_t \approx 10$ timesteps are therefore obtained using a conventional structured approach, without computing confidence intervals. This number effectively represents the lowest number of images which are expected to be required for convergence of any pixel. Values of $N_t = 10 - 30$ are reasonable if mean displacements of reasonably steady flows are sought, though may be considerably larger if higher-order statistics are sought.

Furthermore, it is generally not worth proceeding with the temporally adaptive approach if the size of the ROI is $>\approx 95\%$ of the image. In part, the overhead to calculate the sampling masks must be considered, although this is small. Though more importantly, the ASR is typically larger than the ROI, particularly in the intermediate iterations of the analysis, implying that the vast majority of the domain is likely to be sampled anyway and therefore computational savings are likely to be small.

3.3.4 Distinction of pixelwise and vectorwise calculations

Traditionally, PIV statistics are calculated at each of the vertices of the sampling grid, i.e. at vector locations. These results may subsequently be interpolated to the pixelwise grid, though this is typically just for visualisation purposes. Accordingly, determination of the initial ROI is also limited to and calculated on the coarse sampling grid.

However, as explained in Section 3.3.3, the ROI for intermediate iterations must be defined pixelwise since the entire extent of each correlation window must be considered

in the previous iteration. Pixelwise ROI's must, therefore, be tolerated within the recursive mask generating algorithm and hence the algorithm has been designed with this generality in mind.

The initial ROI can be evaluated on a pixelwise grid by either:

- Interpolate the displacement field pixelwise and evaluate the CI for each; the ROI is then extracted by pixelwise comparison to the threshold
- Evaluate the CI for each vector location and interpolate onto each pixel; the ROI is then extracted by pixelwise comparison to the threshold

Noting that since the operations involved are all linear, either approach should yield the same ROI.

3.3.5 Extension to unstructured

Considering the goals of the remainder of this thesis, application of this method to an unstructured, adaptive, sampling algorithm is desirable. In theory, it is possible, however, particularly stringent requirements must be enforced making it impractical in reality. The first requirement, is that the desired sampling density must be known *a priori*İnstantaneous adaptivity is, therefore, prohibited, since the sampling density would change with each iteration as the objective function is refined. Alternatively, statistical adaptivity as per [74], wherein an ensemble objective function is obtained and kept fixed for each image in the ensemble, remains viable. In addition to the sampling density, the number of samples per iteration must be prescribed, such that the desired inter-sample distance may be calculated per pixel (see Chapter 5 for details about this process).

As in the structured approach, the first step is to obtain the region within which, samples will affect the interpolation within some arbitrary ROI, i.e. region A as per Figure 3.3. For a structured approach, the interpolation kernel is easy to obtain *a priori* and is dependent on the inter-sample spacing. Unstructured interpolation kernels, on the other hand, are not as simple to obtain. When using a triangulation based interpolant, the particular triangulation will vary depending on the arrangement of samples, and as a result, so will the interpolation kernel size, with respect to the image. By making certain assumptions about the expected arrangement of samples, based on the desired inter-sampling distances, it is feasible that a *conservative* interpolation region can be obtained. Such a region would guarantee that samples beyond the region have no influence on the ROI, but not necessarily that all samples within the region influence the

ROI. Unstructured interpolants with compact support are available, such as RBFs [75], however, these typically adjust the support radius based on the distances between nodes, for example based upon the largest inter-sample distance. Since the desired inter-sample distance is known in advance, from the objective function, it may be possible to select a support radius *a priori* allowing region A to be obtained.

The second step in the process relates to vector validation, which, again, requires *a priori* knowledge of the sampling distribtuion. Considering the unstructured vector validation routine requires the eight nearest neighbours (see Section 4.4.5), a conservative region B may be obtained similarly to above, based on the desired inter-sampling distances. Again, such a region would guarantee that no samples beyond the region would be selected as neighbours, but not necessarily that all samples within the region would be required for the vector validation step. Alternatively, a fixed search region may be defined, e.g. some multiple of the local WS, within which all samples are used as neighbours. On the one hand, this allows a more optimum definition of region B, yet on the other, requires careful selection of the search radius such that an appropriate number of neighbours is obtained. Regardless of the approach, it remains feasible to also obtain region B.

Finally, image deformation can be considered similarly to the structured approach, convoluting the local WS for each pixel in region B, although this prohibits the adaptation of WS to the local flow behaviour. As such, given certain modifications, it is indeed possible to use this approach to obtain a sampling mask for an adaptive analysis routine, though fundamentally limited to statistical adaptivity. Furthermore, at each step of the masking process concessions must be made which increase the extent of the sampling mask. This has a significant knock-on effect for iterative methods wherein the mask for iteration k acts as the ROI for iteration $k-1$, as in Section 3.3.3, considerably hindering any attainable speed-ups. As such, while it is theoretically possible to apply this method to an adaptive routine, it is not recommended.

3.3.6 Application to non-steady flows

Although this methodology is best suited to quasi-steady flows, or more specifically flows with temporally uniform fluctuations, it can be applied to more unsteady flows albeit with a degree of caution. When the fluctuations are uniform in time, the estimation of the mean and confidence interval are representative of the entire duration of the experiment; once the confidence in the statistic (e.g. mean) is sufficient, sampling that location may stop and the observed mean will reflect the true mean within this confidence. However,

when the fluctuations vary in time, it possible that the observed mean and CI for the first n is not representative of the true mean and CI for the full N instances. This could lead to the sampling being prematurely stopped.

The simple solution to this issue is to randomise the order in which the images are analysed, assuming that the samples are independent of one another. In this way, should the fluctuations vary throughout the duration of the experiment, the influence on the standard deviation, and hence CI, will be observed. Additionally, depending on the brevity of such transient behaviour, it may be prudent to increase the initial number of images that are indisciminately analysed, i.e. N_t as described previously. For the mean statistic, the equation for the standard deviation is equivalent regardless of the distribution's shape; provided the samples are drawn from the distribution randomly the CI will be valid. For other statistics, however, the formula for the standard deviation of that statistic often has a generally valid version, and a version simplified under the assumption of a normal distribution [66]. Therefore, when using the general formulation for the statistic's standard deviation and when randomising the order of image analysis, the estimate of the statistic will be valid, regardless of temporal fluctuations, *within the bounds of the imposed confidence*.

3.4 Results

3.4.1 Numerical investigation

According to this methodology, the ASR is influenced by the following parameters; correlation window size (WS), window overlap ratio (WOR), number of iterations wherein the WS and vector spacing is refined (N_k), number of iterations wherein the WS and spacing are kept constant, (N_{kr}), interpolation methodology (linear or cubic), vector validation methodology and parameters (for example, NMT or NMT*) Changing such PIV interrogation parameters will naturally affect the extent of the resulting sampling masks. For example, cubic interpolation requires dilation by $2h_k$, instead of h_k for linear interpolation, therefore demanding more correlations to be performed for a given ROI (Figure 3.3). In addition to the parameter configuration, the constitution, i.e. the size and shape, of the ROI will also influence the total number of correlations. Figure 3.8 demonstrates this by comparing the number of samples required for a single block of pixels acting as the ROI, to the number of samples that would be required if the same number of pixels were to represent a more distributed ROI. Due to the practically

Table 3.1: Values used for each setting as part of the parameter sweep

Parameter	Default	Parameter sweep values
Initial Window Size	97	N/A
Final Window Size	25	(15, 35, 55)
Window Overlap Ratio	0.5	(0, 0.5, 0.75, 0.85)
Number of main iterations	3	(2, 3, 4)
Number of refinement iterations	1	(0, 1, 2, 3, 4)
Interpolation method	Linear	(Linear, Cubic)
Vector validation method	NMT	(none, NMT, NMT*)

infinite number of ways to compose the ROI, quantifying this behaviour is difficult, yet a generally valid tendency is that fewer, larger, contiguous ROIs, (which are typically encountered in practical situations), will be more effective at reducing the required number of correlations than many scattered regions.

To better understand the extent to which each parameter affects the total number of correlations required for a given ROI size, a parameter study was conducted, within which the shape of the ROI was kept as a single central square. This parametric study intends to identify which analysis parameters have the most impact on the size of the sampling mask. To investigate, a default parameter configuration was constructed, as shown in Table 3.1. For each parameter, a range of typical values were then applied while keeping the other parameters fixed, over a range of ROI sizes. The resulting number of correlations were then recorded and compared to the total number of correlations required for full analysis with the same settings.

To simplify the discussion, the dilation of the ROI per iteration can be approximated as some scalar multiple of the sample spacing, i.e $\eta \cdot h$. For example, a single iteration with linear interpolation (dilation by h) and NMT validation (dilation by $2h$) would result in $\eta = 3$. When varying the interpolation kernel or vector validation method, η is modified, whereas h remains constant. Accordingly, there is an approximately constant increase in the fractional number of correlations, i.e. the number of correlations performed as a fraction of the total possible for that particular parameter configuration, shown in Figures 3.9(c) and 3.9(f). Similarly, the more iterations there are, the more times that the ROI is dilated by $\approx \eta \cdot h$, thus we see again a steady increase in the fractional number of samples as the number of iterations increases.[5] Conversely, parameters such as the WS and overlap ratio alter the spacing, h, instead of η, and can have a

[5]The value of η would vary slightly here since an additional dilation based on the WS is required to deform the image correctly, see Figure 3.7

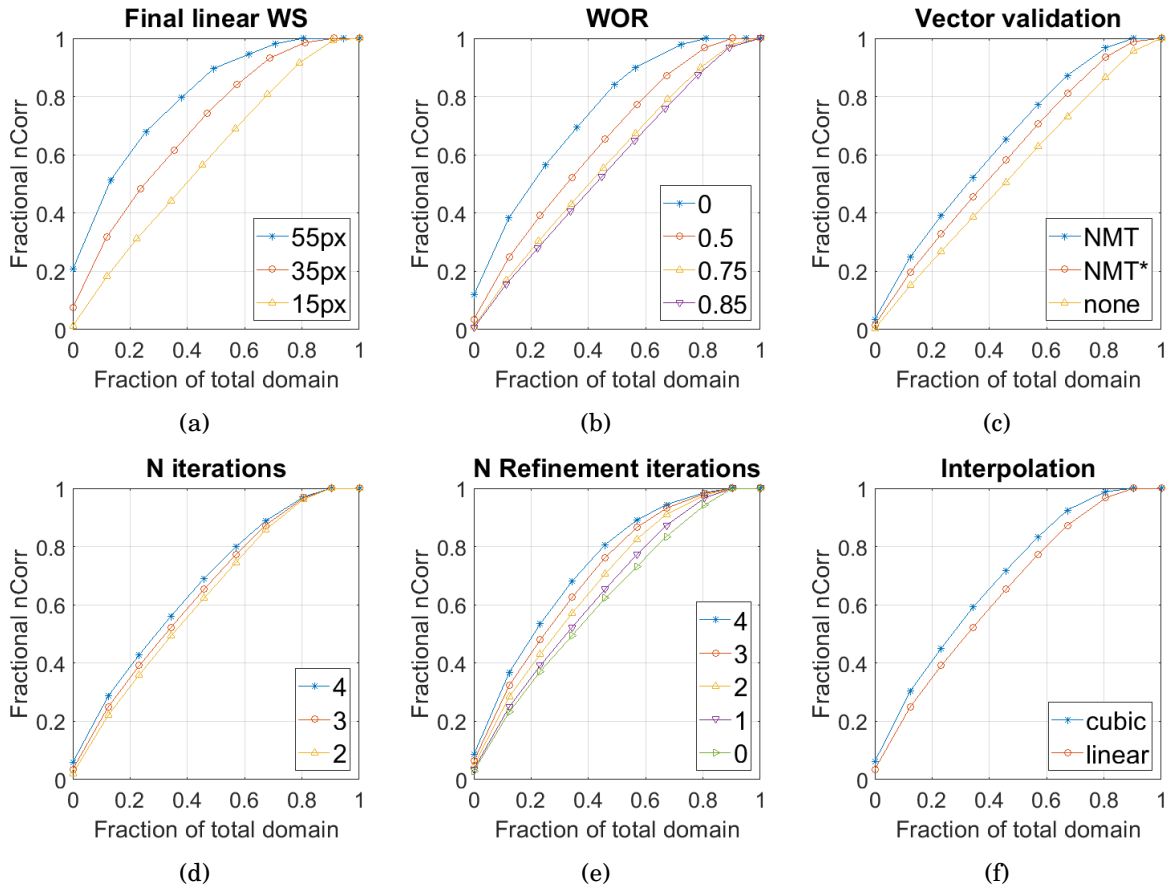


Figure 3.9: Comparison of the speedup effect of a locally selective WIDIM approach for changing (a) final WS. (b) WOR. (c) Vector validation approach. (d) Number of Iterations. (e) Number of Refinement Iterations. (f) Interpolation method.

much more significant impact on the *fraction* of correlations which need to be performed for a given ROI, as shown by Figures 3.9(a) and 3.9(b). The spatial dependency of the analysis is reduced, and hence the computational cost more closely correlates with the amount of domain to be analysed. Put differently, the smaller spacing increases the total number of sample locations in the entire domain, yet the absolute number of samples required for a particular ROI remains approximately the same[6], increasing the number of samples which are now *not* needed to be correlated, significantly improving its efficiency. The key corollary from this study is that more computationally intensive interrogation parameters, with fewer analysis iterations, will benefit the most from the presented methodology.

[6]The number of samples required will only increase if, due to the increased sampling density, there are more contained within the ROI itself.

3.4.2 Experimental analysis

To verify the methodology in a real application, it was applied to the flow behind a blunt trailing edge, investigated by [76] for its aeroacoustic properties. In this study, the effect of trailing edge bluntness on the generation of wake-vortex noise was investigated using Particle Image Velocimetry. The case treated in the following is that of a bluff body with a bluntness thickness, h_b, of 46mm and freestream velocity of approximately 11.6m/s, equivalent to a chord-length (350mm) based Reynolds number of 2.75x10^5. Shown in Figure 3.10 is the reference flow field for this particular case. Experiments were performed by means of Dantec Dynamics planar PIV system in the low turbulence wind tunnel of the University of Bristol, which produces background RMS turbulence levels below 0.03%. It is worth noting that the exemplary PIV recording shown in Figure 3.11 combines image recordings of two, simultaneously triggered, FlowSense EO cameras, each with a sensor of 2072x2072px. The time delay between snapshots was chosen to achieve a freestream displacement of approximately 4px. The first field of view (FOV) observed the boundary layer near the trailing edge while the second covered the near-wake region. Each FOV corresponded to an observed field of view of 161mm x 161mm, equivalent to approximately 3.5h_b x 3.5h_b. The reader is referred to [76] for further details regarding the PIV setup and experimental parameters.

Since the FOV contains both uniform flow and regions of more turbulence, these

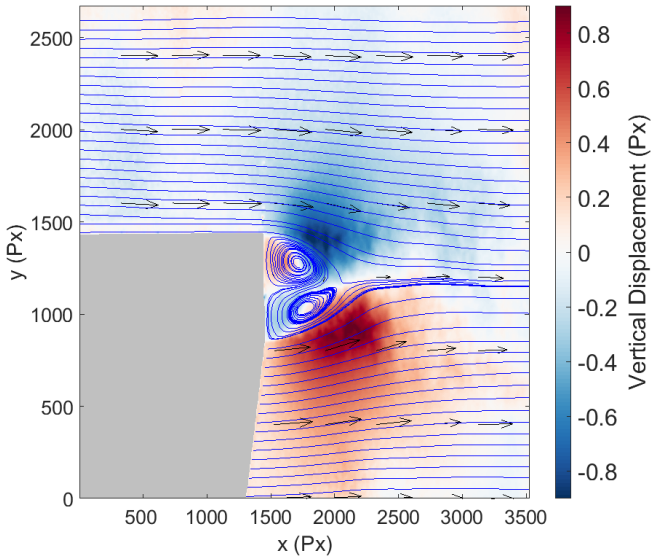


Figure 3.10: Ensemble averaged, reference flow field obtained by averaging all 115 measured displacement fields in their entirety. The contour represents the vertical displacement component. Units in px.

images serve as ideal candidates to test the proposed method. Figure 3.11 shows one of the (in total) 115 raw images and highlights the strong reflection present in nearly all of the images. While the presence of such reflections would not adversely affect the methodology presented within, reflections may still influence the solution accuracy. Since the reflection varies slightly in time, correlations in this region will be influenced to a varying degree per snapshot - leading to increased RMS, or temporal standard deviation. Furthermore, it is expected that some correlations in this vicinity would fail, producing outliers. While the majority of these outliers should be captured by the outlier detection algorithm, some will persist - further increasing the RMS at these locations. Referring back to (3.1b), this would increase the number of samples required for a given CI, therefore, if the reflections can be mitigated with simple pre-processing, then both the solution and computational performance improve. An ensemble-minimum background subtraction [17] was therefore applied to all images, which removed the majority of the reflections (Figure 3.11(b)). Nevertheless, remnants remain in the images due to the temporally fluctuating nature of the reflection. While more advanced pre-processing techniques could certainly have been employed to further improve image quality, however, since it does not adversely affect the demonstration of the proposed methodology's efficacy, these techniques were not investigated any further. It can be seen in Figure 3.12 that more samples are required to contribute to the solution where these artefacts remain, corroborating the expected influence of reflections on the methodology described above.

The ensemble of images was analysed according to 3 different schemes. First, a reference conventional solution was established using conventional WIDIM [72] adopting the default parameters as in Table 3.1. However, the discrete displacement data (following the cross-correlation of interrogation windows) was interpolated pixel-wise using bi-cubic kernels rather than the bi-linear scheme listed in Table 3.2. Subsequently, the presented methodology was used to obtain the ensemble mean at a 95% confidence, of being within 0.1px and 0.05px, using the same parameter configuration as the reference solution. Due to the relatively small number of samples (115) and the unknown standard deviation, (3.1b) was used to obtain the pixel-wise confidence interval after each time step, using the Student's t-distribution to calculate t. Regions where the temporally adaptive method uses all 115 images, e.g. regions where the CI exceeds the threshold, such as in the wake region of Figure 3.10, the solution is numerically equivalent to the reference solution. Outside of these regions, where less than the full 115 images have been considered, the solution is expected to differ from the reference to some

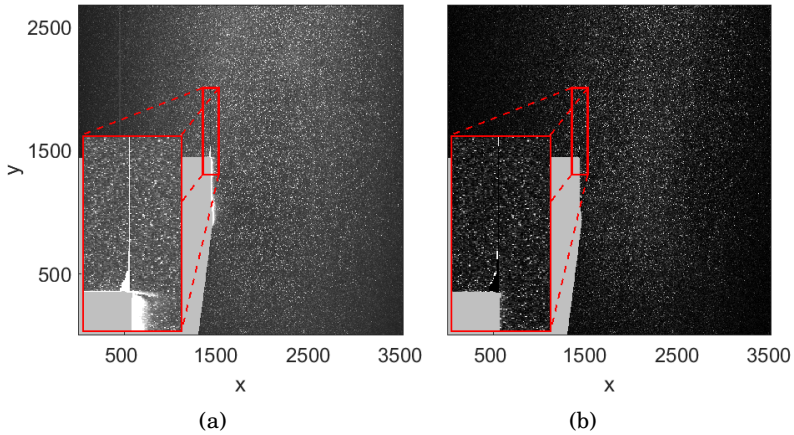


Figure 3.11: Exemplary PIV image from [76]. (a) raw image with zoom inset showing reflection, (b) pre-processed image with inset showing diminished reflection. Contrast enhanced for clarity.

degree, depending on the threshold level. Figure 3.13 shows how much the temporally adaptive solutions differ from the reference WIDIM solution. Plotting the histogram of the magnitude of such differences, i.e. $|U_l - U_w|$, where U_l is the solution from the proposed methodology, and U_w is the reference WIDIM solution, over the domain, Figure 3.14 shows how these follow roughly a normal distribution. In the limit as the number of images increases, this distribution will tend towards a normal distribution with properties equivalent to the imposed conditions, e.g. for 95% confidence of 0.1px, the distribution will have zero mean and a standard deviation of ≈ 0.05, such that 2σ corresponds to 0.1px. The distribution presented within is expected to have a bias towards zero, since the solution in the proposed methodology's solution represents a considerable proportion of the reference solution. Consider a region wherein 100/115 images contribute to the reduced solution. For quasi-steady flow, variations in the final 15 images are unlikely to have a significant impact on the reference solution, yet it is expected that if the reference solution contained many more snapshots, for example in the order of thousands, *from the same underlying statistical distribution*, then the histogram of differences would tend to the normal distribution as described previously.

The number of samples contributing to the ensemble mean, over the entire domain, for each threshold setting is shown in Figure 3.12. In both cases, the freestream requires relatively few samples to reach convergence compared to the wake. This is in line with the expressions for the confidence levels in (3.1b). As the local standard deviation increases, the number of independent samples required to reach a pre-defined confidence level increases quadratically. It can also be seen that the stricter tolerance requires more samples in the fringes of the wake, relative to the more relaxed threshold. Although

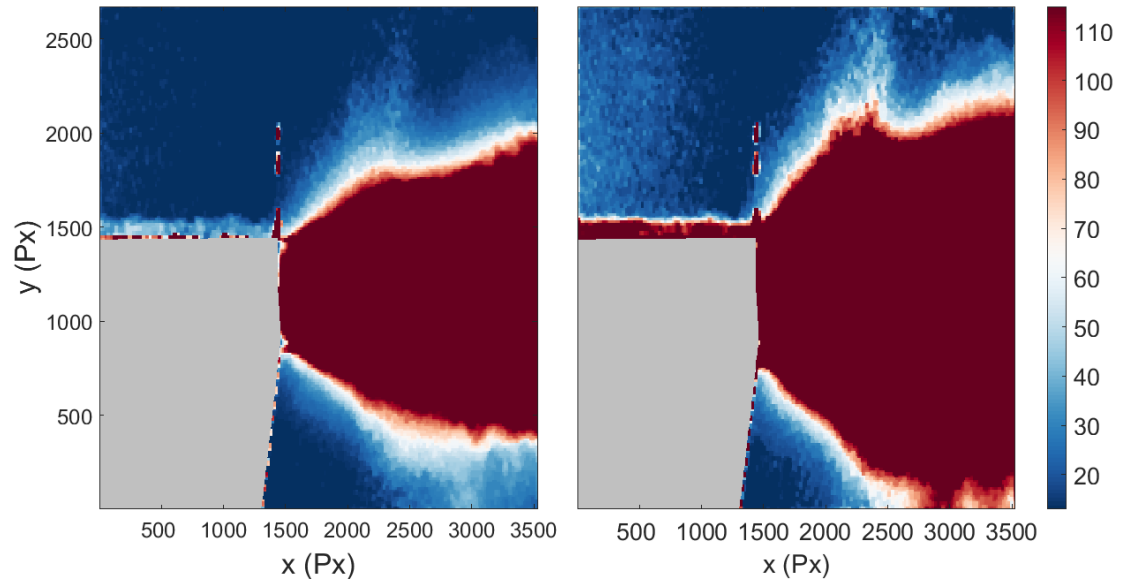


Figure 3.12: Number of contributing samples for a threshold of (a) 0.1px, and (a) 0.05px.

the majority of the image reflection was removed, its influence can still be observed by the increase in the number of contributing samples in Figure 3.12. Comparing the total

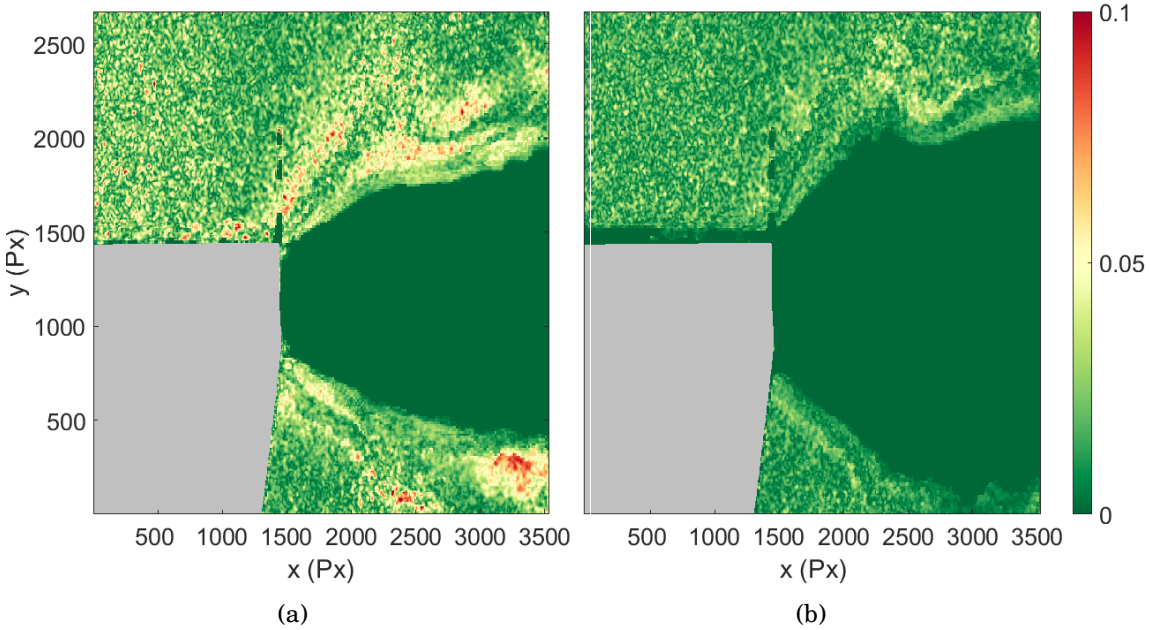


Figure 3.13: Effect of changing confidence intervel threshold on the magnitude difference to a reference full solution, for (a) 0.1px threshold and (b) 0.05px threshold.

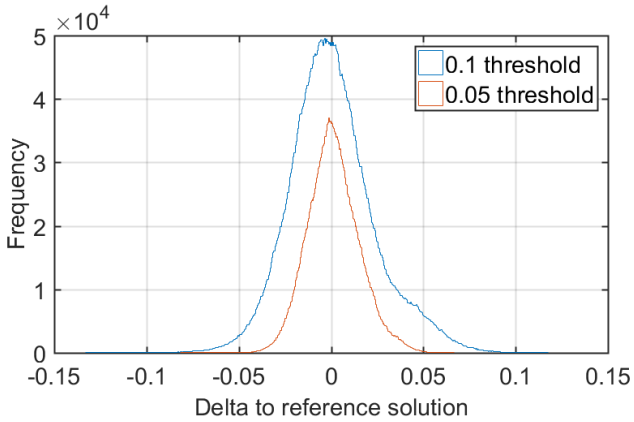


Figure 3.14: Histogram of the delta to the reference WIDIM solution from all pixels where the number of contributing samples is less than that of the reference.

Table 3.2: Comparison of the number of correlations performed for each of the approaches

	Number of correlations
Reference Full Solution	11,867,770 (1.00)
0.05px threshold	8,583,713 (0.72)
0.1px threshold	6,689,502 (0.56)

number of correlations for the entire ensemble, Table 3.2, the strength of the proposed method is made apparent. A reduction of 44% and 28% in the number of correlations is achieved for the 0.1px and 0.05px thresholds, respectively. While overheads exist beyond the correlations, such as interpolation and image deformation, these typically represent a small proportion of the overall computational cost. The number of total correlations is further reduced by using the NMT* approach as described previously, reducing the number of correlations by ≈5% in both cases.

3.5 Summary

A new method has been presented in this chapter which automatically reduces the interrogation domain subject to satisfactory convergence of the flow field statistics. The method has been demonstrated based on the mean flow field, and the concept may be easily extended to arbitrary flow statistics. Convergence is determined by calculating the confidence interval and comparing to a user-defined threshold value, for example, a 95% confidence of 0.1px difference to the expected mean. A ROI is obtained automatically and efficiently based on the flow statistics and used to develop a sampling mask for

each iteration. Generation of the sampling masks considers the imposed interrogation parameters, e.g. WS and overlap ratio, and determines the minimum region of the image which must be analysed to avoid introducing artificial numerical artefacts, known to be associated with image or mask boundaries.

This approach significantly reduces the computational cost required to obtain domain-wide ensemble statistics converged to a pre-defined level by reducing the total number of correlations required. The method was applied to the flow behind a blunt trailing edge, being investigated for its aero-acoustic properties. A numerical assessment revealed that for typical PIV settings, the number of correlations to obtain the ensemble mean was reduced by 44%. The method is expected to be particularly of use when rapid mean solutions are required, i.e. analysing images concurrently with wind tunnel utilisation, or when processing higher-order statistics. Notably, computational savings are made by a reduction in the number of correlations performed, which represents the most time-consuming process in a typical PIV analysis algorithm. Therefore, regardless of the particular hardware configuration, e.g. the use of graphical processing units (GPUs) to accelerate computation of each individual FFT, or the use of large clusters to distribute computations, the presented methodology still offers potential computational savings as fewer correlations must be computed.

The novel method presented hitherto may not align with a traditional adaptive sampling routine, wherein samples are placed according to some objective function, however, considering the spatial extent which must be sampled is adapted according to the data present, and does so automatically, it is argued this method is, in fact, a spatially adaptive analysis algorithm. Accordingly, this method is perhaps the most straightforward of approaches in this category for PIV analysis, and achieves several of the goals set out by this research, notably that methods should be simple, computationally efficient, and enable greater information extraction from PIV images. Despite the fact that two new user-adjustable variables are introduced, i.e. the confidence level and threshold, these may take semi-universal values and provide additional information as to the overall convergence of the solution. These statistics may therefore be extended into a stopping criterion with relative ease, satisfying the original motivation that methods should be autonomous where possible. Although this method is capable of reducing the computational cost, by proxy of the reduced ROI, the sampling conditions, e.g. window size, shape, and location, remain homogeneous and user-defined. Therefore, the following chapters investigate the ability to automatically vary such properties.

CHAPTER 4

ADAPTIVITY; IMPLEMENTATION AND IMPLICATIONS

While Chapter 3 demonstrated that spatially adaptive image analysis could, in some ways, be achieved by simply selectively disabling regions of an image. In doing so, the computational requirements are vastly reduced which may allow for more information extraction if better analysis parameters can now be adopted, making use of the spare resource. Unfortunately, the sampling conditions, i.e. window size and locations, remain undesirably uniform and dependent on user-experience. It would be desirable for a sampling routine to be adaptive, adjusting the parameters according to the local conditions, and indeed there are a couple of attempts in the literature. Adaptive routines, however, require significantly more flexibility than provided by the existing algorithms. Attempts to de-couple even simple aspects of the conventional algorithms quickly cascades into vast algorithmic modifications. The first half of this chapter, therefore, explores the concept of adaptive image analysis within PIV, identifying new questions and challenges that arise as a result of the de-coupling of components from conventional algorithms. The latter half of this chapter introduces novel solutions to two of such challenges - a method to improve the robustness in seeding density estimation and a method to automatically determine the best initial window size.

4.1 Introduction

During Chapter 2 it was argued that conventional methods rely on careful selection of interrogation parameters, which vary on a case-by-case basis, demanding time-

consuming parameter tuning and depending significantly on user experience. It was argued that the conventional method remains inflexible, coupling many key parameters, such as window size (WS) and window overlap ratio (WOR), for which the experimentalist can't reasonably be expected to fully understand the implications of. Moreover, such settings must remain constant over the field of view often leading to sub-optimal sampling conditions, leading to loss of spatial resolution, loss of reliability, increase in computational cost, or all of the above.

Chapter 3 presented a novel method which gradually reduces the analysis region depending on a statistical definition of convergence. In doing so, the computational requirements for a given fidelity may be significantly reduced, allowing more intensive analysis for some finite amount of resources. Although this may allow for better information extraction, in line with the research objectives, image analysis remains fundamentally user-driven, homogeneous, and tightly coupled.

So-called adaptive sampling (AS) approaches aim to combat these issues by spatially varying the sampling parameters and/or sampling locations according to the local conditions. A variety of AS approaches currently exist, yet each approach is associated with tradeoffs and implications which must be considered and understood. For example, simply allowing interrogation windows' size to vary locally challenges the definition of the sampling grid - this must now be set independently, but to what? Taking the example further, should interrogation windows still be halved in size each iteration? Should the sampling grid spacing still be halved each iteration? How should windows be sized and how should signal density be quantified? Since conventional methods (undesirably) couple all parameters to the window size, such questions have not necessarily received much attention and are not trivially resolved. This chapter aims to cover the ways in which various aspects of a PIV image analysis algorithm may be made adaptive. Existing solutions in the literature will be discussed, the benefits, limitations, and implications of such methods considered, and novel solutions proposed where appropriate.

It is worth first clarifying the term adaptive, as it is used frequently in PIV with a variety of meanings. For example, the conventional image analysis process described in the previous chapter is often referred to as adaptive, since the image is deformed according to the output of previous iterations. Hence, by definition, the standard method may be called adaptive. Yet, methods branded as adaptive, for example, Theunissen [59] or Yu [77], implement a fully unstructured approach where the sample *locations* and properties are adapted to the underlying image and/or flow conditions.

Considering the lack of specificity in the term adaptive, the following definitions are

used. In the context of this thesis, *adaptive sampling* is reserved to mean the varying of sampling density over the domain, that is, adjusting the locations of interrogation windows away from a simple cartesian structured grid. On the other hand, the term *adaptive interrogation* is used to refer to other parts of the process which are in some way data-dependent, for example, adaptive window sizing. Despite the fact that, under these classifications, the conventional method, which adapts to the underlying flow by deforming the images, may therefore be classified as adaptive interrogation, in the interest of clarity it will be considered as non-adaptive (conventional) for the remainder of this thesis.

An outline of this Chapter is as follows: First, adaptive interrogation techniques are discussed, with particular attention on adaptive window sizing and shaping, and the implications this has on sampling strategies. Following this AS methods are presented, introducing their motivations, scrutinising the respective components and highlighting the implications such approaches. Following this, two new techniques are introduced to address some of the challenges that have arisen, both focussing on robustness and efficiency. Firstly, an improved method for estimating the seeding density is presented, which allows for more accurate and efficient estimation of the density of particles. Finally, a completely novel approach to obtain the initial correlation window size is presented, which requires no user input and adjusts to both the local image and flow conditions.

4.2 Adaptive Interrogation

4.2.1 Window size

In Section 2.10.3 the requirements for reliable correlation were introduced, as well as the conditions which promote accurate estimation of the particles' displacements. The effective particle density, $N_I F_I F_O$, was introduced which represents the local number of particles, N_I, considering the loss of pairs due to in- and out-of-plane motion, denoted by F_I and F_O, respectively. For reliable cross-correlation, $N_I F_I F_O$, is recommended to be at least eight. Due to the large value of F_I in the first iteration, initial windows are often chosen to be sufficiently large to ensure a reliable, valid, result.

Yet, as is demonstrated throughout Section 2.10.3 (see Figures 2.13, 2.16, 2.18, etc), large windows generally lead to greater uncertainty in the measured displacement. Cross-correlation relies upon the displacement within the window being uniform; when this assumption is sufficiently violated the non-linearity of cross-correlation can result in

significantly modulated, or even out of phase, displacement measurements (See Figure 2.16). In addition, there is a computational penalty as the FFT scales with O(n^2 log n), where n is the size of the window. On the other hand, provided that the largest peak in the correlation map corresponds to the underlying signal, the window is permitted to be arbitrarily small.

In traditional PIV the window size is determined by laborious, user-guided, parameter tuning of the initial and final window sizes, WS_0 and WS_f, respectively, until desirable results are obtained. Not only is this process time-consuming and user-dependent, but it is also unable to be locally tailored according to the varying image or flow conditions. Regions with locally poor seeding or image quality will require larger windows, whereas regions with good seeding, small displacements, and/or moderate displacement gradients will benefit from smaller windows. Thus the user must trade between reliability and accuracy, based on their personal experiences and priorities, leading to potentially significantly differing solutions for the same image set, as demonstrated by the various PIV challenges [78, 2, 79]. It is, therefore, desireable to be able to automatically size windows, locally, based on the observed conditions

Considering only the seeding density, window size can be determined using (4.1), where $WS(x,y)$ is the linear dimension of the window at location (x,y), NI_t is the desired number of particles within the window, and $S_d(x,y)$ is the approximate measured seeding density at the specified location, in particles per pixel. This raises two issues:

- A quantitative measure of seeding density is required,
- Vector spacing determined by WS and WOR is no longer applicable.

$$WS(x,y) = \sqrt{\frac{NI_t}{S_d(x,y)}} \tag{4.1}$$

The former of these issues shall be discussed in detail in Section 4.5. The latter issue is caused by vector spacing being calculated using (4.2) where WOR is the window overlap ratio between adjacent windows, expressed as a percentage. For constant WS, the WOR is commutative; the overlap of some window A on some window B, $WOR(WS_A;WS_B)$, where $WS_A = WS_B$, is the same - $WOR(WS_A;WS_B) = WOR(WS_B;WS_a)$ However, in the case that $WS_A \neq WS_B$, the commutative property is lost, making it impossible to unambiguously define h to achieve some desired WOR Therefore, the vector spacing must now be defined explicitly, or windows must be

positioned using other means, such as adaptive sampling which will be discussed in Chapters 5 and 6.

$$h = WS(1 - WOR) \tag{4.2}$$

In addition to window size variation according to the seeding density, it is possible to also adjust the window size based on the displacement information. This can be particularly beneficial in regions with larger displacement gradients as per Figure 2.13 Theunissen *et al*. [59] implement this by considering the local velocity standard deviation, and adjusting the WS based on seeding according to (4.3), where $C_0 = 0.3$ and K is limited to $-1 < k < +1$ to limit the variation of WS due to signal. The local velocity standard deviation, σ_u, is calculated over a kernel with the same size as the correlation window, and the reference, σ_{ref}, is the median of σ_u.

$$WS^{'} = (1 + C_0 k)WS \quad \text{where} \quad k = \frac{\sigma_{\text{ref}} - \sigma_u}{\sigma_{\text{ref}}} \tag{4.3}$$

While this approach puts in place limitations on the variations of WS, it doesn't consider the *actual* quality of the cross-correlation the reductions in WS may make it unable to achieve a valid correlation. With this in mind, Wieneke and Pfeiffer [80] consider as well the signal to noise ratio (SNR) of the correlation in the calculation of the window size. In their approach, a scaling term, W, is first calculated as per (4.4), where σ_u is as before, and γ_a and γ_b are coefficients to control the influence of each term in the calculation of the scaling term. The authors recommend a value of $\gamma_a = \gamma_b = 1$ in their paper.

$$W = \frac{1}{\text{SNR}^{\gamma_a} \sigma_u^{\gamma_b}} \tag{4.4}$$

Following this, values of W are then mapped to a factor between 0.5 and one, which is then applied to a user-defined reference window size. The incorporation of SNR in this context is logical since the SNR implicitly parameterises a range of effects such as seeding density, image noise, and out-of-plane motion, however, a reference window size must still be defined by the user, along with tuning of the variability factor. While the definition of a reference window size mitigates the challenges with regards to vector spacing, it still undesirably depends on the experience and input of the user.

4.2.2 WS reduction profile

Modern PIV image analysis routines are iterative in nature, allowing for the reduction of window size to improve spatial resolution. This typically takes the form of a gradual reduction of WS over the first few iterations, often just halving each iteration, and then a couple of cleanup iterations, without changing the WS, to converge the solution. By halving the WS each iteration, parameter definition is easier for the user, however, given the freedom to deviate from this strategy, the best rate of WS reduction appears not to have been discussed in the literature, except for a brief mention by DiFlorio [81].

On the one hand, a more gradual reduction is likely to be more robust, since F_I remains greater while the residual displacement is reducing (see Section 2.10.3). Yet this robustness comes at the expense of computation, since more iterations, and hence many more correlations, may be required to reach the desired WS_f. A greater rate of reduction may greatly accelerate the procedure through fewer, smaller, correlations, although this approach would also increase the risk of invalid data which may irrevocably distort the underlying image.

In the paper by DiFlorio, it was noted the results are strongly dependent on the particular WS reduction sequence used; occasionally the non-halving approach would perform significantly better, however, more commonly there would be no noticeable difference. Furthermore, they noted that the magnitude of improvement is dependent on the underlying displacement field.

Fundamentally, assuming that WS_0 appropriately evaluates the local displacement, nothing prohibits the immediate reduction of WS to a size limited purely by seeding, i.e. $N_I F_I F_O$ with $F_I \approx 1$. Unfortunately, it is difficult to validate this assumption during analysis since it is feasible for the correlation response to be completely out of phase with the true displacement. Such behaviour makes it impossible to specify lower bounds on a robust rate of WS reduction. Despite these challenges, criteria that reliably maximises the WS reduction rate, without introducing instabilities, may bring significant computational savings.

4.2.3 Super-Resolution PIV

One common method to extract more information from PIV images is to use a super-resolution technique which aims to combine the strengths of both correlation-based PIV and tracking-based PTV. The concept was originally proposed by Keane *et al*. [82], although has been proposed with varying implementations by, for example, Guezennec

and kiritsis [83], Susset *et al*.[84], and Hart [85]. A hybrid approach allows the search region in the second image to be reduced, mitigating the challenge of PTV to unambiguously pair particle images as described in Section 2.5.1.

While this approach can improve the spatial resolution by increasing the number of vectors towards the number of particles, it remains sensitive to the aforementioned (Section 2.5.1) noise in the sub-pixel fitting process, and to outliers from the initial displacement estimate. PIV is not plagued by such sensitivity to noise in the peak fitting process, since the fit is applied to an average of several particles. Considering this effect, it is noted in the PIV challenges [2, 79] that PTV is unlikely to be superior to PIV unless the seeding density is very low, wherein significantly large windows would be necessitated thus compromising spatial resolution, or where the velocity gradients are extreme. While super-resolution PIV is appealing in theory, it does not, therefore, represent a practical solution under typical experimental conditions.

4.2.4 Window shape

Beyond adapting window size, it is also possible to adapt the window shape to improve the accuracy of cross-correlation in the presence of displacement gradients by reducing the range of displacements present in a window[86, 81, 87, 74, 80]. There are, broadly speaking, three methods of non-isotropic windowing implementation; use of rectangular correlation windows, gaussian weighting of intensity values, and interpolation of image values.

The first of these methods is the simplest and is particularly useful near object or mask interfaces, though, appears not to have been discussed much in the literature[1]. The second, and perhaps most common, method is to apply a Gaussian weighting function to a square correlation window, where the weighting in each direction may be adjusted to obtain the desired window eccentricity[86, 80, 87]. The square correlation window, with scaled pixel intensity values, can then be correlated using a traditional two dimensional FFT approach. Alternatively, the non-uniform window may be obtained via image interpolation in much the same way that image deformation is applied, e.g. using a B-Spline interpolant [88], as proposed by [74]. Although it is argued that this approach reduces the computational cost relative to Gaussian weighting, this remains a significant computational overhead.

While adjusting window shape can be shown to reduce the interrogation error, the

[1] A lack of literature on this topic is also reported in [86]

approach is disadvantaged by two key issues; computational overhead and sensitivity to noise. For example, when applying gaussian filtering, oversized correlation windows must be used, adding significant cost by itself, to avoid artificial truncation. In their paper [86], Scarano notes that an eightfold increase in the computational cost is observed when considering window size implications and the cost of filtering.

Furthermore, calculation of second derivatives from instantaneous data remains noisy, both due to erroneous vectors and the inherent uncertainty fluctuations of instantaneous correlation [74]. Notably, the uncertainty in regions of large spatial gradients, where such approaches are intended to be of primary benefit, is known to be inflated and therefore, instantaneous gradient data, particularly second derivatives, may be unsuitable for window shape adaptation.

Due to the significant computational requirements, and the limitations imposed by measurement uncertainty, window shape adaptation is not considered in the current works. Solution improvements should be first sought from window size and location adaptivity alone, leaving window shape adaptation as a secondary option where critical.

4.2.5 Remarks on adaptive interrogation

Adaptive interrogation is an umbrella term to represent the adaptive aspects of PIV image analysis not relating to the adaptation of sampling locations, which can lead to improvements in either accuracy or robustness, both desirable for improving data yield. Several approaches have been briefly discussed in this section, such as adaptive window sizing and shaping, and controlling the reduction of window size over some adaptive number of iterations.

In conventional methods these aspects are often held fixed or are implicitly determined by other parameter choices, e.g. the number of iterations is normally dictated by the initial and final WS. Accordingly, the optimal choice of such parameters has received little attention in the literature, yet when decoupling in adaptive *sampling* methods, it is important that adaptive interrogation is also configured appropriately. This is particularly challenging due to the amount of coupling currently present. For example, consider adjusting the WS reduction profile, using (4.2) to decide vector locations. In addition to the window size varying, the vector spacing and locations are also influenced, which can both affect the solution in a significant manner thereby conflating multiple causes to a single effect. Therefore, a key driver for this research is to de-couple these parameters, enabling more specific research into each to be conducted.

4.3 Motivation for Adaptive sampling

Chapter 2 demonstrated the rigidity of the existing analysis approach and argued that a more flexible method would be beneficial. In addition to adapting the interrogation parameters, e.g. window size or number of iterations, it is possible to add yet further flexibility by using an unstructured sampling methodology, selecting the best locations based on an estimate of the available information. AS methods generally intend to maximise the amount of information gained per unit of resource, e.g. compute time.

Traditionally, AS methods are particularly useful when the cost per sample is very large; consider a geological survey where each sample requires travel to and from some physical location. In such cases, a full-factorial sampling approach, i.e. sampling on a regular grid, may significantly hinder the attainable spatial resolution due to limited resources. It is therefore desirable to more carefully identify the locations which will yield the most useful information.

In the context of PIV, the cost per sample (correlation window) is relatively cheap, however, a significant number are required and so benefits may still be obtained if the total number can be reduced by some fraction. The benefits are particularly noticeable when the dynamic range of flow scales is large, i.e. there are some very small flow scales, requiring lots of samples per unit space, and some very large flow scales, where fewer samples are required. It is important to consider the total cost to both allocate and sample when comparing conventional and adaptive methods.

Consider the simple example of a sinusoidal signal with linearly increasing frequency, i.e. $f(x) = \sin(x^2)$, sampled at N equispaced locations and reconstructed using cubic spline

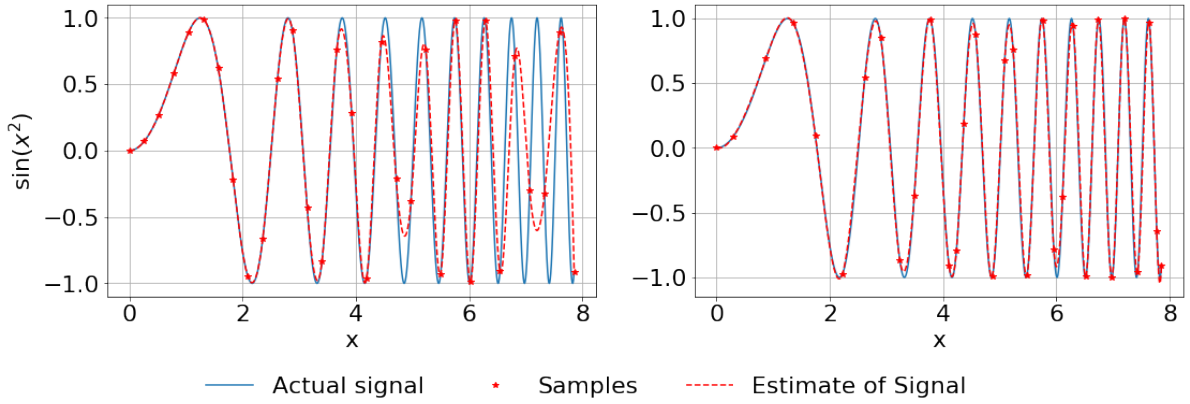


Figure 4.1: A typical illustration of the benefits of adaptive sampling. A sinusoid of increasing frequency is sampled with 30 (left) equispaced samples (right) adaptively placed samples.

interpolation, as shown in Figure 4.1 (left) for $N = 30$ and $0 \leq x \leq 2.5\pi$. The accuracy of the surrogate model becomes poor in the high-frequency range due to insufficient sampling, despite the fact that the sampling rate is still ≈ 1.5 times the Nyquist rate[2].

Figure 4.1 (right) shows the reconstructed signal when samples are placed in an adaptive manner, showing significant improvement. In this example, five equispaced samples yield an initial estimate of the signal, following which the location of maximum error is assigned a sample. In practice, the true error is often unknown and locations of additional samples must be determined from other metrics. In fact, sampling according to the maximum error does not necessarily represent an optimal AS strategy, and may lead to a *greedy algorithm*[3], however, it serves to illustrate the benefit of AS approaches.

Figure 4.2 shows the mean absolute error between the cubic interpolation model and the imposed signal. Interestingly, for $N \leq 20$ the equispaced sampling approach has less error than the adaptive approach, though, this is effectively random error since in neither case is the general structure of the underlying signal well-captured. Beyond this point, though, the error for the adaptive approach is consistently less than the equispaced sampling approach.

Yet, as previously mentioned, the total cost of allocation and sampling must be considered. Figure 4.2 also demonstrates the relative performance if the cost per sample for the adaptive method is 30% or 50% greater than the cost for equispaced sampling. The shifted curve suggests that, depending on the amount of overhead for the adaptive method, sometimes an equispaced sampling approach with more samples may still outperform an inefficient adaptive process. Accordingly, although the efficiency of proposed adaptive methods is paramount, significant improvements in computational performance or accuracy can be achieved through the use of adaptive processes.

4.4 Adaptive Sampling

As previously mentioned, adaptive sampling describes the process of adjusting the smapling locations away from a regular structured grid. The first mention of adaptive sampling in the literature appears to be by Scarano [72], wherein it was called 'adaptive space resolution'. In their method, samples are allocated in an unstructured manner according to the local flow curvature and then recovered onto a structured grid by means

[2]This also serves as a reminder that the Nyquist frequency is the *minimum* sampling rate at which it is *possible* to correctly sample a signal.

[3]A greedy algorithm is one which always selects the best immediate option, without considering the bigger picture. They are, therefore, often not good strategies for finding the globally optimal configuration.

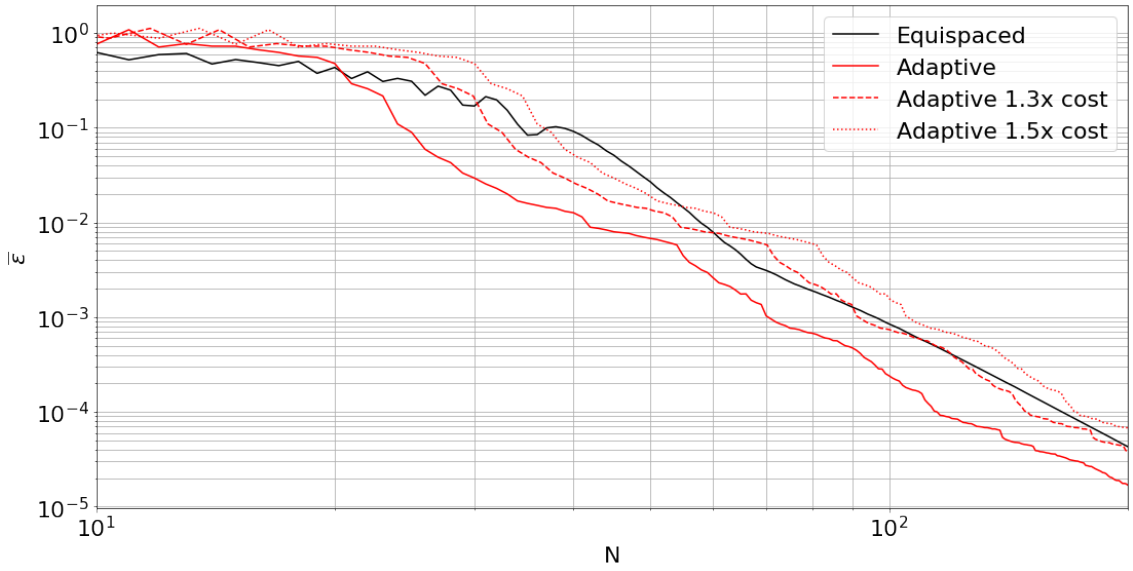


Figure 4.2: Reduction in mean error when sampling a sinusoidal signal with an adaptive methodology rather than equispaced sampling. Additionally, illustrates the importance of minimising compuational cost for adaptive methods, by shifting the adaptive curve by a factor representing the additional cost per sample.

of a least-squares regression[4] and compared to a structured implementation. Linear window size is also adjusted between 8 and 32 pixels according to the local curvature, with smaller windows in regions of higher curvature. The method is shown to improve the solution by increasing the velocity gradient range, allowing shear layers to be better identified.

Developing on this, Theunissen *et al*. [59] present a more detailed methodology wherein the desired sampling density is governed instead by the local standard deviation of displacement and local seeding density. The reasoning for this objective function is that where the local fluctuations in displacement are large, i.e. regions with large displacement gradients, or where more information is available, due to the presence of more particle images, more windows should be attributed. Samples are then allocated according to a two-dimensional pdf transform method [89], which is covered in more detail in Chapter 5. Incorporation of seeding density information is said to significantly improves the robustness of this approach. Improvements in the ability to capture regions of high velocity gradients were shown, specifically an experimental vortex centre, supersonic shock waves, and a boundary layer.

More recently, Yu *et al*. [77] argue that the distribution method used by Theunissen

[4]It is not clear from the text how this is performed.

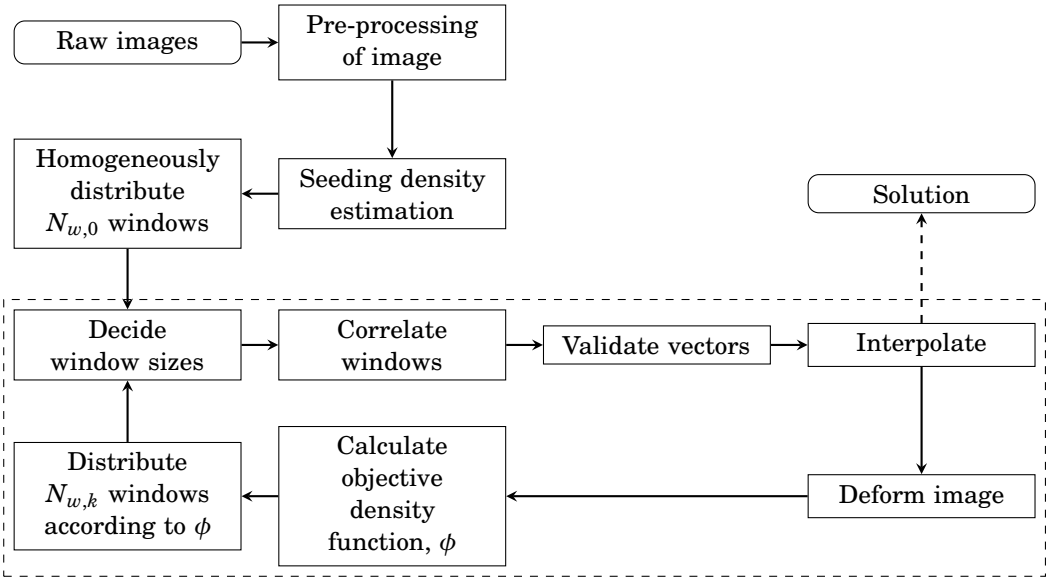


Figure 4.3: Flow chart outlining the main steps in an adaptive sampling algorithm. The dashed box indicates the main loop of the process, where k is the iteration counter and will exit after K iterations.

results in excessive clustering of samples which degrades reconstruction interpolation accuracy. Since mesh smoothing is purely mathematical, and does not consider the original objective function, they argue that a tradeoff between mesh quality and faithfulness to the desired distribution is imposed. A new method is proposed, based on a method originally proposed by [90], within which samples are iteratively moved towards equilibrium by means of a spring-force analogy. In addition, the authors proposed an alternative objective function based on the local curl and velocity gradient magnitude.

The algorithms above can be broadly characterised by the flow chart in Figure 4.3, with the particular implementation varying between the previous references. With the exception of distribution methods, which are covered in significant detail in Chapter 5, the remainder of this section breaks down each component discussing the motivations and challenges of each.

4.4.1 Objective function

Adaptive sampling strategies currently within the PIV literature, notably Theunissen *et al*. [59] and Yu *et al*. [77], construct objective functions which serve to identify the desired sampling densities across the domain to maximise information extraction. This is achieved by balancing flow interest, image quality, and space-filling desire, as well as possibly considering previous sampling efforts. It is equally possible for objective

functions to represent either the desired sampling *density* or *spacing*. In the context of PIV it is argued that objective density functions are more intuitive, as there is a positive correlation between e.g. seeding density or flow gradients and the desired sampling intensity. For the purposes of this thesis, the term 'objective function' will be synonymous with objective density function unless explicitly clarified.

Typical PIV objective functions can be broken down into flow and signal adaptivity. While signal adaptivity can be derived from the seeding density of the image, flow adaptivity is less well defined. Theunissen *et al*. [59], for example, use the local spatial standard deviation, $\tilde{\sigma}$, of displacements, over a kernel of linear size k, as shown for a single displacement component in (4.5).

$$\tilde{\sigma}_u(x,y) = \frac{1}{k^2} \sum_{i=x-k/2}^{x+k/2} \sum_{j=y-k/2}^{y+k/2} (u(i,j) - \overline{u}) \tag{4.5}$$

Taking the euclidean norm of $\tilde{\sigma_u}$ and $\tilde{\sigma_v}$ then yields a scalar function indicating the relative fluctuations over the domain. Applied to the displacement field over a backwards facing step[5], Figure 4.4 shows the resulting local standard deviation, $\tilde{\sigma}$, for a range of kernel sizes, k.

Alternatively, Yu *et al*. [77] use a combination of the displacement curl, i.e. vorticity, and the magnitude of the displacement gradient to identify regions of interest, as shown by (4.6) and (4.7).

$$\lambda_1(m,n) = ||\nabla \times \mathbf{U}(i,j)|| \tag{4.6}$$

$$\lambda_2(m,n) = ||\nabla |\mathbf{U}|(i,j)|| \tag{4.7}$$

The two terms are then normalised by the global standard deviation of each

$$\lambda_{1,2} = \frac{\lambda_{1,2}}{\sigma_{\lambda_{1,2}}} \tag{4.8}$$

Finally, they are once again normalised by the median of each term to obtain an objective density function. Their distribution method (See Section 5.2), however, requires an objective spacing function and so they transform $\lambda_{1,2}$ using (4.9)

$$g(\lambda_k) = \begin{cases} -0.5\lambda_k^{0.88} + 1.5 & 0 < \lambda_k < 1 \\ 0.4\lambda_k - 1.10 + 0.6 & 1 \leq \lambda_k \end{cases} \quad k = 1,2 \tag{4.9}$$

Finally, the scalar objective spacing function is then obtained by taking the minimum of each of these terms. This spacing function can then be used to modify the existing

[5]Analysed using a WIDIM algorithm with WS_0 = 97, WS_f = 23, and WOR=0.5, over three main iterations, where the window size is decreased from the initial size to the final size, and a final iteration to allow the residual to be reduced.

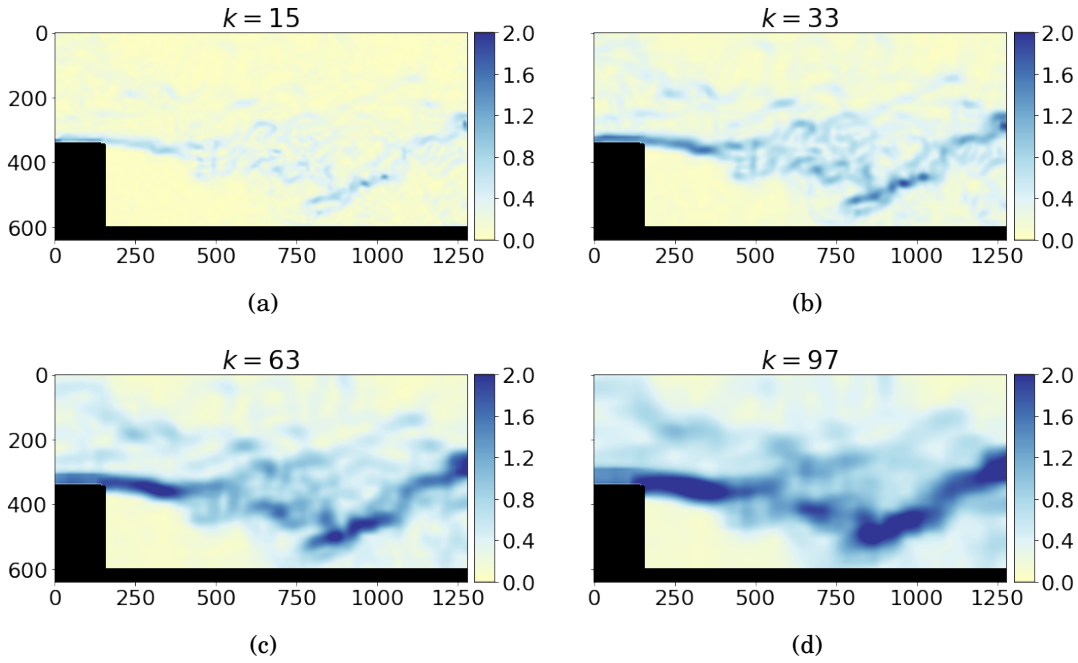


Figure 4.4: The local, spatial, standard deviation for the displacement field of the flow over a backwards facing step. k corresponds to the kernel size over which the standard deviation is calculated, as per (4.5).

sample distribution. The equivalent[6] objective *density* functions, for both the individual components and the scalar result, are shown in Figure 4.5.

These two methods have similar macroscopic behaviour, although the method of Yu *et al*.and the local standard deviation computed with a large kernel size, result in a more homogeneous objective function. A complete study into the advantages and disadvantages of each, as well as considering other options such as curvature or vortex detection criteria, is yet to be conducted. Although the curvature appears to be another attractive choice for objective function, second derivatives obtained from instantaneous displacement fields are known to be noisy [91, 69] and this may result in an erratic and poor objective function. Such a study could be significantly aided by the work of Theunissen and Edwards [35], which allows for the expected response of a correlation window to be obtained for an arbitrary underlying displacement field. Using this methodology, the influence of different objective functions could be more rapidly evaluated as ensemble solutions may be obtained in seconds to minutes, as opposed to several hours required to analyse an entire ensemble of images using a cross-correlation

[6]See Chapter 5, specifically Section 5.3.2, for a discussion on how to convert objective spacing functions into objective density functions.

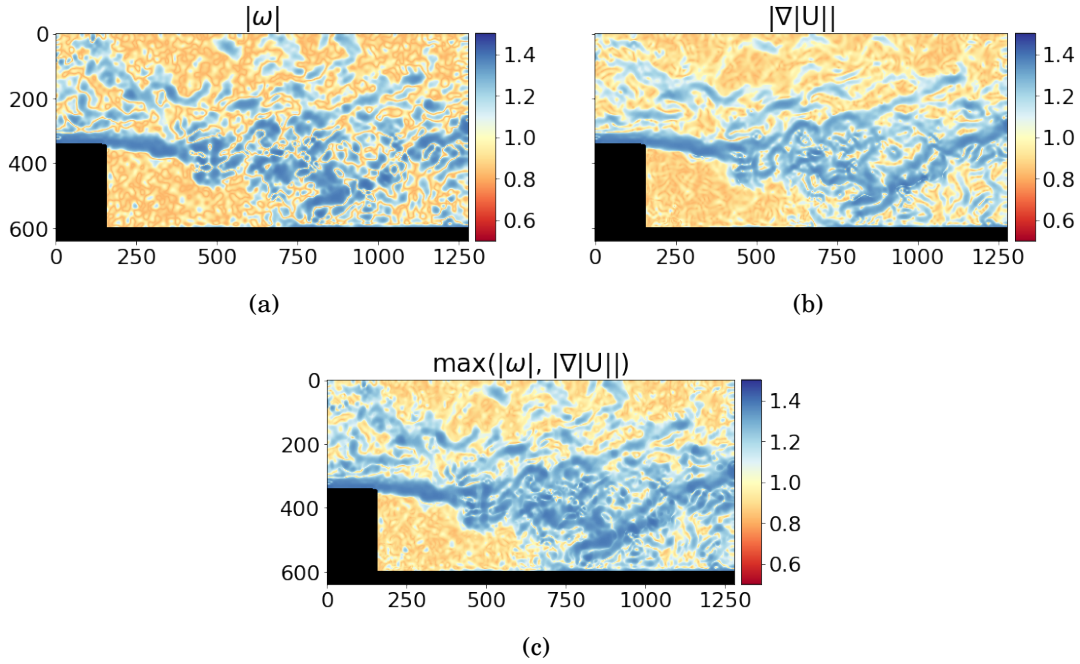


Figure 4.5: Equivalent objective density function using the approach of Yu *et al.*. (a) and (b) show the individual components of vorticity and displacement gradient magnitude, as from (4.6)-(4.9). (c) shows the resulting scalar objective density function.

based method.

While more attention to the choice of objective function is required for adaptive PIV, it is pointless to compare options if the actions which follow, notably sample distribution, aren't faithful to the objective functions. Therefore, before the *logic* can be scrutinised and compared, more flexibility and robustness in the sampling approach is required.

4.4.2 Number of windows

For conventional PIV analysis, the number of correlation windows is an output depending on the size of the image, M x N pixels, and the vector spacing, h_k, as shown in (4.10), where $N_{w,k}$ is the number of windows for iteration k, and h_k is found from (4.11).

$$N_{w,k} \approx \text{floor}\left(\frac{M}{h_k}\right)\text{floor}\left(\frac{N}{h_k}\right) \tag{4.10}$$

$$h_k = (1 - WOR)WS_k \tag{4.11}$$

Typically, the window overlap ratio (WOR) is user-defined and values of 50% and 75% are commonly adopted, with 50% being the most common. The motivation for this choice

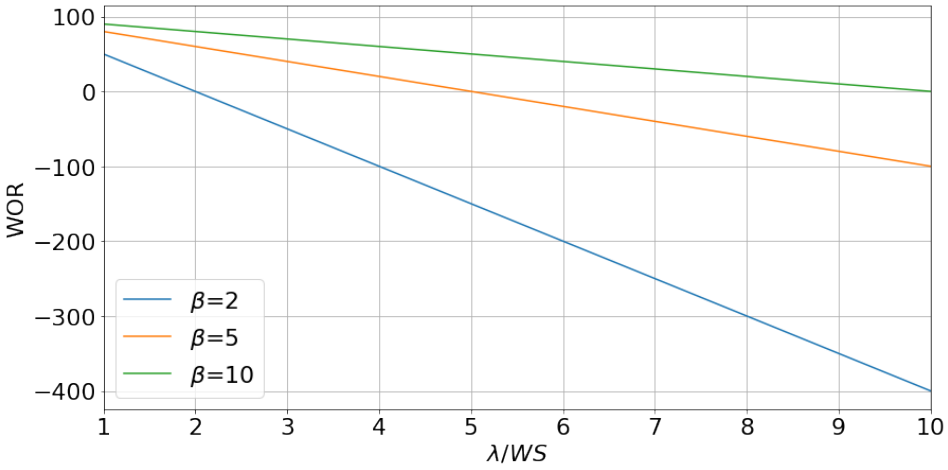


Figure 4.6: Minimum window overlap ratio as a function of wavelength to WS ratio for a range of desired number of samples per wavelength.

is due to the typical use of evenly sized windows[7], e.g. 16, 32, or 64, such that the vector spacing, h_k, as per (4.11), results in integer vector spacing.

A minimum WOR can be easily determined for a given wavelength, λ, window size, and required number of samples per wavelength, β, shown in (4.12).

$$WOR = 1 - \frac{\lambda}{\beta WS} \tag{4.12}$$

The minimum overlap ratios are shown in Figure 4.6, where the blue line, $\beta = 2$ represents the nyquist sampling frequency. Despite being algebraically straightforward, it is not practically simple to evaluate this, since accurate knowledge of the wavelength is required. Therefore, a WOR of ≥50% is often adopted since this ensures that at least the Nyquist sampling theory is satisfied for even the smallest flow scales expected to be encountered.

In adaptive PIV, on the other hand, the user (or algorithm) is free to determine the appropriate number of windows. A straightforward approach is to calculate it using (4.13), where N_p is the number of detected particle images, NI_t is the desired number of particles per correlation window, and WOR is the mean window overlap ratio. This method was used by both Theunissen [59] and Yu [77] and acts as a good benchmark for determining a suitable number of windows. While the WOR no longer has a physical

[7]It is also common to use adjacent odd window sizes, e.g. 15 or 17, 31 or 33, or 63 or 65, such that the centre of the window exists at an integer pixel location, which eases several aspects of implementation. Despite this, for the purpose of vector spacing, it is convenient to treat such odd windows as their even neighbour to obtain integer vector spacing.

interpretation (see Section 4.2.1), it allows the frequency of sampling to be adjusted in a manner that is familiar to most users.

$$N_w = \frac{N_p}{(1-WOR)^2 NI_t} \tag{4.13}$$

In addition to the final number of windows, an appropriate number of windows must be determined for each of the intermediate iterations. In conventional PIV the number of windows increases exponentially as the window size, and hence vector spacing, gets halved in each iteration. Yet, this is predicated on the fact that very large windows, often 128px or larger, are required in the first iteration.

Freedom is therefore given to allow any profile of $N_{w,k}$, and thus the question becomes, 'should the number of windows still follow an exponential profile with respect to the iteration number?'. In the context of adaptive PIV, the number of windows in the first iteration, $N_{w,0}$, should generally be at least as large as its structured counterpart to obtain a good initial estimate of the flow field, in order to guide future sampling efforts.

In the present works, a linear increase in the number of windows is implemented, to ensure the first iteration is sufficiently sampled, and to encourage a more gradual addition of windows, such that divergent behaviour, in the case of outliers distorting the objective function, may be minimised. This conservative choice is made due to the absence of extensive research on this topic. The motivation for restricting window addition rate is to mitigate misguided sampling efforts; if the objective function can be relied upon following the first iteration, then a more drastic increase in N_w might reduce the required number of iterations - which would significantly reduce computational costs. Anecdotal evidence appears to support this, however, no formal investigation has been conducted.

4.4.3 Window distribution

Having determined the required number of windows, the subsequent challenge is to distribute the correlation windows across the domain faithfully to the imposed objective density function. Furthermore, the distribution must be of high quality such that other aspects of the analysis procedure, namely vector interpolation and validation, remain uninhibited. Since many PIV images contain arbitrarily shaped masks, the chosen sample distribution method must also handle these effectively. Chapters 5 and 6 explore this topic in significant detail.

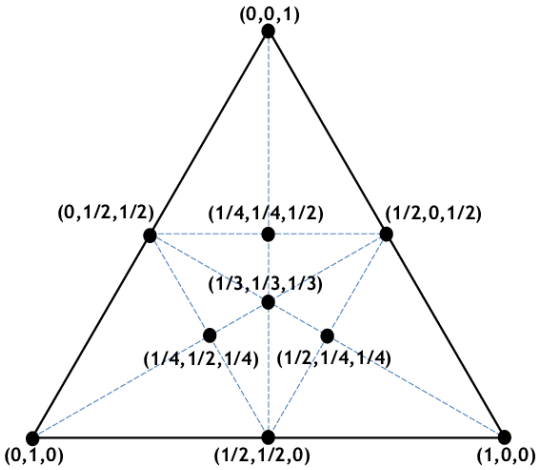


Figure 4.7: Barycentric coordinates for an equilateral triangle [92]

Considering the cost of distribution methods, it is noted that typically the entire sampling distribution is re-generated for each iteration. Since each correlation window must be re-correlated after image deformation, anyway, there is no benefit to preserving a correlation windows' location. However, if samples can be *added* to an existing distribution, of say n samples, while maintaining high distribution quality and reflecting the imposed objective function, rather than fully redistributing, then the computational cost for distributing the first n samples may be saved. For fully unstructured in practice, this overly constrains the addition of new windows, increasing the complexity and restricting the ability to obtain a high-quality distribution. Semi-structured on the other hand, as in Chapter 6, does not suffer from this issue though and can offer further improvements in efficiency.

4.4.4 Interpolation Implications

A somewhat obvious implication of the unstructured data, is that an unstructured interpolation method is required to obtain the solution on a structured grid for post-processing. Although there are a wide range of methods available, the three most common and simple are; linear, cubic, and natural neighbour interpolation.

Unstructured linear interpolation is the simplest of the three, and is synonymous with traditional structured linear interpolation. The interpolant is the weighted sum of the vertex values, where the weighting is obtained from the barycentric coordinates, as shown in Figure 4.7. While simple, the method is only C0 continuous, that is, even the first derivative may not be continuous.

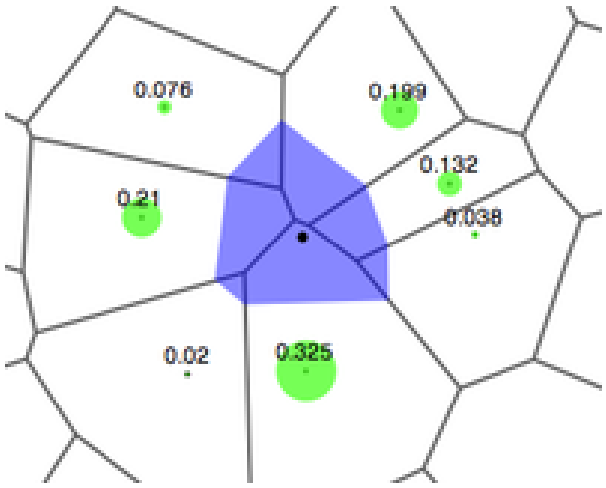


Figure 4.8: Diagram illustrating how neighbour weightings are evaluated. Image created by Markluffel and licensed under CC BY-SA 3.0[8]

The next method is the cubic interpolant, called the Clough-Tocher interpolation scheme, which constructs a cubic polynomial for each triangle in the triangulation. Construction of the cubic polynomial requires calculation of the derivatives in x and y, as well as the drivatives at the edges of the triangle, and hence is more computationally involved than the linear scheme. The resulting interpolation is C1 continuous and is readily available in open source interpolation libraries.

Finally, natural neighbour (NN) interpolation of Sibson [93] offers another C1 continuous interpolant, except at data points where it remains C0. NN interpolation is based on the Voronoi diagram, the dual of a triangulation, and has previously been used in PTV applications [94, 95]. The NN scheme weights data values according to the amount of space that an additional node at the evaluation point would steal from the existing Voronoi diagram, see Figure 4.8.

When applied to PIV, these methods suffer from expensive evaluation when compared to structured equivalents, primarily due to the need to evaluate and then repeatedly query the Triangulation or Voronoi. Based on anecdotal user experience, the cost for unstructured is about an order of magnitude more than a comparable structured interpolant.

[8] https://creativecommons.org/licenses/by-sa/3.0/deed.en

Interpolating Function	$\bar{\epsilon}$
Natural Neighbour	3.6e-4
Linear	3.8e-4
Cubic (Clough-Tocher)	9.8e-5

Table 4.1: Simple comparison of mean interpolation error when sampling Franke's function at 2,500 homogeneously distributed locations.

$$\begin{aligned} f(x,y) = &\frac{3}{4}\exp\left(-\frac{(9x-2)^2}{4}-\frac{(9y-2)^2}{4}\right)+\frac{3}{4}\exp\left(-\frac{(9x+1)^2}{49}-\frac{9y+1}{10}\right)+ \\ &\frac{1}{2}\exp\left(-\frac{(9x-7)^2}{4}-\frac{(9y-3)^2}{4}\right)-\frac{1}{5}\exp\left(-(9x-4)^2-(9y-7)^2\right) \end{aligned} \tag{4.14}$$

To briefly compare performance, Table 4.1 shows the pixel-wise mean absolute error when interpolating 2,500, homogeneuosly distributed, samples of Franke's function, defined in (4.14).

Another option for unstructured interpolation is to use radial basis functions (RBF), which allow for high-order meshless interpolation, and can provide semi-analytical interpolation of derivative information [75]. A radially symmetric function is defined such that for a particular location, $\mathbf{X} = (x, y)$, the interpolated value, s, is the summation of all such defined functions, weighted by some value w_n, as defined by (4.15)

$$s(\mathbf{X}) = \sum_{i=1}^{N} w_i \psi(\|\mathbf{X} - \mathbf{X_i}\|) \tag{4.15}$$

The system can then be represented as a linear set of equations and solved using matrix operations.

The choice of RBF, ψ, determines how the control point's influence varies as the radial distance varies, for which there are several options, a few of such are depicted in Table 4.2.

Despite their benefits, the main drawback for use in PIV is, once again, computational cost due to the number of evaluations required. More critically, the size of the matrix to be inverted can easily exceed available resources; a 2k x 2k image with 16px^2 windows and 50% overlap would require the inversion of a 58Gb matrix. More memory efficient methods can be implemented, though at the expense of computational cost.

To investigate the difference compared to structured interpolation, a structured grid was used to sample Franke's function, (4.14), over a domain of 1000x1000px, and then interpolated onto a pixel mesh using both a structured and unstructured cubic

Radial Basis Function	$\psi(r)$
Gaussian	$e^{-(cr)^2}$
Inverse Multiquadrics	$(r^2+c^2)^{-1/2}$
Thin plate splines	$r^2\log(r)$
Wendland Compact C^4	$(1-r)_+^6(35r^2+18r+3)$
	$(1-r_n)_+ = \begin{cases} 1-r_n & 1-r_n \geq 0 \\ 0 & 1-r_n < 0 \end{cases}$

Table 4.2: A selection of radial basis functions

interpolation. The relative error for each pixel was calculated relative to the known true value, and averaged over the domain. A variety of grid sizes were used and the results are shown in Figure 4.9, with a linear interpolation scheme being shown for comparison.

The uncertainty of unstructured interpolation is typically greater than that of structured interpolation, particularly as the number of samples increases. This is in line with theory, where the convergence rate of structured cubic interpolation is of order two, whereas the convergence rate for unstructured is bounded by order two, and typically slightly less depending on the distribution. By contrast, linear interpolation has convergence rate of order one, as seen in Figure 4.9. The magnitude of the error, however, is significantly below the uncertainty of a single correlation window evaluation, around 0.1-0.01px, hence the interpolation shouldn't, therefore, inhibit the accuracy of the analysis process of-itself.

The Clough-Tocher scheme is, therefore, the chosen scheme due to it's computational performance, C1 continuity, and reduced level of interpolation error compared to linear or NN.

4.4.5 Vector validation implications

As a result of the unstructured data it is no longer possible, at least without modification, to use a method such as the universal outlier detection of Westerweel and Scarano [39], which relies on the data being gridded (see Section 2.11). Two challenges are raised; Firstly, the neighbours need identifying, and secondly, the distance between neighbours must be considered.

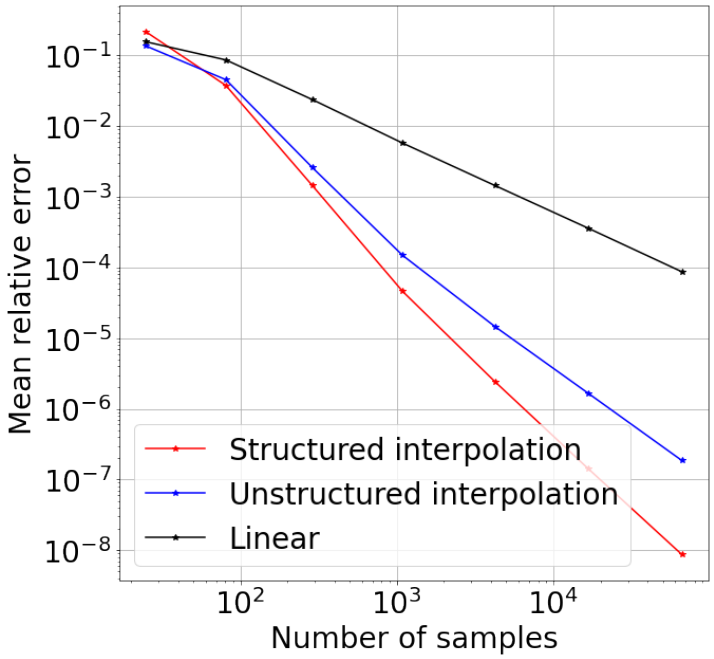


Figure 4.9: Comparison of interpolation error between structured and unstructured (Clough-Tocher) cubic interpolation of the same control points, from a domain of 1000x1000px.

Identification of neighbours for unstructured data is typically obtained through either a triangulation, such as the Delaunay as in [40], or through a kd-tree. While the triangulation approach has the benefit that it may already have been computed from the interpolation step, a kd-tree offers simpler querying to find an arbitrary number of neighbours.

On the second issue, the vectors' locations in space must be considered, as those further away should have less impact on discerning the validity of the selected vector. Accordingly, an inverse distance weighting (IDW) should be applied as per Duncan [40] (See Section 2.11), shown in (2.25).

$$r_0^* = \frac{\left| \frac{U_0}{\text{median}(d_i+\epsilon_a)} - \text{median}\left(\frac{U_i}{d_i+\epsilon_a}\right) \right|}{\text{median}\left[\frac{U_i}{d_i+\epsilon_a} - \text{median}\left(\frac{U_i}{d_i+\epsilon_a}\right) \right] + \epsilon_a} \qquad \text{(2.25 revisited)}$$

Appropriate validation of vectors is a challenging task even for traditional PIV image analysis, wherein unfiltered outliers may cause local image distortion, preventing subsequent iterations from resolving the displacement. This challenge is amplified for adaptive approaches, since uncorrected outliers may misguide adaptive efforts and potentially further disrupt image analysis. Although Duncan *et al*. noted that distance

weighting is only necessary when the flow scales relative to the sample spacing were small, it is not possible to know whether this will be the case without *a priori* information.

Due to the importantace of a reliable vector field for adaptive PIV, the additional cost for the more robust validation method is warranted, and, therefore, IDW validation should be used for unstructued analysis.

4.4.6 Remarks on adaptive sampling

Adaptive sampling can be an effective way to improve the efficacy of each individual sample allowing more useful information to be extracted. Yet, moving to an unstructured algorithm opens up a range of new questions, such as:

- How many windows should be used?
- What size should the windows be?
- How many iterations should there be?
- How many windows should be in the first iteration versus the last?
- How should the windows be distributed?
- How should the vectors be validated?
- How should the vector displacements be interpolated onto the dense predictor?

Furthermore, many of these questions are intricately linked and coupled; for example, the number of windows may depend on the number of iterations, which may itself depend on the size of the correlation windows. While some questions can be resolved relatively easily, i.e. the choice to use IDW validation, others require significantly more investigation, for example, optimal sample distribution.

One particularly prominent question is the question of window size. Adaptive window size strategies as described in Section 4.2.1 depend on an understanding of the available signal in the PIV image, i.e. the seeding density. Furthermore, the seeding density is often used in determining the objective function, as in Section 4.4.1. Therefore, a robust determination of the seeding density is required such that new adaptive methods may be as informed as possible.

4.5 Seeding density

This section presents improvements for the robustness and efficiency of the calculation of local seeding density. Many adaptive methods in PIV adjust according to the local

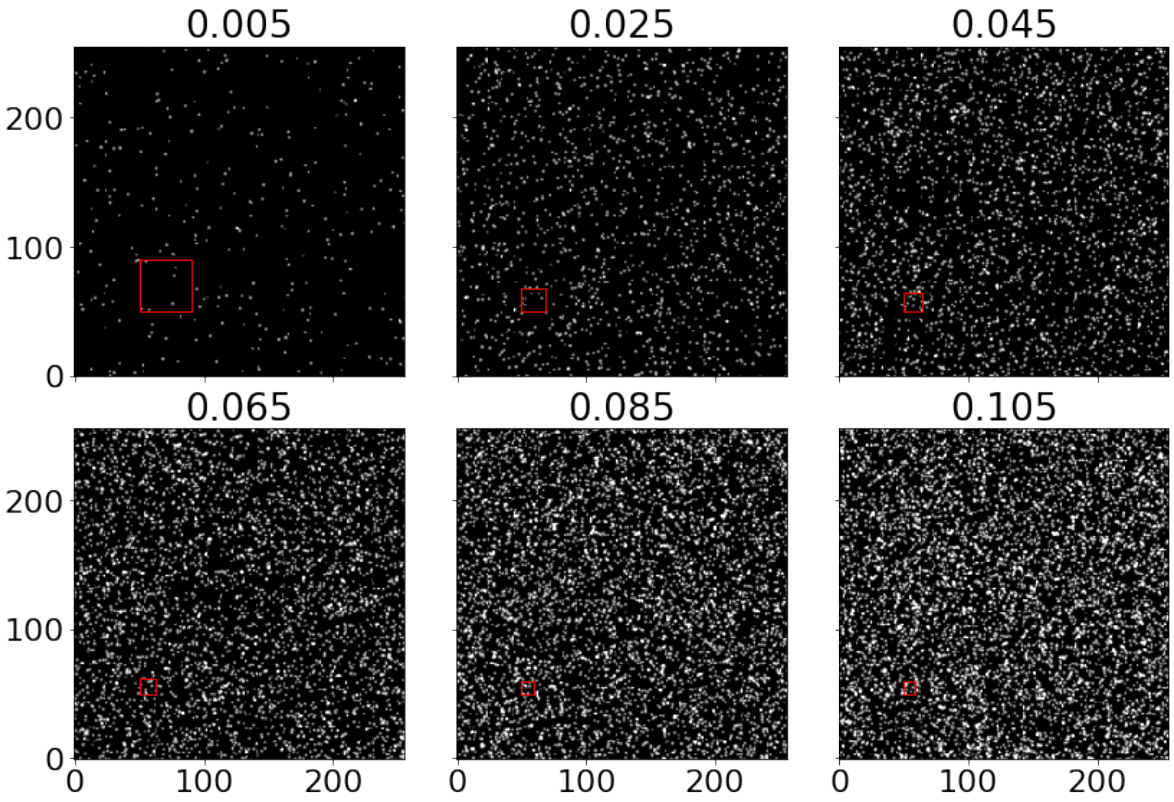


Figure 4.10: Demonstration of what PIV images look like for varying seed densities, stated in particles per pixel. Overlaid is the window size that should, on average, contain NI=8 particle images.

density of particles within the image, sometimes referred to as *signal adaptivity*. Figure 4.10 shows the effect of seeding density on window size for a range of seeding densities. In addition to such window size variation, window locations may also vary according to seeding as per [59].

For both these applications to be effective, a robust estimation of the seeding density is required, which involves two key steps; First, the particles must be identified, and subsequently, the local density of particles must then be evaluated.

4.5.1 Particle detection

Since particle images are typically characterised by bright spots on a dark background, many particle identification methods rely in some way on defining a threshold, where pixels with an intensity value greater than this threshold are determined to be particles. For example, Guezennec and Kiritsis [83] consider the histogram of pixel intensities in order to determine a global intensity threshold. Whereas, Ohmi and Li [96] note

that intensity properties, such as background noise or mean particle intensity, typically vary over the image, and therefore propose a method which locally adjusts the intensity threshold, at the expense of computational cost.

Intensity thresholding approaches may fail to detect dim particles, though, which is undesirable for particle tracking velocimetry (PTV) applications where the particle's brightness is of little concern, provided it can still be tracked between images. To address this, Takehara and Etoh [97] proposed a particle mask correlation approach which searches for any particle-shaped artefact. Despite its ability to handle variable particle intensity, the method remains susceptible to image noise and is computationally intensive.

Although identifying dim particles is useful for PTV, the benefit is less significant for PIV, since a particle's contribution to the correlation map is proportional to its intensity. Therefore, detecting only particles of a certain intensity is synonymous with detecting particles which will positively impact the correlation map.

Developing on the local thresholding of Ohmi and Li, Theunissen [98] introduces a particle detection algorithm wherein local intensity maxima (obtained using a maximum filter) are compared to their surrounding background intensity. Particles are identified where the local maxima are sufficiently larger than their surroundings, as presented in (4.16) where $I_p(i)$ is the intensity of the candidate particle, i. The background intensity is calculated as med(I_{min}), where I_{min} are the local intensity minima, while β is a contrast threshold to determine if the particle is sufficiently bright and is suggested to be 3.5 based on empirical findings. A further dynamic range criteria may be imposed requiring that $I_p(i) > 0.5\max(I_p)$. It should be noted that a dynamic range criterion in this context is effectively a global threshold; in fact, the number of particles detected by this algorithm is quite sensitive to this term.

$$I_p - \text{med}(I_{min}) >= \beta\text{med}(I - \text{med}(I_{min})) \tag{4.16}$$

While this approach robustly identifies particle images, adjusting to local variations of background intensity, it requires many expensive computations of the median operator. A more simple, global thresholding alternative, follows a similar method to those of Guezennec and Kiritsis [83] and Theunissen *et al*. [59]. To address the issue of large proportions of overlapping particles, identified by Ohmi and Li, a maximum filter with a 3x3 stencil is first applied to the image. From the identified maximum intensity pixels, a global threshold is defined using Otsu's threshold method [99], which finds the intensity

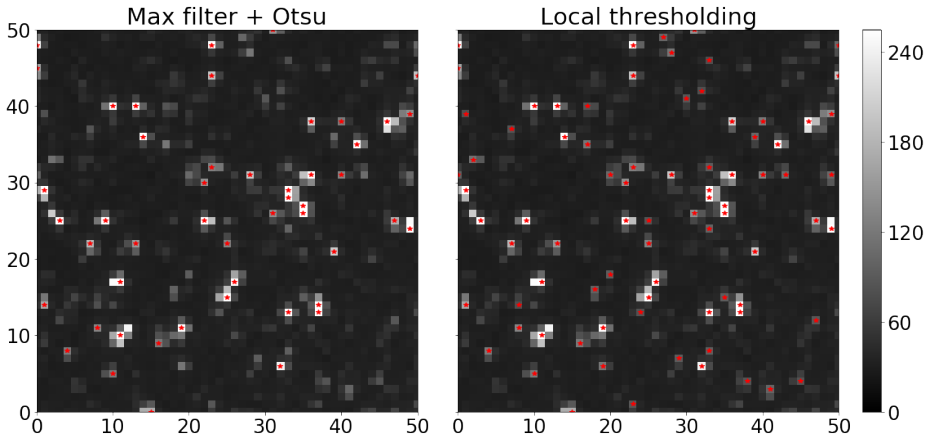


Figure 4.11: A comparison of particle detection performance for a typical region of a PIV image. Red stars indicate detected particles using (a) Global threshold using Otsu's method. (b) Local thresholding as per [98] for a contrast threshold of 3.5 and a dynamic range threshold of 0.5.

threshold which minimises the intra-class variance[9]. Figure 4.11 shows a small region of a typical PIV image, overlaid with particles detected by the global threshold method used herein, and by the local threshold method of Theunissen [98].

The global thresholding method described above shall be used for the remainder of adaptive work in this thesis. It offers an efficient, simple to implement, methodology to detect particles, and performs similarly to local threshold methods in most cases. Despite its performance, it is worth bearing in mind that the ability to correctly identify particle images may be degraded if background noise levels are significantly non-uniform. An interesting line of research would be non-binary weighting of particles for seeding density estimation. It has been noted that for cross-correlation purposes, a dim particle will have less influence on the correlation map than a bright particle, yet it will still have *some* influence. By considering dimmer particles as fractional NI, the proportional impact of seeding on correlation validity may be more accurately understood, allowing for a more universal NI_t to be determined.

[9]This thresholding approach is very similar to that proposed by Guezennec. The choice to use Otsu's method instead derives from the availability of open-source, efficient, implementations of such.

4.5.2 Estimating seeding density

Having located the particle images, the next task is to size correlation windows such that they contain some number, NI, of particles, typically 6-10, as explained in section 2.10.3. A naive approach to this problem is to perform a KNN search, using e.g. a kd-tree, for each window location in order to find the distance to the NIth particle and use this to inform the window size. Such an approach becomes computationally unfeasible for any non-trivial number of particle images.

Before continuing, it is worth mentioning that the seeding density can, in some cases, be obtained directly from the image properties without needing to detect individual particles. For example, Garcia *et al*. use the mean image intensity as a particle density heuristic [100], whereas Westerweel considers the ratio of intensity variance to mean intensity [101]. Although these methods are suitable for a back-of-the-envelope approximation, more information about the experimental setup is required for these values to be useful, and, furthermore, are very susceptible to image artefacts such as glare and reflections. Therefore, these methods are not considered any further.

Instead, the local density of particles is estimated per pixel, which can then be multiplied by NI to determine the window size. The efficient way to approximate the local density is to count, by means of a spatial convolution, the number of particles within a region of known size, in order to compute the local particles per pixel, i.e. the seeding density, S_d. Given a binary image, $p(x,y)$, with ones indicating the locations of particles over the domain, the local particle count, $P(x,y)$, over a region of size m x n can be found using a convolution filter as in (4.17).

$$P(x,y;m,n) = \sum_{i=x-n/2}^{x+n/2} \sum_{j=y-m/2}^{y+m/2} p(i,j) \tag{4.17}$$

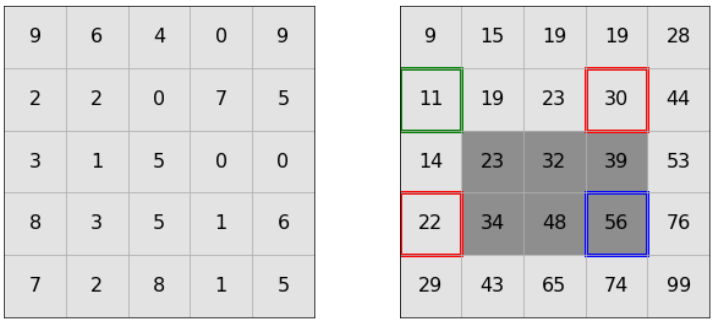
The local seeding density, $S_d(x,y)$, is then approximated using (4.18).

$$S_d(x,y;m,n) = \frac{P(x,y;m,n)}{mn} \tag{4.18}$$

The adaptive WS, which contains NI_t particles, can then be found by (4.1)

$$WS(x,y) = \sqrt{\frac{NI_t}{S_d(x,y)}} \tag{4.1 revisited}$$

If the local estimate of seeding density is accurate, then $NI_{act} = P(x,y;WS,WS) \approx NI_t$. The convolution approach has complexity O($MNmn$), which can be further improved by the use of a summed area table (SAT)



(a) (b)

Figure 4.12: Demonstration of a summed area table. (a) An array of numbers. (b) Summed area table with desired summed region shaded in dark gray. Sum is obtained as (blue - red + green), i.e. $56-30-22+11 = 1+5+0+3+5+1$

An SAT works by first creating the integral image, I_{sat}, see (4.19), where each pixel represents the sum of all values below and to the left, inclusive, which is calculated in O(MN) time.

$$I_{SAT}(x,y) = \sum_{i=0}^{x}\sum_{j=0}^{y} p(i,j) \tag{4.19}$$

The sum of any arbitrary region can then be obtained in O(1) by querying the four corners of the region, as demonstrated in Figure 4.12, and evaluating the expression in (4.20). Thus the seeding density can be evaluated in O(MN), representing a significant computational saving.

$$\begin{aligned} P(x,y;m,n) = \sum_{i=x-n/2}^{x+n/2}\sum_{j=y-m/2}^{y+m/2} p(i,j) = I_{SAT}(x+\frac{n}{2}, y+\frac{m}{2}) \\ - I_{SAT}(x-\frac{n}{2}-1, y+\frac{m}{2}) \\ - I_{SAT}(x+\frac{n}{2}, y-\frac{m}{2}-1) \\ + I_{SAT}(x-\frac{n}{2}-1, y-\frac{m}{2}-1) \end{aligned} \tag{4.20}$$

Although straightforward in theory, it turns out that the choice of convolution kernel is particularly important. When using a universal kernel size, the accuracy of seeding density estimation, i.e. the proportion of times a window sized from the estimate contains the desired number of particles, is dependent on the underlying seeding density. When using a small kernel with low seeding density, there are larger relative fluctuations in the number of particles counted within the kernel. Conversely, when using a large

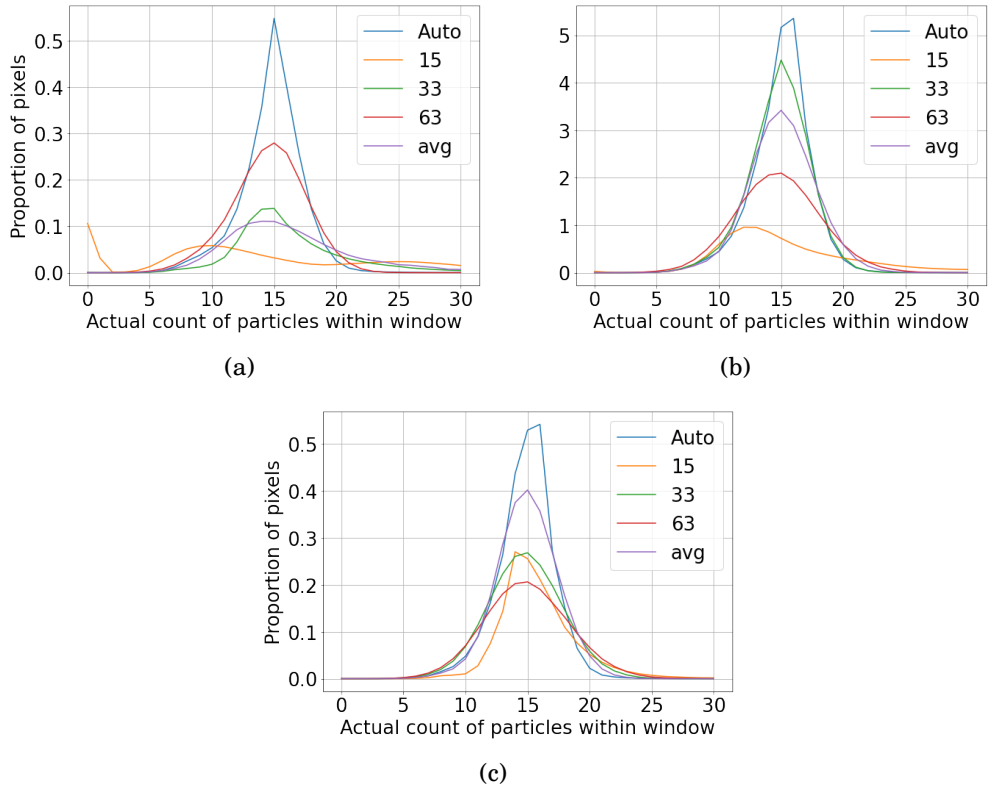


Figure 4.13: The variation in actual particle count found in adaptively sized windows, with a target $NI = 15$, when the seeding density is estimated using different kernel sizes, for (a) low seeding density (0.008ppx), (b) medium seeding density (0.025ppx), and (c) high seeding density (0.06ppx). The kernel sizes detected by the auto method were 51px^2, 31px^2, and 21px^2 respectively.

kernel with high seeding density, excessive spatial averaging takes place reducing the efficacy of adaptive windows. To demonstrate this, Figure 4.13 shows the number of particles actually present within adaptively sized windows, for $NI_t = 15$, where the seeding density has been estimated using kernels of varying dimensions.

One solution, also shown in Figure 4.13, is to average the S_d estimated from several kernel sizes. In this way, the method is less sensitive to large local fluctuations of seeding density. Yet, as can be seen the performance remains mediocre across the different seeding densities tested. Furthermore, this approach requires multiple kernel convolutions adding computational cost. Rather, the kernel size should vary according to the mean seeding density.

The number of particles that appear within any given region will naturally vary due to the random realisations of particles. If the expected number of particle images, denoted by λ, found in a kernel of size $m \cdot n$ is small, i.e. $\lambda = S_d mn <\approx 5-10$, then the

local presence or absence of an additional particle image, by random chance, leads to significant relative error. More critically, if λ is small, then the probability that exactly zero particle images exist within the kernel increases, leading to a zero S_d implying the need for infinitely sized windows[10]. Conversely, if the expected number of particles is large, then the locality of the seeding density estimate is lost and values may be considerably influenced by clusters of particle images near the limits of the kernel.

In order to decide the kernel size, it is helpful to look at the statistics of a Poisson distribution. The kernel size should be calculated using (4.21), which is found by substituting the global mean density, $\overline{S_d}$, into (4.1) in place of $S_d(x,y)$, such that m and n represent the dimensions of the kernel.. For the sake of clarity, NI_t, i.e. the desired number of particles per kernel on average, shall be referred to as P_t, in keeping with previous notation, whereas NI_t will be reserved to mean the target number of particles in the adaptively sized window. In Figure 4.13, the seeding density was estimated using an auto kernel with $P_t = 20$, and its superior performance under a range of underlying seeding density is clear.

$$m = n = \sqrt{\frac{P_t}{\sum p(x,y)/(MN)}} \tag{4.21}$$

Therefore, the question now moves to choice of P_t. Assuming that each pixel within a region of size $m \cdot n$ has an equal probability, S_d, of containing a particle image, the probability that the region contains exactly K particles can be found from the Poisson distribution, shown in (4.22). The Poisson distribution has both mean, μ, and variance, σ^2, equal to λ, hence to minimise the relative error it is desirable for λ to be large, yet not so large that locality is lost. Using (4.22), the reliability of the chosen P_t can be verified within an expected range of local seeding density fluctuations about the global mean.

$$\Pr(K) = \frac{\lambda^K e^{-\lambda}}{K!} \tag{4.22}$$

Most importantly, the number of particles within each kernel should be strictly greater than zero. Assuming the most sparse seeding region has a density of $\eta \cdot \overline{S_d}$, where η is some factor less than unity, then following (4.22), it follows that the probability of there being zero particles can be calculated with

$$\Pr(K = 0) = e^{-\left(\eta \overline{S_d}\right) mn} \tag{4.23}$$

[10]Of course this is impractical and should be caught and accounted for, however, the question of appropriate window size in this situation remains.

Table 4.3: The probability of an automatically sized region containing no particles in the sparsest region of an image, if the actual seeding density in such a region is a factor of η less than the global mean.

	η		
NI_t	$\frac{1}{7}$	$\frac{1}{5}$	$\frac{1}{3}$
10	23.97%	13.53%	3.57%
15	11.73%	4.98%	0.67%
20	5.74%	1.83%	0.13%
25	2.81%	0.67%	0.02%

Using (4.21), the probability of having zero particles depends on the choice of P_t, and on the local density of particles relative to the global mean.

$$\Pr(K=0) = e^{-\left(\eta\overline{S_d}\right)\frac{P_t}{\overline{S_d}}} = e^{-\eta P_t} \tag{4.24}$$

Although η is unknown, it is possible to set a reasonable bound on its value and, subsequently, calculate the maximum probability of having no particles in a region. Table 4.3 shows the probability of zero particles within a kernel calculated using (4.21), for a range of seeding density ratios.

The accuracy of the the seeding density estimate, using the automatic kernel size, may be improved by setting $P_t = NI_t$, as seen in Figure 4.14. However, if $P_t! = NI_t$, then improving performance is found as P_t increases, demonstrated by the consistently solid performance when $P_t = 20$, since the relative error in the estimation of seeding density is reduced. Furthermore, Figure 4.14(a) shows that despite $P_t > NI_t$, adequate performance is still observed.

In light of these findings, the following recommendations are made:

1. $P_t = 20$ represents a robust universal value for the desired particle count in the kernel

2. If large fluctuations of seeding density are present, P_t can be increased to reduce the probability of $P = 0$. Alternatively, the image may be processed in sections, such that fluctuations of seeding density are mitigated at a minor computational expense.

3. If the seeding density is deemed sufficiently uniform, such that the probability of $P = 0$ is minimised, P_t may be reduced in line with NI_t for better locality in seeding density estimate, however, $P_t > 10$ is recommended as a lower limit.

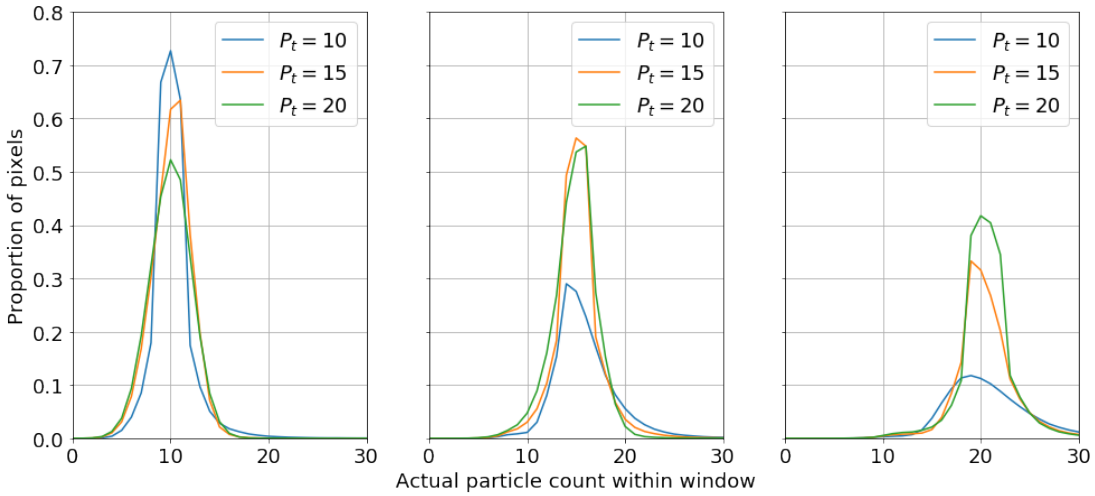


Figure 4.14: The variation in actual particle count found in adaptively sized windows, using various P_t (10, 15, 20), to size windows where NI_t is (a) 10, (b) 15, and (c) 20.

The proposed method for local seeding density estimation has been shown to be robust, accurate, and efficient. A single, automatically selected, kernel size is able to outperform pre-defined kernel sizes, as well as averages of various kernels. Importantly, the automatic selection of the kernel size considers the statistical likelihood of poor S_d estimates in its process. As a result, more faith can be placed in the S_d estimate in the first iteration, enabling another new approach to be presented; automatic initial window sizing.

4.6 Initial Window Size

It was mentioned in Section 4.2.1 that there have been a few attempts to adjust the final window size, based on seeding and flow information. Despite this, these methods don't address the correct, adaptive, choice of the *initial* window size, WS_0. In the methodologies presented in Section 4.2.1, i.e. Theunissen [59] and [80], WS_0 is user-defined and uniform across the image. Therefore, the user's experience (and personal biases of accuracy vs reliability) is still heavily relied upon.

One solution for automatically finding WS_0, considering the lack of displacement information, is to size windows according to the measured seeding density alone, as per (4.1), first introduced by Yu *et al.* [102]. This approach still requires defining a parameter, NI_t, however, this parameter can be set to an almost universal value, typically eight.

While this method is effective under benign conditions, it fails to consider the in- and out-of-plane motion which may cause a correlation to fail. In the paper by Yu *et al*. the test case has very small displacements (1.5px) and hence very little in-plane loss of pairs. In the case of moderate to high seeding, causing small windows, and moderate to large displacements, these correlation windows would likely fail due to lack of correlation. For example, a fairly typical seeding of 0.0355ppp (particles per pixel), with $NI_t = 8$, gives WS_0 = 15px, limiting displacement to approximately four pixels when considering the one-quarter rule of Keane and Adrian [32].

A novel method is presented which tackles this challenge, by iteratively increasing the WS based upon correlation metrics, and is referred to as adaptive initial window sizing (AIW). At each location, the minimum viable WS, WS_0^*, is found based purely on seeding, as in (4.1). The window is then correlated and the following checked:

- SNR $\geq$ SNR$_{thr}$
- $|\Delta\mathbf{X}| \leq WS/4$

Where SNR$_{thr}$ is the imposed SNR threshold and $|\Delta\mathbf{X}|$ is the displacement magnitude. If both are satisfactory, the WS is accepted, otherwise it is incremented by ΔWS and re-correlated and re-tested. To avoid runaway of the method a maximum WS, typically in the region of 80-100px[11], is imposed, though if WS_0^* is already greater than this limit then it is left as is.

To demonstrate the choice of SNR$_{thr}$ and ΔWS, a relatively simple test case of four contra-rotating vortices is investigated first. Although appearing straightforward this flow case contains a good range of flow gradients and hence is an excellent initial case. The displacement field is defined by (4.25), where A_0 is the maximum displacement per component, and L_x and L_y are the dimensions of the domain, in pixels. An exemplery image, with displacement field overlaid, is shown in Figure 4.15(a), where $A_0 = 10$ and $(L_x, L_y) = (1000, 1000)$. The imposed seeding density (S_d) is uniform across the domain and is 0.027ppp, with a mean particle diameter of 3px. Accordingly, the expected window size considering seeding and displacement information is shown in 4.15(b)

$$u(x,y) = A_0 \cos\left(\frac{2\pi x}{L_x} + \frac{\pi}{2}\right)\cos\left(\frac{2\pi y}{L_y}\right) \tag{4.25a}$$

$$v(x,y) = A_0 \sin\left(\frac{2\pi x}{L_x} + \frac{\pi}{2}\right)\sin\left(\frac{2\pi y}{L_y}\right) \tag{4.25b}$$

[11]The maximum WS is chosen by considering the upper range of particle displacement to be 20-25px

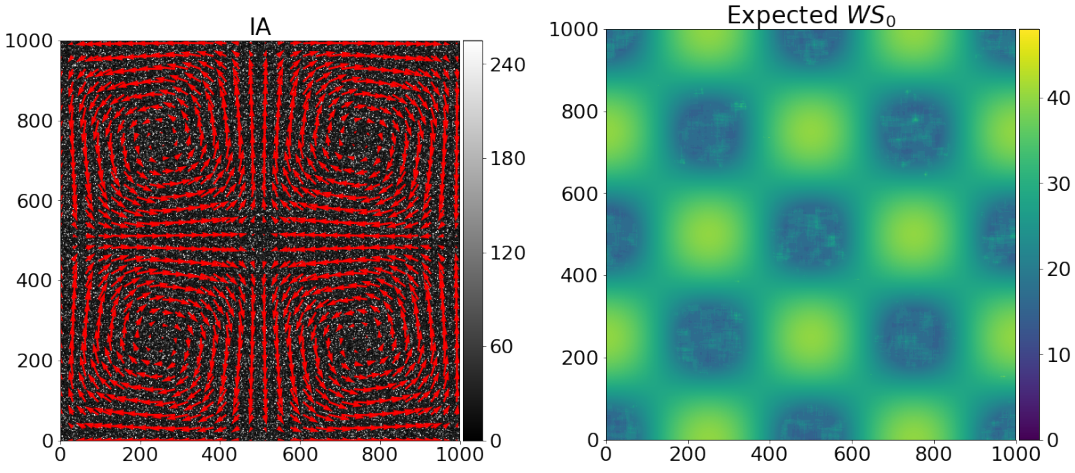


Figure 4.15: (a) A synthetic image with a known displacement field and (b) the expected window size considering both seeding and displacement

The first parameter to consider is the SNR_{thr}. The AIW algorithm was applied using a range of thresholds, with $\Delta WS = 6$px, on a grid with uniform spacing of 32px. The chosen WS_0 at each location is then interpolated onto a pixel grid and compared in Figure 4.16. Note that in addition to adaptively determining a suitable initial WS, this method returns the initial displacement field as would be expected following the first iteration, which is also shown in the figure, before any vector validation.

When SNR_{thr} is small, a larger number of invalid vectors are considered valid, however, as the threshold increases, valid vectors are no longer accepted causing WS to bloat. It is important to note that increasing the WS does not guarantee a larger signal-to-noise ratio (SNR), particularly in regions with moderate to large gradients wherein multiple signal peaks may be observed. Due to the flow gradients, increasing the WS further does not improve the SNR above the required threshold, hence the WS keeps increasing until the imposed maximum WS is reached - at which point the WS and displacement must now be accepted. This is observed in Figures 4.16(c) and 4.16(d) by the brightly coloured spots. The major disadvantage of such a runaway of WS, is the computational cost associated with the unnecessary correlations.

Although it is not simple to *prevent* the unnecessary correlations in these circumstances without *a priori* information, the resulting solution can be improved by reflecting on the history of correlations for each location. If the output and size of each correlation are stored, namely u, v, SNR, and WS, then, upon the maximum WS being reached, the correlation which returned the best SNR may be selected as WS_0. Figure

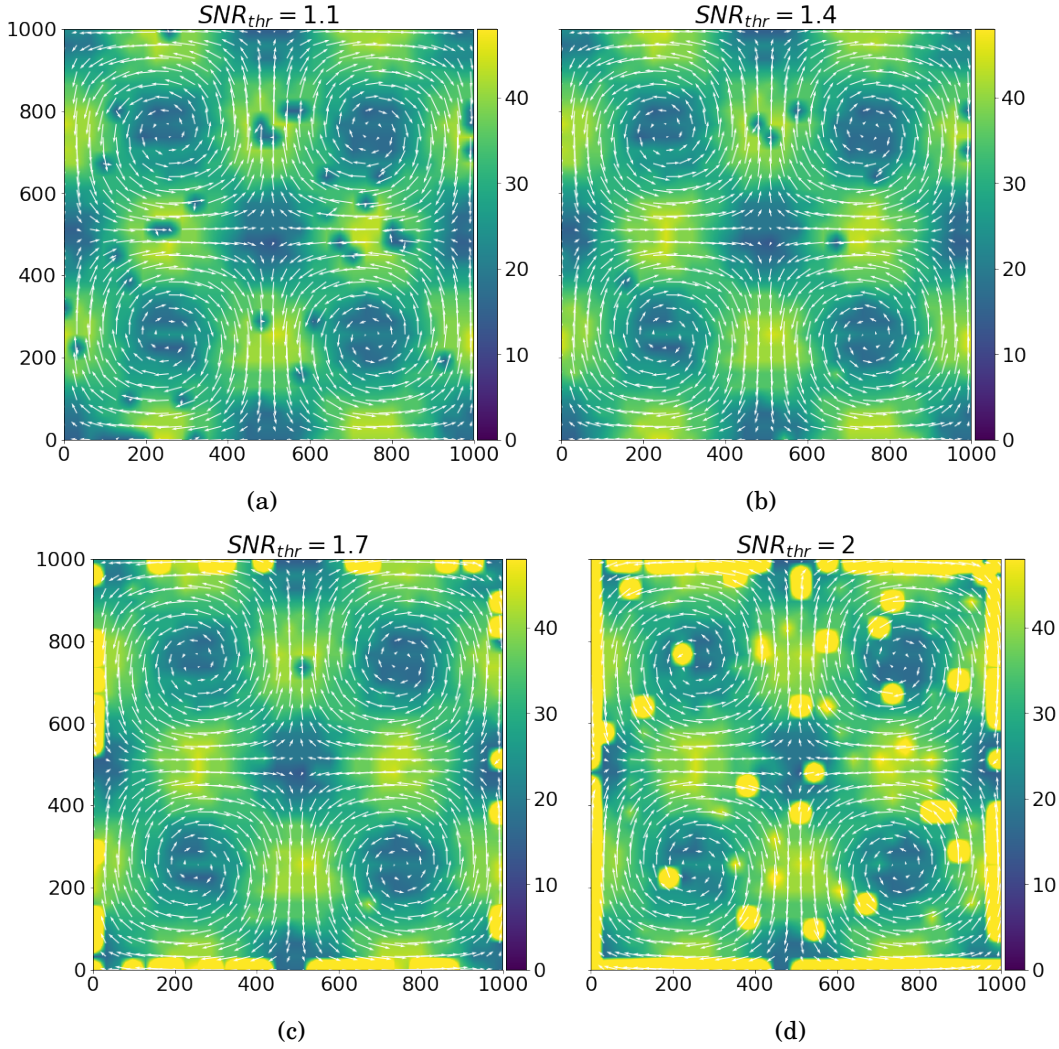


Figure 4.16: Initial window size, WS_0, as selected by the AIW algorithm for various signal-to-noise ratios. Note the different colour scales. The maximum WS_0 in this case was limited to 255px. For illustration purposes, colours are limited to within the range represented by the colorbar.

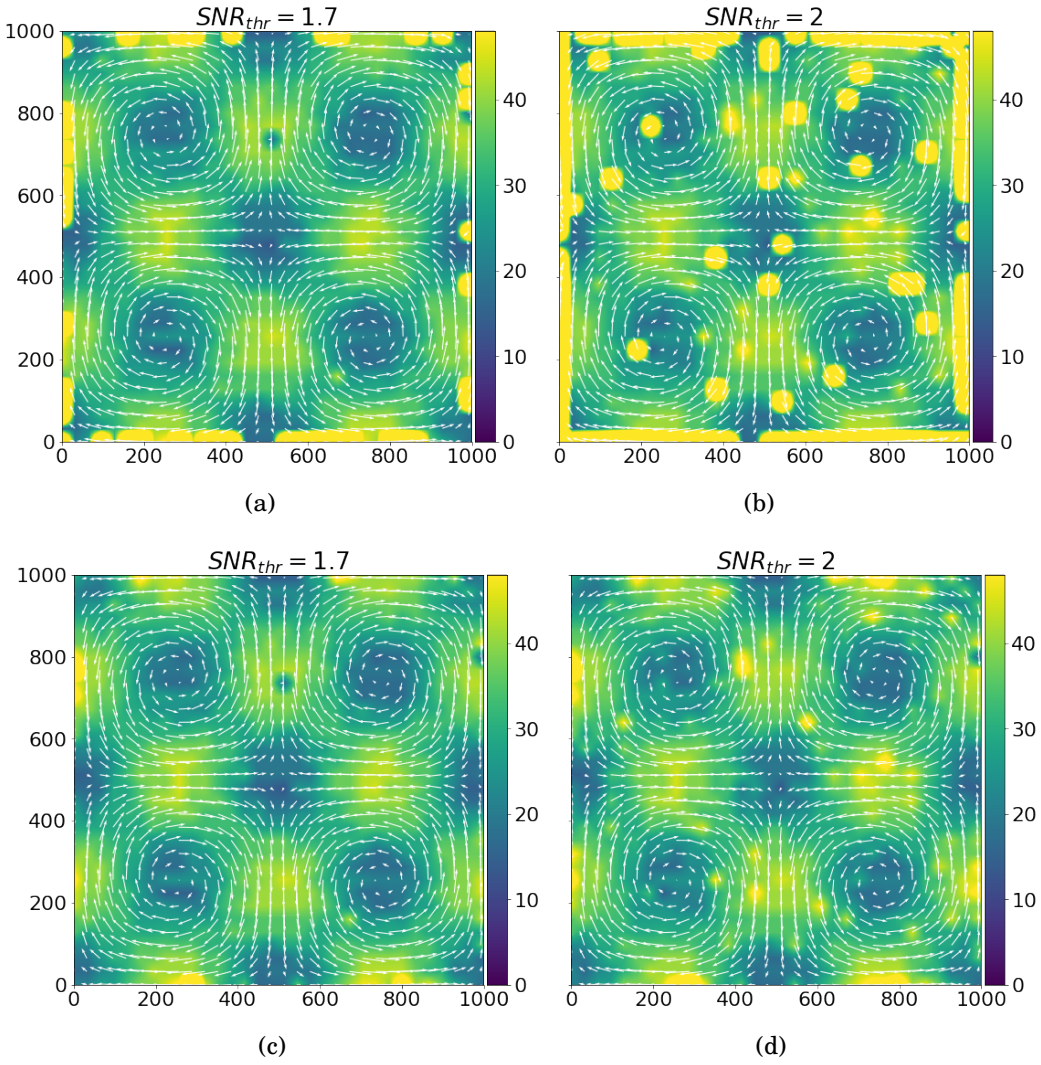
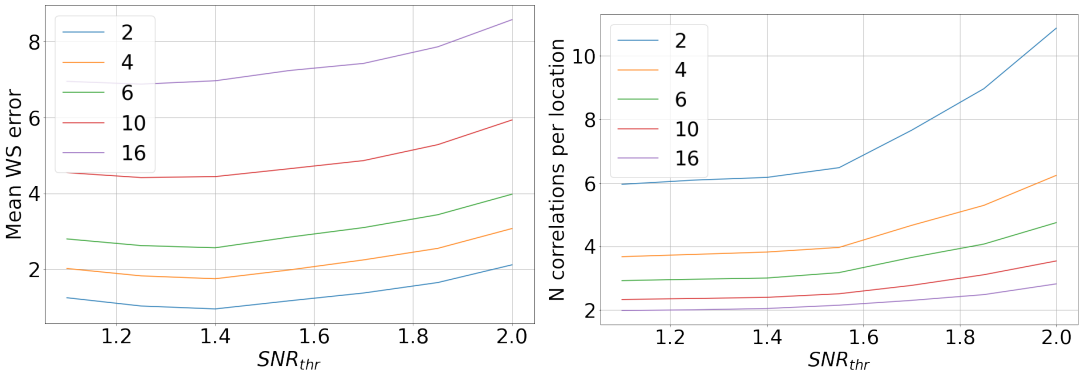


Figure 4.17: Initial window size, WS_0, as selected by the AIW algorithm for various signal-to-noise ratios with and without history preservation. The maximum WS_0 in this case was limited to 255px. For illustration purposes, colours are limited to within the range represented by the colorbar.

4.17 shows how the output of AIW, for the moderately high (1.7) and high (2.0) SNR's, may be significantly improved by adopting this approach. The output for the low (1.1) and moderate (1.4) SNR's is not shown as these are unaffected. History preservation allows the SNR_{thr} to be set relatively conservatively, ensuring a better selection of WS_0, without risking deteriorating the solution if WS_0 becomes excessively large. Accordingly, an SNR_{thr} of 1.4 using history preservation is recommended for most situations.

Another parameter that is important to the efficiency of this method, is the choice of WS step size, ΔWS. The smaller ΔWS is, the closer to the optimum WS_0 is obtained, at



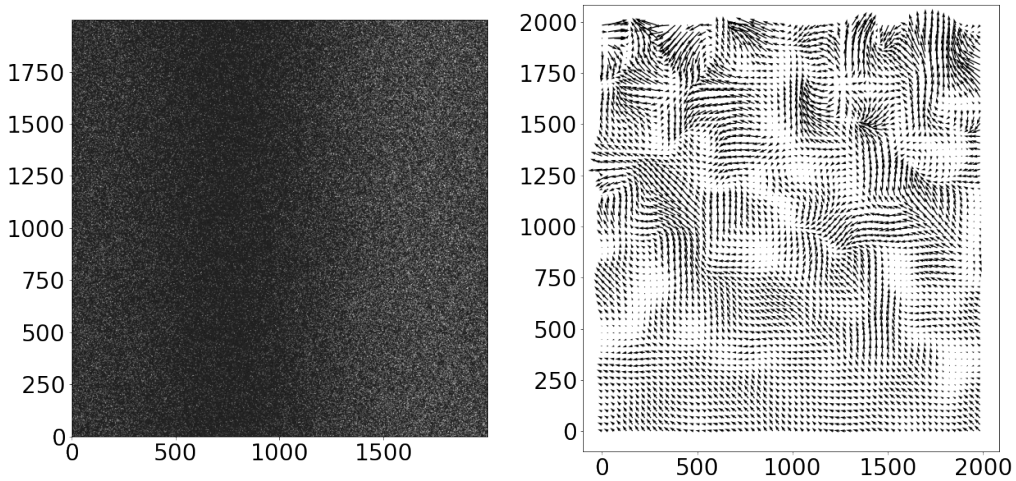
(a) (b)

Figure 4.18: Effect of ΔWS, in px, on the (a) accuracy, defined as the mean absolute difference of the chosen WS_0 to the expected max(WS_0^*, $WS_0^{'}$), and (b) efficiency, defined as the average number of correlations per location, for a range of SNR_{thr}. In both cases, lower is better.

the expense of worse computational performance. Figure 4.18(a) shows how close to the so-called optimum WS the AIW gets, on average, for a range of step sizes. Corroborating expectations, Figure 4.18(b) shows the impact on the number of correlations performed per location.

A reasonable compromise of $\Delta WS = 6$ still requires approximately three correlations per location to get to the final value. While this may seem significant, the size of the correlation window must be considered before assessing whether this approach is more or less computationally intensive, considering the cost associated with correlation to be O(N^2 log N), where N is the size of the correlation window. Therefore, multiple smaller windows may evaluate quicker than a single larger window. This effect is further compounded by successive iterations - a smaller WS_0 leads to smaller WS in later iterations. Furthermore, this method can be run without user input, saving the time required to tediously tune the WS manually.

To demonstrate the efficacy of this method, it is applied to a more complex synthetic flow case and compared with regular structured analysis for both run time and accuracy. The flow consists of gaussian smoothed random noise, with displacement magnitude scaled along the vertical axis, with a maximum component of 17px, of the image to represent a range of displacements and gradients as would be encountered in practice. Smoothing was performed by two sequential passes, first with a kernel with standard deviation of 16, and subsequently a kernel of 51. The seeding density varies across the horizontal axis of the image, also to represent practical images, with a minimum



(a) (b)

Figure 4.19: (a) Image with seeding density varying horizontally with minimum 0.007ppp and maximum 0.05ppp. Contrast enhanced for clarity. (b) Displacement field obtained by smoothing random noise, with vertically linearly varying magnitude, with a gaussian kernel with a maximum displacement 17px.

of 0.007ppp and maximum of 0.05ppp. An exemplary image and displacement field are shown in Figure 4.19.

The image was analysed using the AIW algorithm for single- and double-pass, where the self-selected window size was kept constant over each iteration to demonstrate its robustness. To compare, a structured analysis was also performed for single- and double-pass with initial window sizes 97, 63, and 33. In the second pass the window size was halved as in a traditional structured approach. The vector spacing for all structured analysis was kept the same to allow for a comparable juxtaposition, and set to 24px for the first pass and 12px for the second. This spacing corresponds to an overlap of 75% for the 97px case, 40% for the 63px case, and 28% for the 33px window case.

The accuracy of the methods was quantified by calculating the pixelwise mean absolute error; calculating pixelwise allows for more representative comparison between the two different sample spacings. The runtime of each of the analyses was recorded by averaging over seven repeats for each case, and normalised by the runtime for the single pass AIW. The results are shown in Table 4.4.

There are several interesting conclusions to take from Table 4.4, firstly, that the error when analysing using the automatically determined window size was lowest for both single- and multi-pass. Most importantly the accuracy of single-pass AIW is almost the same as multi-pass for $WS_0 = 97$, despite no image deformation and subsequent

Table 4.4: Table comparing run time, normalised by single-pass AIW, and accuracy of the AIW method against a structured analysis with different initial window sizes.

	Run time		Mean error (px/px)	
	Single pass	Two pass	Single pass	Two pass
AIW	1.00	2.41	0.23	0.14
WS_0 = 97	1.16	2.69	0.68	0.21
WS_0 = 63	1.05	2.48	0.35	0.15
WS_0 = 33	0.49	1.34	0.25	0.18

refinement. While the table shows that by selecting a better WS_0 the user may improve the solution, this is not straightforward in practice where the actual error is not known, and hence difficult to optimise. Furthermore, this trial-and-error based parameter tuning takes up valuable time from the user.

Comparing run-time, the benefit of small windows is highlighted by the performance of $WS_0 = 33$ relative to the other configurations. While this approach is significantly quicker and provides similar accuracy, at least for single-pass, there are three potential issues. Firstly, there are regions of the image where an initial WS of 33px does not satisfy the one-quarter rule and where the seeding density requires larger windows. Although this does not represent an issue of itself, these are both known to reduce valid detection rates considerably placing more dependence on vector validation routines and being more likely to fail in the presence of image noise or artifacts. Secondly, the final window size, 17px in this case, is too small and has resulted in sub-par second pass performance. This can be adjusted by changing the WS reduction profile, though this brings with it its own challenges as per Section 4.2.2 and adds another parameter to be optimised. Finally, it is worth noting that the WOR plays a significant role in the number of windows and therefore the computational requirements. Although in this case a 33px WS_0 with fairly low WOR performs favourably, this would require also optimising for WOR, adding yet more constraints to the optimisation process.

Although the AIW routine uses more correctly sized windows, and thus may benefit from the computational savings associated with smaller windows, it must spend time up-front to detect particles and determine the seeding density. A suggestion to improve the performance of this method, when applied to an ensemble, is to preserve the seeding density over the domain between images. This is likely to remain reasonably constant over the ensemble, and each correlation must still pass the SNR_{thr} to be accepted and so wont degrade the solution.

To conclude this section, a method has been presented which allows the initial window size in a PIV image analysis algorithm to be automatically determined. The algorithm requires an estimate of the image seeding density, from which a minimum window size, WS_0^*, is calculated and used to interrogate the image. The resulting correlation map is assessed based on correlation metrics; if unsatisfactory, the correlation window size is increased and the correlation repeated, otherwise the result of the correlation is accepted. Key parameters involved in the algorithm, SNR_{thr} and ΔWS, have been investigated to find suitable values, for which 1.4-1.7 and 6-10, respectively, are recommended. Although these still represent parameters which may be tuned by the user and thus be dependent on the user's experience, they are more universal than WS, and do not need adjusting in the majority of cases.

4.7 Summary

In Chapter 2, the rigidity of conventional PIV algorithms was identified as a limiting factor for incorporating adaptivity. This chapter has explored how such adaptivity would be included into an algorithm, the implications that this would have, and the necessary amendments that would be required. Due to the significant coupling of parameters in the conventional approach, such amendments are vast and complex to solve. This chapter attempts to clarify the situation, somewhat, has proposed a classification of adaptivity, and has presented two novel techniques to aid the automatic, efficient, and robust adaptive analysis of PIV images.

Adaptivity has been classified into two categories: adaptive interrogation and adaptive sampling. The former refers to adjusting, for example, window size and shape, number of iterations, or the profile with which window size should be reduced. Due to coupling, even a simple desire, on the surface, to locally adjust window size necessitates changes to the definition of vector spacing, and poses subsequent questions such as what *should* the vector spacing be and how *should* the vector spacing change over the analysis process. These questions were breifly explored, though remain as unanswered questions.

The latter category, adaptive sampling, refers to adjusting the sampling locations away from a structured grid, to facilitate more efficient sampling rates. Yet to achieve this, there are many significant implications for many components of the algorithm, which have been explored in this chapter. For example, an objective function is required to govern the analysis process, but how *should* a particular flow field be sampled? Furthermore, given some abitrary objective function, the correlation windows now

require distributing over the domain, yet the topic of how to achieve this in an efficient and faithful manner is large, and will, in fact, be the focus of the next chapter.

Still further, the unstructured samples now require an unstructured interpolation scheme, since a dense predictor is still required for the deformation of images. In Section 2.12.4 various methods were presented, each with their strengths and weaknesses. For the time being, an unstructured Clough-Tocher cubic interpolation scheme appears to be a prudent choice. One key limitation, however, is that such a scheme is only C1 continuous, and so second derivatives may be discontinuous. Schemes that produce smoother interpolations are possible, but the computational penalty grows sharply, possibly negating the attraction of adaptive schemes.

Yet another implication is that of vector validation. Vector validation methods filter erroneous data from the displacement field by comparing to the local neighbourhood. In this chapter it was discussed, and will be elaborated on more in the next chapter, that inhomogenous spacing between local vectors can significantly hinder these routines, placing more demands on the quality of the aforementioned distribution techniques. In the context of adaptive sampling, vector validation techniques are especially crucial, as failure to identify erroneous data may corrupt the adaptive process, acting as red-herrings incorrectly attracting more resource.

In addition to exploring the implications of adaptive algorithms, two novel adaptive interrogation procedures have also been introduced. Firstly, an improved method for seeding density estimation has been presented. This method automatically determines the most suitable kernel size, over which detected particles should be counted in order to approximate the local seeding density. The implementation has been considerably improved by the novel use of a summed area table which enables a reduction in cost from O(k^2MN), where k is the kernel width and $M \times N$ are the image dimensions, to O(MN).

Secondly, a completely new approach has been presented to adaptively determine a robust initial window size, without user-input or *a priori* displacement information. The introduction of such an initial window sizing algorithm is a substantial step towards a fully autonomous analysis routine. Moreover, this method typically results in the smallest viable correlation windows, and as a result the computational cost of cross-correlation is greatly reduced.

To achieve the research objective of maximising (efficient and autonomous) extration of information from PIV iamges, it is clear that more flexibility in the sampling process is required. Such flexibility, however, requires vast changes to the algorithmic procedure. This chapter has made progress towards the research objective, in two ways. Firstly,

more understanding of the implications of such changes is now present, and secondly, two new techniques which adapt to local conditions and reduce user input have been proposed.

CHAPTER 5

SPATIAL DISTRIBUTION OF SAMPLES

Conventional PIV image analysis routines use homogeneous sampling conditions, derived from a relatively small number of user-defined parameters, such as window size (WS), window overlap ratio (WOR), or number of iterations, K. Chapter 3 showed that it is possible to greatly accelerate the analysis process, in line with the objectives of this research, by utilising sampling masks, in effect causing a degree of spatial adaptivity, albeit still utilising conventional, homogeneous, sampling parameters. Addressing this, Chapter 4 explored adaptive methods in the context of PIV, defining the classification of 'adaptive interrogation', wherein sampling parameters such as WS or K are adjusted, and 'adaptive sampling', where the locations of samples are adjusted. Furthermore, Chapter 4 introduced two new 'adaptive interrogation' methods, improving the accuracy of seeding density estimation and enabling the robust, automatic, determination of initial window size, both methods making progress towards the overall goal of an efficient, automatic, and robust PIV image analysis algorithm. The concept of adaptive sampling was then introduced, demonstrating that although allowing for favourable sampling conditions, a significant number of modifications are required to facilitate such approaches, which must remain computationally efficient otherwise they may lose their perceived benefit. Onesuch modification, at the heart of adaptive sampling methods is the need to distribute a certain budget of correlation windows, according to some arbitrary objective function, without inhibiting other components of the analysis algorithm, such as interpolation or vector validation. Efficient, robust, fully-unstructured sample distribution shall therefore be the focus of this chapter.

5.1 Introduction

To enable an effective adaptive sampling (AS) routine, a robust sample distribution methodology is essential, which must remain computationally efficient when compared to structured implementations. Of course, there is more benefit to an adaptive sampling approach than time alone, such as local optimality of sampling frequency. Yet if similar results are attainable using a structured approach, with comparable total resource usage, then the attractiveness of the adaptive approach is diminished

Adaptive sampling strategies currently within the PIV literature, notably Theunissen *et al*. [59] and Yu *et al*. [77], construct an objective function which identifies regions of available information within the domain (Section 4.4.1). It is essential that a distribution method is capable of faithfully reflecting the intentions of the governing objective function, while preserving distribution quality, while handling arbitrary mask geometries typically encountered in PIV experiments [20, 23], and being computationally efficient. Theunissen *et al*.[59] originally allocated sampling locations by converting the objective function into a probability density function (PDF) which could be sampled according to the inverse of the associated cumulative density function. This approach required additional mesh smoothing operations to yield distributions of satisfactory quality, hence Yu *et al*.[77] considered the so-called spring force distribution (SFD), which offers higher quality distributions at the expense of robustness and increased computational effort.

Distribution quality has a strong impact on PIV image analysis routines, specifically the vector interpolation and validation stages. In the context of vector interpolation, even spacing of vectors is generally understood to lead to better interpolation and fewer overfitting problems. Vector validation also benefits from evenly spaced vectors as it leads to neighbouring samples that are more independent, and therefore better able to highlight outliers (See Section 2.11). A general desire, therefore, is for samples to be as quasi-equispaced as possible, wherein the variation of sampling densities should be gradual to promote quality interpolation and robust vector validation.

Despite attempts to make the SFD method more robust, which are discussed in Section 5.3, the SFD will be shown to remain prohibitively expensive for adaptive PIV. Nonetheless, the quality of the distributions serve as a good benchmark for alternative methods. Accordingly, a novel method is proposed which strikes a balance between computational cost and distribution quality.

In the following section, a more detailed discussion of existing methods is presented, highlighting the challenges faced by sample distribution in the context of PIV. Due to

a lack of implementation details in the articles of Yu et. al., a significant amount of work was required to improve the robustness of their presented distribution method, which is covered first. Whereas the distribution method of Theunissen produces samples according to a density function, i.e. greater input values correspond to more samples and smaller intersample spacing, the method presented by Yu *et al*. distributes according to a distance function, i.e. greater input values correspond to larger intersample spacing and fewer samples. To allow fair comparison of the distribution methods, and to further improve the generality of the SFD method proposed by Yu *et al*., a more general transformation function between distance- and density-based objective functions is proposed. Upon demonstrating the method remains computationally infeasible, a new distribution methodology is presented, which is based upon a stippling methodology with roots in computer graphics. Next, the various distribution schemes are then compared, with the focus on robustness, accuracy, and efficiency, wherein the proclaimed benefits of stippling are shown on the basis of numerical simulation, analysis of synthetic images, and an experimental case of the flow over a backwards facing step.

While it is desirable to investigate more elaborate and/or optimal objective functions, this is not possible without a distribution method that can accurately realise a robust distribution. As such, sample distribution is the focus of this chapter, rather than adaptive vs structured sampling per se; although a brief comparison of such is provided in the results section. Despite the improvements, unstructured adaptivity has a number of challenges that won't be improved by more optimal objective functions or improved implementations, primarily the dependence on random behaviour. Semi-structured adaptive sampling, introduced in the next chapter, addresses these issues and represents a more attractive method for implementing adaptivity. Yet, the flexibility of fully unstructured sampling is still desirable, particularly if a more optimal objective function can be derived, hence this work represents a step forward in realising the potential of fully unstructured adaptive sampling.

5.2 Existing Solutions

The method adopted by Theunissen *et al*.[59], originally developed by Secord *et al*.[89], is referred to as the PDF Transform method, or simply 'PDF method'. Sample locations are allocated following a regular sampling of the inverse cumulative density of the objective function, as illustrated for one dimension in Figure 5.1. While this approach is effective in one dimension, extension to multiple dimensions leads to significant

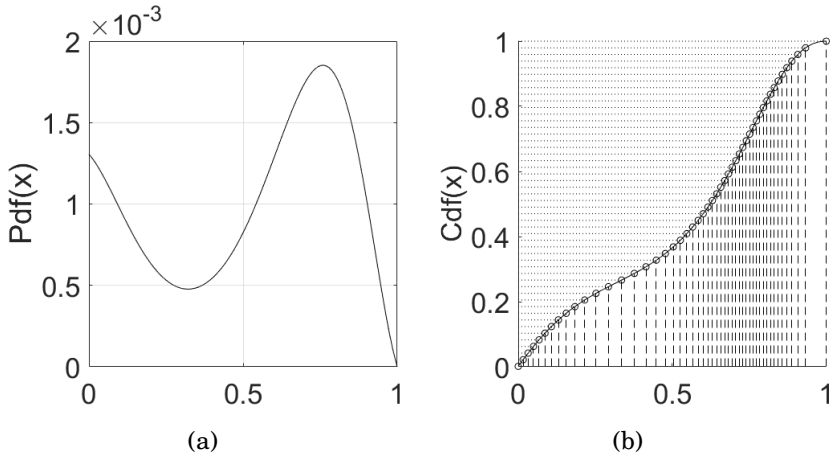
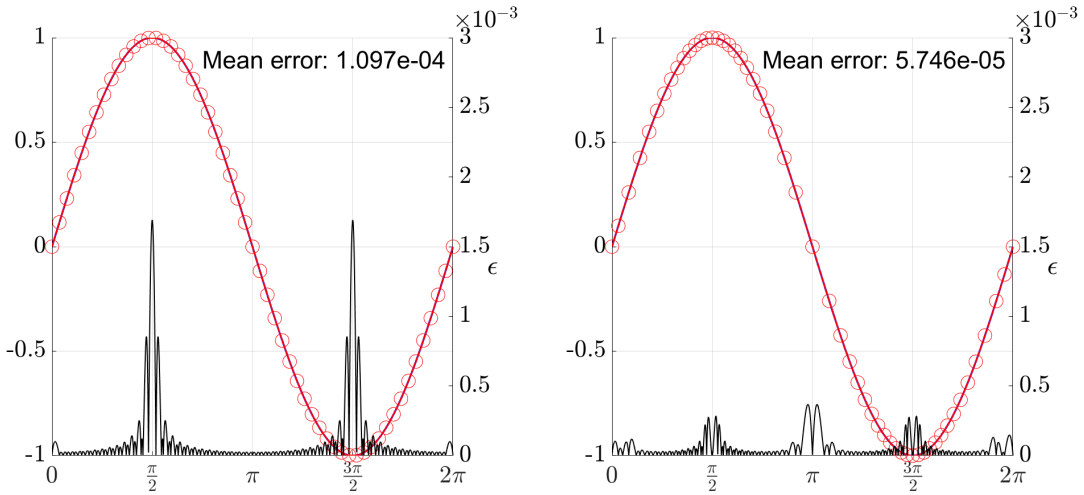


Figure 5.1: (a) Arbitrary probability density function and (b) its cumulative density function (CDF), with horizontal sample locations chosen by regular sampling of the inverse of the CDF.

clustering of samples, since samples are placed independently of eachother, without considering adjacent sample locations. The effect of such clustering is shown in the numerical assessment in section 5.5. Laplacian smoothing can be adopted to ameliorate the distribution, yet the results remain unsatisfactory and, as stated by Yu *et al*.[102], correspondence between the desired objective function and final sampling distribution is reduced.

The performance of this approach in one dimension is demonstrated in Figure 5.2 for a sinusoidal signal with unit amplitude, sampled at 60 locations and interpolated using a cubic spline. Figure 5.2(a) represents equispaced sampling, whereas samples in Figure 5.2(b) are adaptively located, according to the second derivative acting as a heuristic for curvature, using the PDF transform described above.

In light of these shortcomings, Yu *et al*.modified a mesh distribution method, originally by Persson and Strang [90], for PIV application wherein an initial distribution of samples is iterated towards their ideal locations by use of a spring-force analogy, as depicted in Figure 5.3. Accordingly, this method is referred to as the Spring Force Distribution (SFD) Method. The initial distribution is created using a quasi-random Halton sequence [105], from which repulsive forces between samples are calculated. The forces are found by comparing the current inter-sample separation, L_{ij}, with the desired separation, h_j, defined by the objective function, as shown in 5.1 Attractive forces, representative of inter-sample distances being too large, are not allowed as this complicates the force system solver [106].



(a) (b)

Figure 5.2: (a) Influence of signal curvature on reconstruction error, considering a sinusoidal signal of unit amplitude (red) imposing (a) equispaced sampling (red dots), and (b) adaptively located samples with greater frequency in regions with greater signal curvature, as per Secord [103]. Black lines depict the absolute error between the imposed signal and reconstruction adopting a cubic B-spline. The mean of the error magnitude is presented for each case.

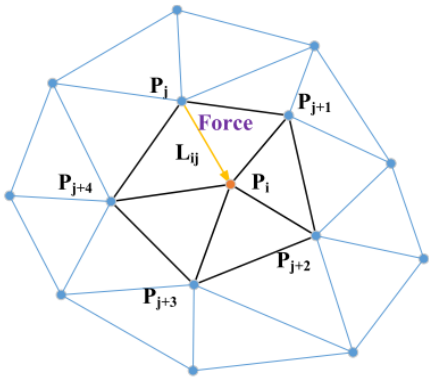


Figure 5.3: Illustration of how the force is applied between nodes for the SFD distribution method. Image from [102]

$$\left|F_{ij}\right| = \begin{cases} k\left(h_j - L_{ij}\right) & \text{if } h_j > L_{ij} \\ 0 & \text{if } h_j < L_{ij} \end{cases} \tag{5.1}$$

The resulting system of forces then perturbs the nodes over a small timestep, Δt, following the forward Euler method, 5.2, where $\boldsymbol{p}_n$ is the location of some arbitrary point for iteration n and $\boldsymbol{F}$ represents the sum of forces on $\boldsymbol{p}$ calculated from 5.1.

$$\boldsymbol{p}_{n+1} = \boldsymbol{p}_n + \Delta t \boldsymbol{F}\left(\boldsymbol{p}_n\right) \tag{5.2}$$

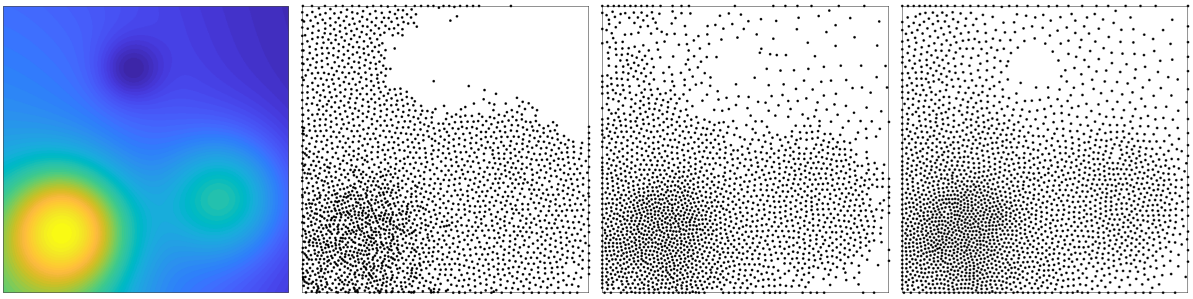
Larger Δt promotes faster convergence, at the expense of stability. The process is repeated until the perturbation of all nodes is less than some imposed tolerance [1].

A mesh quality quantity can be obtained by calculating, for each triangular facet in Delaunay triangulation, the ratio of inscribed circle diameter to the radius of the circumscribed circle, often denoted by q. The meshes produced by this method consist of near equilateral triangles, observed by $q \to 1$, which are argued in [107] to result in reduced reconstruction error, and were demonstrated to reduce the error of PIV analysis when compared to a structured approach [77].

Despite these improvements, the SFD method carries a number of significant challenges, namely user-sensitivity and computational performance, which will be demonstrated later in Section 5.5. The absence of attractive forces means that regions with too-low a sampling density must wait until neighbouring overly-dense regions displace nodes inwards. As such, when the dynamic range of the objective frunction is large, the number of iterations required to converge may drastically increase. To mitigate this, additional control logic is imposed at regular iteration intervals, to add or remove samples in regions where the discrepancy between current and target spacing exceeds a predefined threshold. However, the number of removed or added samples is not necessarily zero-sum, and so additional steps must be taken to ensure the total number of samples remains close to the desired number. Unfortunately, such thresholds are not quantified in the works of Yu *et al.*.

Furthermore, the frequency at which such density control is imposed must be tuned; too infrequent and the method becomes slow to converge, too frequent and the method may once again become unstable. Additionally, since re-calculation of the triangulation is required each time points are added or removed, increasing the frequency may significantly increase computationl cost. Again, the frequency of such is not specified in their paper.

[1] See [90] for a detailed description of the original method along with the source code.



(a) (b) (c) (d)

Figure 5.4: Demonstration of the influence of maximum iteration counter on the resulting distributions obtained through the SFD methodology, imposing the Franke function (a) as objective sampling density with 2,500 samples. (b)-(d) Solutions after the 8th, 17th, and 26th iterations, respectively.

Another challenge with the SFD method, is that while iterating nodes towards their equilibrium, they must be allowed to move with sub-pixel locations. It is possible to restrict sample locations to integral locations throughout the whole process, although this can cause the method to stall in the case that the node's displacement is less than 0.5px. Alternatively samples with sub-pixel location can be used during distribution, and the final locations rounded to integer locations, however, this may result in multiple samples per pixel which require detecting and removing.

Finally, the method lacks guaranteed termination [90], requiring an artificial, user-defined, iteration limit to be imposed. This limit has significant influence on the performance of the method and varies depending on the number of samples and dynamic range of the desired sampling density. Early termination may result in unrepresentative sample distributions, whereas late termination carries significant computational penalty. To complicate matters further, this limit must also be considered in conjunction with the density control frequency, since artificial termination shortly following density control is likely to result in very poor quality distributions. This behaviour is illustrated in Figure 5.4, wherein the desired sampling density is defined by Franke's function, (5.3) [108], and the SFD is used to produce a sample distribution with density control performed every eight iterations.

$$\begin{aligned} f(x,y) = &\frac{3}{4}\exp\left(-\frac{(9x-2)^2}{4}-\frac{(9y-2)^2}{4}\right)+\frac{3}{4}\exp\left(-\frac{(9x+1)^2}{49}-\frac{9y+1}{10}\right)+ \\ &\frac{1}{2}\exp\left(-\frac{(9x-7)^2}{4}-\frac{(9y-3)^2}{4}\right)-\frac{1}{5}\exp\left(-(9x-4)^2-(9y-7)^2\right) \end{aligned} \tag{5.3}$$

Although it is often possible to achieve a good balance of parameters on a case by case

basis, it is not possible to obtain optimal parameters for a wide range of applications. User input is therefore required, in turn limiting robustness and universality. Furthermore, given the trend of increasing camera sensor resolution, and consequently number of samples, safeguarding computational efficiency becomes pivotal.

Neither method is suitable as a sample distribution method in their current format. Due to the appealing nature of the quality of the SFD distributions, the next section will cover the steps that are taken to make this method more robust, and subsequently the modifications which allow the method to be more general in nature. Despite these improvements, it will become clear in Section 5.5 that the method remains fundamentally too computationally inefficient for application to adaptive PIV analysis.

5.3 Building on the SFD method

To address the issues of robustness and high case-dependency, this section aims to explore and select more general and optimal parameters for the SFD, which are somewhat missing in the original works. It will be shown that despite best efforts, the method remains unviable, and so to facilitate juxtaposition with the new method proposed in Section 5.4, the transformation from distance- to density-function is also generalised. An outline of the process is shown by Figure 5.5, but will be explored in more detail throughout this section.

5.3.1 Parameter development

The expensive part of the SFD method lies in the repeated construction of the Delaunay Triangulation. Steps are taken to reduce the number of times the triangulation is determined, for example only fully rebuilding if the maximum nodal displacement is above some threshold, as shown in Figure 5.5. Yet this is by itself is insufficient and is mainly hindered by the lack of attractive forces.

As previously mentioned, density control logic can ameliorate the situation by adding or removing points where the discrepency between the current and desired inter-sample spacing is deemed to be too large. Undesirably, the behaviour of the algorithm is particularly sensitive to the imposed thresholds, and additional steps are required to ensure a net-zero change in the total number of samples. Furthermore, several spring-force iterations following density control are required to smooth the distribution to the desired level As a result, there must be sufficient iterations between the control logic

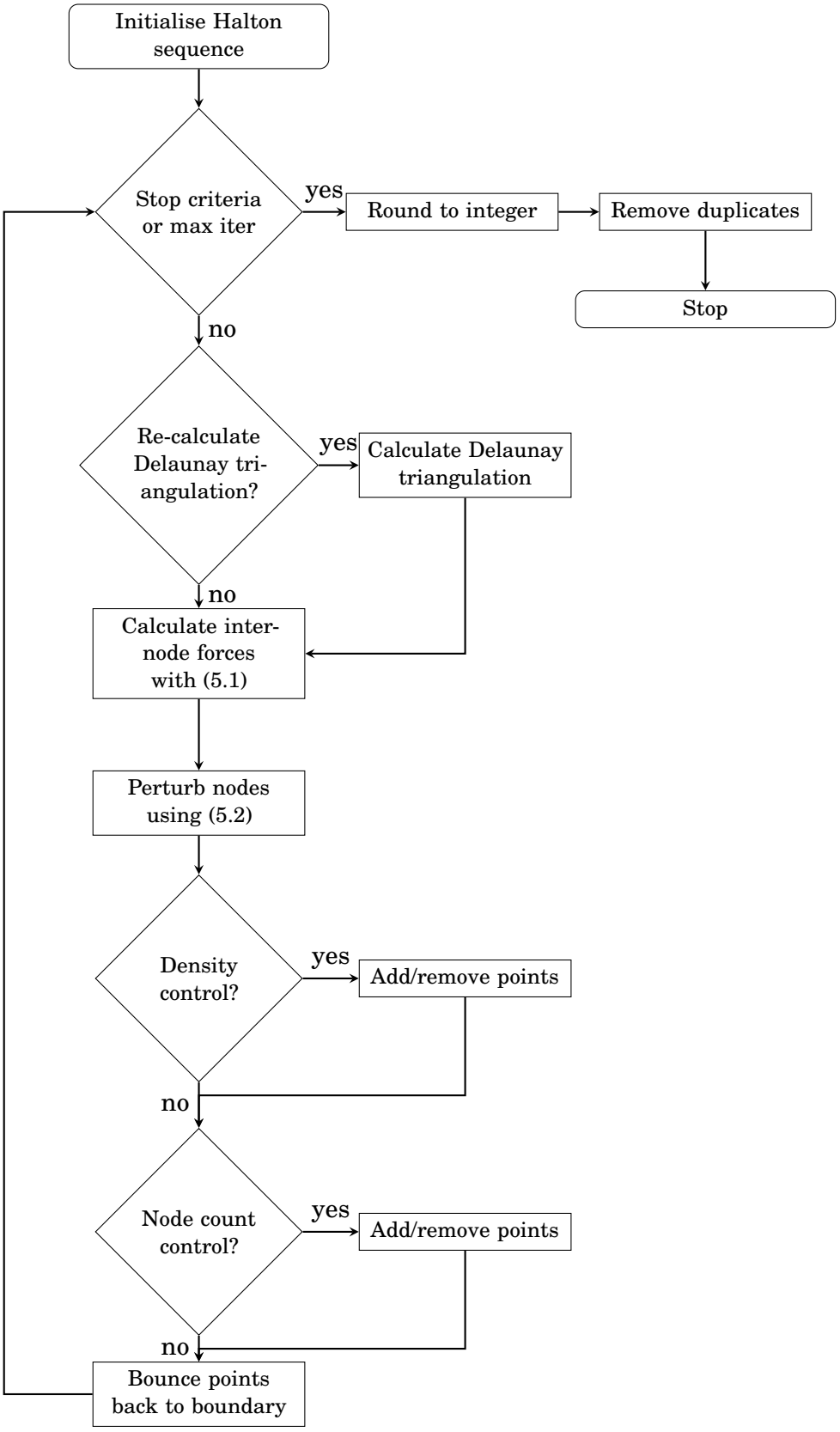


Figure 5.5: Flow chart related to the SFD process.

application, otherwise unstable oscillatory behaviour is likely to occur. To complicate matters further, the maximum iteration counter must be chosen considering the density control frequency, otherwise the final distribution is likely to be in an intermediate state.

Although the interaction between each setting is complex, it is generally valid that the more readily points are added or removed by the density control, the more intermediate iterations are required maintain stability. Defining h_d as the desired inter-sample distance, and h_a as the actual inter-sample distance, the following thresholds were identified through extensive trial-and-error investigations:

- Dense threshold: $h_d > 2.2h_a$
- Sparse threshold: $h_d < 0.6h_a$
- Density control: every eight iterations
- Maximum iterations: 55

In addition to the parameter settings, the paper presented by [102] does not discuss handling masks. In the original method and paper of Persson *et al*. [90], however, boundaries are handled by imposing a repulsive force, proportional to the distance beyond the boundary. The distance in this case was found by combining a number of simple analytical functions in order to build more compound domain geometries. However, such approach is simply not feasible for typical PIV applications due to the complexity of mask shapes encountered. Accordingly, a pre-processing step is required to numerically calculate the distance of each pixel to the nearest boundary.

Attempts to alter these parameters to improve the computational efficiency, i.e. more frequent imposition of density control logic, stricter[2] thresholds, larger timestepping, or a reduced maximum iteration counter, quickly deteriorate the stability of the method. The strength of the method lies in the spring force iterations, which effectively act as a smoothing operator with an ideal density, rather than a distribution method.

5.3.2 Generalising Yu *et al.*'s Spring-Force Distribution method

As per Section 4.4.1, objective functions are typically of the form of (5.4), where the local seeding density, $S_d(x,y)$, and chosen flow metric, $\xi(x,y)$, are individually weighted, by some scaling terms α and β, respectively, and combined to obtain the overall sampling objective function, $\phi(x,y)$. Considering that seeding and many flow metrics, e.g standard

[2] In this context, this refers to more readily adding or removing points

deviation[59], vorticity, or flow curvature, correlate positively with the desired sampling behaviour, i.e. more samples where the values are larger, they do not need additional pre-processing to be suitable for a distribution method such as the PDF transform.

$$\phi(x,y) = \alpha\xi(x,y) + \beta S_d(x,y) \tag{5.4}$$

The SFD method of Yu *et al*. [77], however, requires input in terms of the desired inter-sample spacing, $h(x,y)$, rather than sampling density, i.e. regions with smaller values of $h(x,y)$ will be attributed more samples. The objective function, $\phi(x,y)$, therefore, requires modification before it may be used as input for the SFD.

The approach taken by Yu *et al*.is to first obtain the initial inter-sample spacing, $h_0(x,y)$, according to $S_d(x,y)$ alone using (5.5), where WOR is the desired global average window overlap ratio, and NI_t is the target number of particle images per correlation window.

$$h_0(x,y) = (1 - WOR)\sqrt{\frac{NI_t}{S_d(x,y)}} \tag{5.5}$$

In subsequent iterations, flow adaptivity is incorporated by converting flow metrics, i.e. $\xi(x,y)$, into their distance-based counterparts through use of a transformation function, $g(\cdot)$, shown in (5.6).

$$\xi^* = g(\xi) = \begin{cases} -0.5\xi^{0.88} + 1.5, & 0 < \xi < 1 \\ 0.4\xi^{-1.1} + 0.6, & 1 \leq \xi \end{cases} \tag{5.6}$$

The output from $g(\xi)$ is a local scaling term $\xi^*(x,y)$, limited between 0.6 and 1.5, to adjust the existing inter-sample spacing, h_0, to obtain the new target inter-sample spacing, as per (5.7), which now incorporates flow adaptivity.

$$h(x,y) = h_0(x,y)\xi^*(x,y) \tag{5.7}$$

The motivation for limiting refinement due to flow metrics is to limit 'excessive refinement or coarsening of the sampling points' [77]. Such limitations may be conducive to producing high quality sampling distributions, however, the effect is not dissimilar to the effect of retrospective mesh-smoothing. As such, the obtained distribution may no longer faithfully reflect the input objective function, as demonstrated by Figure 5.9(g), which Yu *et al*. previously argued as a drawback to the PDF transform method. Furthermore, the bounds in (5.6) may need to be selected on a case-by-case basis according to the underlying displacement scales and image quality, requiring significant tuning to obtain an appropriate, smooth, scaling function.

This scaling function also complicates fair comparison between methods, since, using the same flow metric and seeding density estimate may result in significantly different distributions[3]. Effectively, the objective function is implicitly manipulated, making it difficult to distinguish variations in reconstruction error due to distribution method or choice of objective function.

Therefore, a new, unbounded, transformation function is required to create distributions analagous to those produced by other methods. Additionally, to facilitate modularity, the transformation function should act on the entire objective at once, as opposed to separating seeding and flow adaptivity into individual steps. The proposed transformation function, g', is presented in (5.10), and is based on a similar philosphy to the adaptive incremental stippling (AIS), which will be introduced in the following section. The idea is that the summation of objective density function, $\phi(x,y)$, within a disk of diameter $h(x,y)$, surrounding each disk, is constant. With the simplifying assumption that $\phi(x,y)$ is approximately constant within $(x \pm h(x,y), y \pm h(x,y))$, then the following equality holds true

$$\frac{\pi h^2(x,y)}{4}\phi(x,y) = F \tag{5.8}$$

Where F is calculated using (5.9), with N_w being the desired number of windows in distribution.

$$F = \frac{\iint \phi(x,y)dxdy}{N_w} \tag{5.9}$$

Subsequently, the inter-sample distance function, $h(x,y)$, is therefore

$$h(x,y) = g'(\phi(x,y)) = \sqrt{\frac{4F}{\pi\phi(x,y)}} \tag{5.10}$$

Importantly, while (5.10) allows for unbounded conversion from density function to distance function it does not prohibit the local relative densities from being bounded. Such bounding may be conducive to improved adaptive sampling performance, yet, this should be applied explicitly during the construction of the objective density function, and not implicitly during distribution. For example, consider the objective function of the form of (5.11), as it (effectively) is in the works of Yu *et al*.

$$\phi(x,y) = S_d(x,y)\xi(x,y) \tag{5.11}$$

Substituting (5.11) into (5.10), and dropping the (x,y) for clarity, yields

$$h(x,y) = \sqrt{\frac{4\iint \phi\, dxdy}{\pi N_w S_d \xi}} \tag{5.12}$$

[3]This behaviour is demonstrated in Section 5.5, c.f. Figure 5.9

The number of windows, for a desired average WOR, is given as (5.13), where NI is the target average number of particles per window, and N_p is the total number of particles within the image.

$$N_w = \frac{\iint S_d \, dxdy}{(1-WOR)^2 NI} = \frac{N_p}{(1-WOR)^2 NI} \tag{5.13}$$

Therefore

$$\begin{aligned} h(x,y) &= \sqrt{\frac{4}{\pi}} \sqrt{\frac{(1-WOR)^N I \iint \phi \, dxdy}{N_p S_d \xi}} \\ &= \sqrt{\frac{4}{\pi}} \sqrt{\frac{\iint \phi \, dxdy}{N_p}} (1-WOR) \sqrt{\frac{NI}{S_d}} \sqrt{\frac{1}{\xi}} \\ &= \sqrt{\frac{4}{\pi}} \sqrt{\frac{\iint \phi \, dxdy}{N_p}} h_0(x,y) \sqrt{\frac{1}{\xi}} \\ &= h_0(x,y) \sqrt{\frac{1}{\xi}} \end{aligned} \tag{5.14}$$

Within the SFD algorithm, relative lengths are used[4], as opposed to absolute lengths, the constant term, C, is subsequently cancelled. In fact, while using (5.10) will return the actual intersample distance, it is sufficient for implementation to use the square root of the inverse of the objective function. Following (5.14), the exact same distribution as Yu *et al*. can be obtained by, therefore, using

$$\phi(x,y) = S_d(x,y) \frac{1}{g(\xi)^2} \tag{5.15}$$

5.4 Adaptive Incremental Stippling

A new method is proposed based on AIS, which was initially developed by Ascencio-Lopez *et al*. [109] as a means to rapidly construct stippled[5] images, such that the stipple's spatial density is reflective of the local image intensity. Before this method may be used for PIV sample distribution, however, a number of modifications are required, such as the ability to define a desired number of samples, and achieve this number with accuracy, as well as the handling of boundaries.

It is worth first describing the original method; In this method, each stipple is surrounded by a disk of radius h which is not permitted to overlap with any other disk. By varying the radius of the disks according to the local image intensity, i.e. smaller

[4] This is to avoid over-constraining the system of forces[90].

[5] A stipple is simply a small dot. In computer graphics, they may sometimes be referred to as a primitive, however, 'primitive' is a broader term and may also refer to small hatchings or patterns.

disks where the image is darker and thus needs more stipples, the spatial density of the resulting distribution is representative of the underlying image intensity, while maintaining good distribution qualities free from excessive clustering or voids. A brief outline is shown in Figure 5.8.

Given an image $I(x,y)$ of size $L_x \times L_y$ scaled from $[0,1]$, we define the image density, $\phi(x,y)$, as (5.16)[6].

$$\phi(x,y) = 1 - I(x,y) \tag{5.16}$$

To initialise the process, a disk is placed randomly within the image domain of size M x N, and is ascribed a radius according to (5.17).

$$h_1 = \sqrt{\frac{FMN}{\iint \phi(x,y)}} \tag{5.17}$$

In the original method, F is left as a user-defined constant which varies the overall darkness of the stippled image.

From h_1, the disk radius is varied until the sum of interior image density is approximately equivalent to the user defined F, within some tolerance, where d indicates the extend of the disk.

$$\sum\sum_{d} \phi(m,n) = F_d \approx F \tag{5.18}$$

Once the correct radius is obtained, the disk is added to a stack of active locations, and drawn onto a binary disk buffer, of equal dimensions of the image and initially all empty valued. When a disk is accepted, any pixel within the radius of the disk is filled, such that overlap between disks can easily be determined. This initialisation process is distinguished from the main body of the algorithm in Figure 5.8 by dashed lines, since this can be modified to allow for better representation of image boundaries and will be described later.

The main loop proceeds by taking the topmost disk in the active list and calling it D_a. Around the disk D_a, a candidate disk, D_c, with radius $h_c = h_1$, is placed at a random angle, α, adjacent to the active disk, i.e.

$$\begin{aligned} x_c &= x_a + (h_a + h_c)\cos\alpha \\ y_c &= y_a + (h_a + h_c)\sin\alpha \end{aligned} \tag{5.19}$$

The radius of the disk, and hence the location (x_c, y_c), is then adjusted until it encloses an amount F of the image density. Once sized, the disk is checked for overlap with any

[6]The use of ϕ to symbolise the image density, as well as the objective function in the previous section, is intententional to highlight the similarity between image density and objective function density.

other disk in the disk buffer. Assuming there is no overlap, the disk D_c is accepted and added to the stack of active locations. If overlap *is* detected, then D_c is rejected and a failure is recorded.

Candidate disks continue to be generated around D_a until a threshold of failures is exceeded, i.e. there is no more room for properly sized disks adjacent to D_a. At this point, D_a is placed in the output points list, and the topmost disk in the active list is popped to become the new D_a. The process continues until the active list is empty, at which point the image will be completely stippled.

By replacing the image density with the target objective function, this method can be adapted to create a high-quality unstructured distribution of sample locations for spatially adaptive PIV applications. The original method requires a user-defined input value, F, which indicates how much darkness each stipple should represent, and implicitly defines the final number of stipples/samples. In the context of PIV, on the other hand, it is the number of windows which is defined, either explicitly by the user, or implicitly through some adaptive calculation (see Section 4.4.2. The value of F can be obtained through the relationship presented in (5.9).

For stippled images, the final number of stipples is unlikely to be of much interest to the user, however, within PIV it directly influences the resulting solution accuracy, c.f. Figure 4.2, and computational requirements. Accordingly, it desirable to have more precise control over the number of samples. This can be achieved by considering the fact that (5.9) implicitly assumes that all pixels will be covered by a disk, which is not possible when packing non-overlapping circles into a domain.

The ability to pack regular circles into a shape is a common mathematical problem [110, 111, 112], and packings are ascribed a packing density, which describes the ratio of cumulative area covered by all circles to the total available space. The packing densities of the best known packings for up to 200 circles is shown in Figure 5.6, wherein it can be seen that optimal packings have densities in the region of 0.75-0.85. These packings, however, represent an optimal, or at least the best known, arrangement of circles, whereas randomly packed circles have packing densities in the range 0.6-0.7.

The value of F which would yield the desired number of samples can hence be found with (5.20), where η is the packing density.

$$F^* = \eta \frac{\iint \phi(x,y) dx dy}{N_w} \tag{5.20}$$

However, packing density is further influenced by non-uniformity of the circle diameters to be packed, which is difficult to predict [113]. Since the variation of circle

diameter is dependent on the particular objective function, it is not practical to define a universal value for η

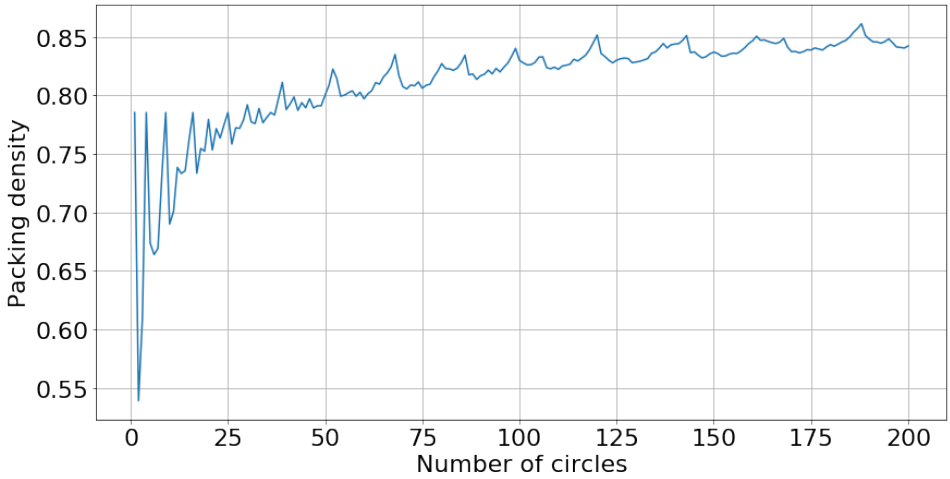


Figure 5.6: Best known packing densities for up to 200 circles within a unit square. The densities for 1-30, inclusive, and 36 are known to be optimal, the rest are simply the best known.

Despite η varying for different N_w, ϕ, or domain shape, it remains reasonably consistent for a given combination of such parameters. This is demonstrated by Figure 5.7, which shows the fluctuations of η for a given objective function. Due to this consistency, the packing density can be estimated iteratively.

The algorithm is first allowed to run with F as per (5.9)[7]. From this, an estimate of η may be used to obtain $F^* = \eta F$ as per (5.20). Subsequent distribution generations can further refine η, allowing the actual number of distributed samples to tend towards N_w. A tolerance of N_w of 3% is typically achieved within two or three repeats of the algorithm. Despite the need to repeat the distribution process, this method remains efficient, as is shown in the following section, due to its linear time complexity and avoidance of mesh computation. Nonetheless, a method to predict η in advance would bring considerable computational speed-up.

In addition to the modifications and improved N_w control above, it is important to be able to sample boundaries effectively in PIV. The boundaries can be easily obtained from the mask using one of many edge detection routines available. Then, the objective function along these boundaries can be extracted and samples allocated according to the

[7]It is, unfortunately, necessary to allow the entire domain to be populated before is valid η, since it is known to vary with fluctuations of ϕ. As such, estimating η following the population of a small subsection of the domain is unlikely to yield beneficial results.

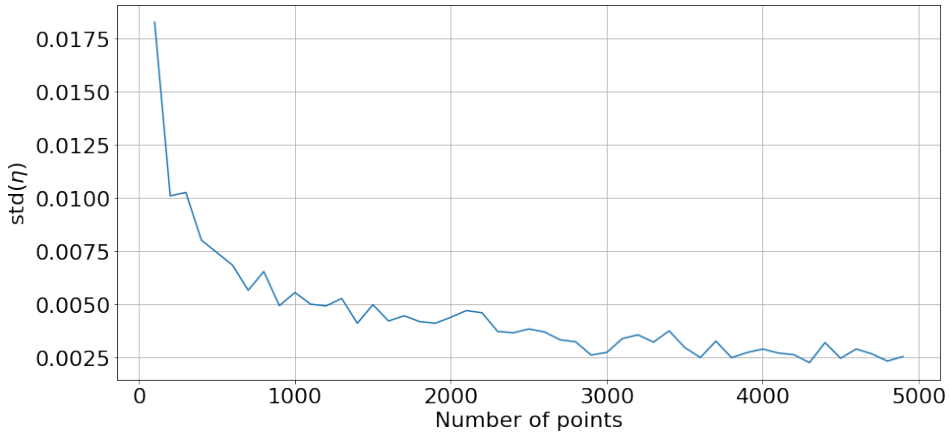


Figure 5.7: Standard deviation of the packing density as a function of N_w, distributing samples according to an objective density function defined by (5.3).

one-dimensional PDF transform of Secord [89]. Using the one-dimensional transform method of Secord results in nicely spaced sampled along the boundary, which can then be used to initialise the AIS algorithm, in lieu of the random seed.

5.5 Results

The performance of each of the sample distribution methods is assessed in four ways; visual comparison, numerical assessment, application in synthetic PIV image analysis, and finally, application in experimental PIV image analysis. Within each setting, the distribution quality of the AIS is shown to be similar to that of SFD, while simultaneously the computational costs are shown to be similar to that of the PDF method. On the principle that both of the existing distribution methods, i.e. PDF and SFD, have been demonstrated to outperform a conventional, structured, analysis, in both synthetic and experimental settings [77, 59], comparison of the AIS to a structured approach is not given here. The intention of this section is to provide an apples-to-apples comparison of the available unstructured distribution methods.

5.5.1 Spatial sample allocation

In the visual comparison, two objective density functions are imposed, one being uniform while the other being defined by Franke's function, (5.3). Four different methods are investigated at this point; The two dimensional PDF method of Theunissen *et al*. [59],

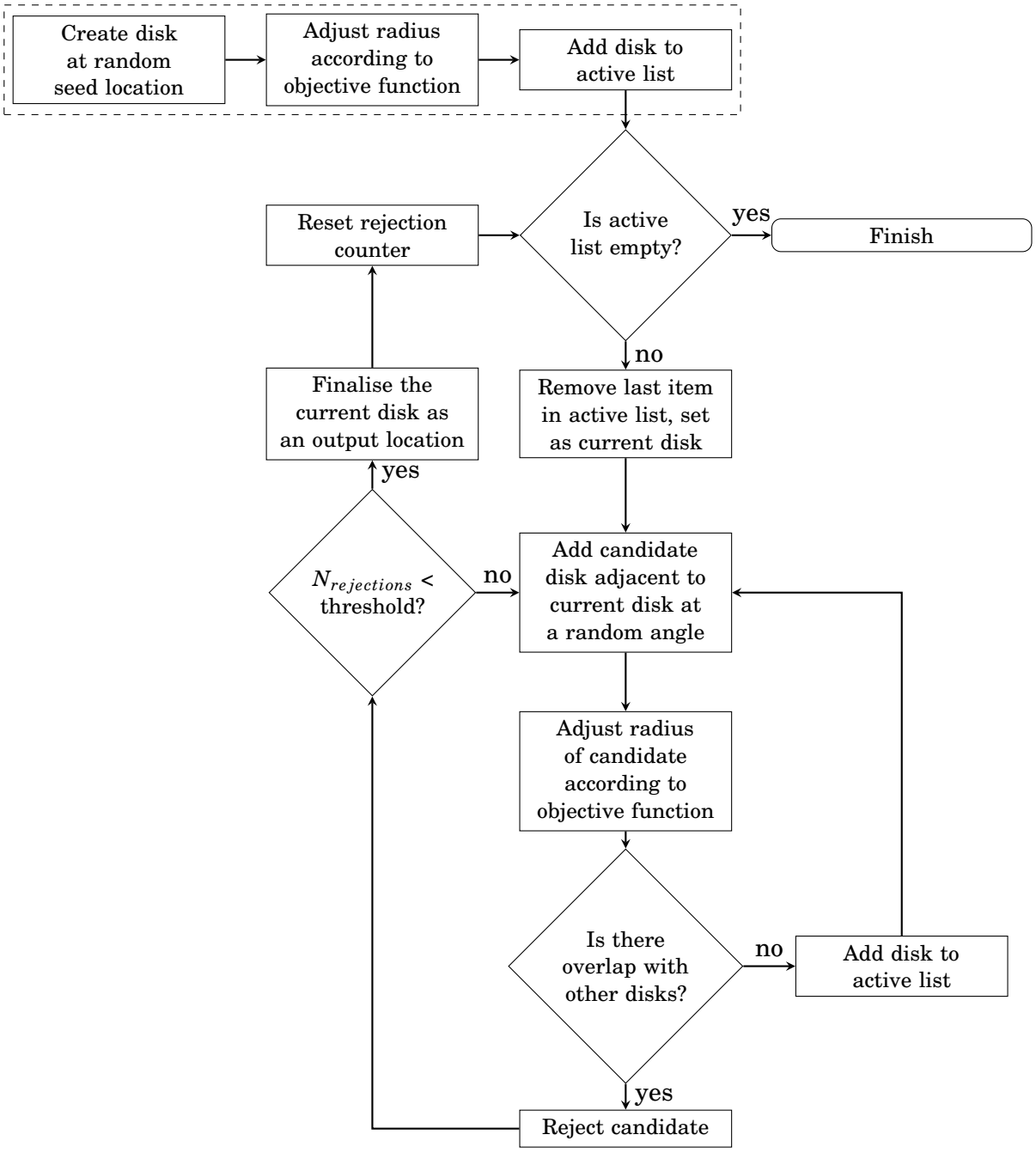


Figure 5.8: Flow chart related to the AIS process.

the proposed AIS method, the SFD method as proposed by Yu *et al*. [77][8], where the input objective density function is converted to a distance function using (5.6), and, finally, the SFD method wherein the objective density function is transformed using (5.10). The resulting distributions are shown in Figure 5.9. Approximately 3,000 samples are distributed in each case.

As described in Section 5.2, the PDF method suffers from signifcant since the placement of individual locations does not consider the existing distribution. This is apparent in Figures 5.9(a) and 5.9(e), whereas, clustering is much less significant in the other three presented methods. The distribution produced using the AIS method appears much better, compared to the PDF method, and there is no significant local clustering of samples, however, there does remain some visual artefacts within the distribution. The homogeneous distributions produced by either SFD method are, as expected, practically identical, and exhibit the most regular of inter-sampling spacings. However, the effect of the bounded transformation function is evident when comparing their distributions produced for a non-uniform objective density function, as seen in Figures 5.9(g) and 5.9(h), wherein the spatial variations of sampling density are far less noticeble using the bounded transformation. While the distribution produced by the (new) SFD approach, i.e. Figure 5.9(h), does appear to be of higher quality than that of the AIS method, i.e. Figure 5.9(f), the apparent differences are small when compared with the equivalent PDF distribution, Figure 5.9(e). Owing to the lack of generality of the bounded transformation method, the remaining results pertaining to the SFD method will only consider the general transformation method.

5.5.2 Computational effort and accuracy

The three distribution methods, AIS, PDF, and SFD, are assessed in terms of computational requirements and interpolation accuracy on the basis of Monte Carlo simulations. Run-time performance was assessed for a range of sample quantities, N_w, distributed according to an objective density function obtained by evaluating Franke's function over a domain of size $1000px \times 1000px$[9]. For each N_w, 300 distributions were created and, subsequently, the trends depicted in Figure 5.10(a) represent the average of the 300 run-times, normalised by the duration required to distribute 1,000 using the

[8]While the authors did not present the chosen parameters for their implementation, trial and error was used to obtain suitable values, which were presented in Section 5.2, which are considered to result in a representative performance for the SFD, balancing robustness and efficiency as much as possible.

[9]Normally, Franke's function is only defined over the domain (x, y) ∈ [0, 1]. Therefore, the input location is scaled accordingly to obtain the Franke function defined over the desired domain size.

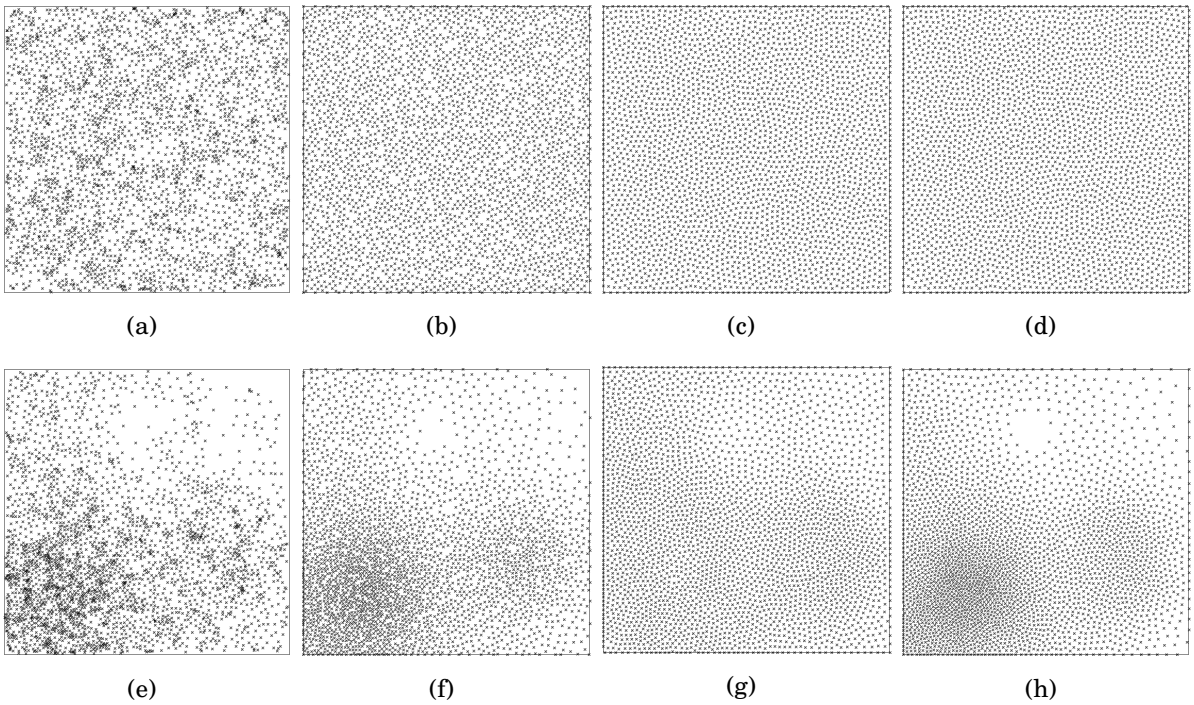


Figure 5.9: Visual comparison of sampling distributions, each for 3,000 samples, produced by the investigated methods. From left to right: 2D PDF transform method, proposed AIS method, SFD method using (5.6), and SFD method using the general (5.10). Top row objective density function is uniform, whereas, the bottom row uses Franke's function as per (5.3).

AIS method. While differences in implementation, for the same algorithm, may have a noticeable impact on the observed performance, the differences presented here are in the orders of magnitudes. Importantly, the trend of run times shows how the computational costs vary with N_w, wherein it can be seen that the costs for both the SFD and PDF grow much quicker than for the AIS approach. Noting that a relatively normal 2k x 2k image with $S_d = 0.035$, $NI_t = 8$, and $WOR = 0.5$ would result 70,000 windows according to (5.13). Considering the trend of image size, the trend of computational performance here is, therefore, critical. This performance difference is further amplified, in absolute terms, by the need to re-evaluate the sampling distribution for each iteration in a multi-pass analysis. It is important to note as well that the computational requirements for the SFD depend on the uniformity of the objective function, since more iterations will be required for less uniform objective functions.

Interpolation accuracy is quantified by considering the average relative error between an interpolated surface and the known Franke function. For each distribution method,

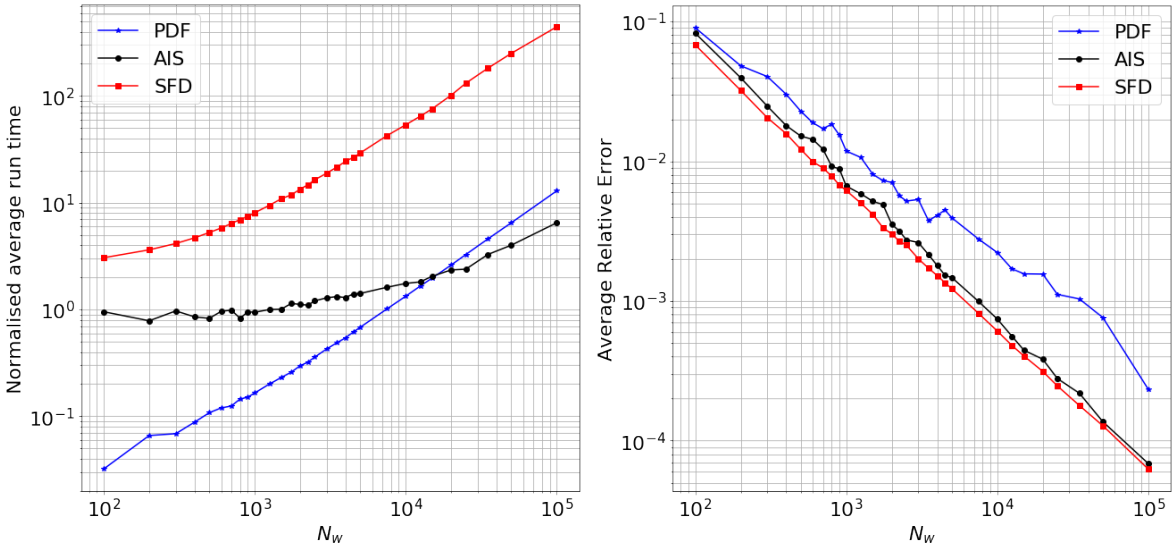
N_w samples are distributed homogeneously over the domain, and the exact value of the underlying function obtained at each location. The sampled values are then interpolated onto a pixel-wise grid using a natural neighbour interpolation scheme, which has shown to perform well in both PIV using feature tracking[94], and particle tracking velocimetry[95]. For each pixel location, the absolute relative error is calculated, and averaged over all pixels and distributions. The result, shown in Figure 5.10(b) gives an idea as to the typical error per pixel for an instantaneous snapshot. This, it is argued, is more important than calculating, for example, the spatially averaged bias error, since these distributions are intended to be used for instantaneous adaptive sampling.

Differences between AIS and SFD can be seen to be small when compared to the error caused by PDF sampling. The majority of the increased error arises from poor interpolation quality near regions of clustering, and from poor sampling near the borders of the domain. While it was discussed in Section 5.2 that it is possible to place samples along borders prior to the main algorithm, in a similar manner to the approach discussed in the AIS method, it is not, however, then possible for the remaining sampling locations to adjust their locations accordingly. Therefore, with or without the additional sampling, the distribution quality near the boundaries is further decreased resulting in increased interpolation error. Distributing samples using the Spring-Force method on the other hand leads to lower errors but drastic increases in computational effort. The proposed stippling methodology offers a combination of speed-up while retaining interpolation accuracy.

5.5.3 Synthetic PIV image analysis

Synthetic PIV images allow the performance of a proposed method to be quantified, since the true, expected, solution is known. Whereas in the previous section, the analysis purely focused on the numerical accuracy of interpolation, analysis of synthetic images allows the effect of e.g. correlation and vector validation to be inferred. Two test cases were investigated, each with 750 images. The images were created with a mean seeding density of 0.05particles per pixel (ppp), a uniform particle image diameter of $3px$, and a peak particle intensity chosen from a standard normal distribution. Finally, image noise is omitted and a pixel fill factor of unity is used. An exemplary image is shown in Figure 5.11(c).

The first flow field tested is a two x two array of contra-rotating vortices [38], defined in (5.21), where A_0 is the maximum single-component displacement and $L_x = 1000px$ and $L_y = 1000px$ represent the domain dimensions. Secondly, a Gaussian smoothed



(a) (b)

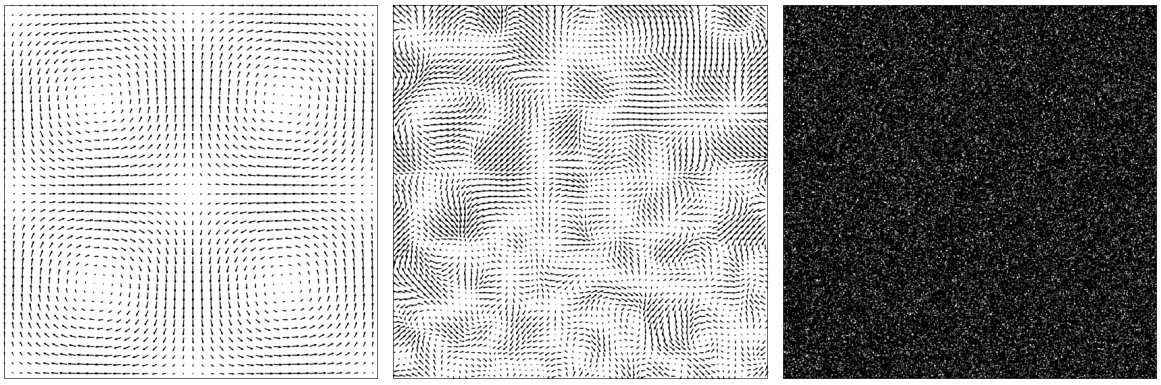
Figure 5.10: Numerical comparison of the distribution methods in terms of (a) average run-time required to distribute samples according to a spatial sampling density determined by (5.3), normalised by the time to distribute 1,000 samples using adaptive stippling (AIS) and (b) average relative interpolation error from homogeneously distributed samples.

velocity field is used wherein a moving Gaussian filter is applied to isotropic random noise, in order to produce a displacement field more analogous of turbulence. The Gaussian filter had linearly varying kernel size, from 51px at the top of the domain to 15px at the bottom, and a maximum displacement of 10px was enforced. Both flow fields are displayed in Figure 5.11.

$$u = A_0 \cos\left(\frac{2\pi x}{L_x} + \frac{\pi}{2}\right) \cos\left(\frac{2\pi y}{L_y}\right) \tag{5.21a}$$

$$v = A_0 \sin\left(\frac{2\pi x}{L_x} + \frac{\pi}{2}\right) \sin\left(\frac{2\pi y}{L_y}\right) \tag{5.21b}$$

The images were analysed according to the following algorithm. In the first iteration, 2,500 windows were distributed uniformly using one of the three scrutinised distribution methods, utilising only the seeding density as objective function $\Phi(x,y) = S_d(x,y)$. At each sample location, the AIW algorithm (Section 4.6) determines the appropriate window size, according to the seeding and displacement magnitude, which is then interpolated onto a pixel wise grid to obtain $WS_0(x,y)$. Interrogation areas were cross-correlated and sub-pixel accurate displacements were obtained using two three-point Gaussian



(a) (b) (c)

Figure 5.11: Imposed displacement fields used in the synthetic PIV image analyses. (a) Vortex Array (5.21), subsampled by a factor 24 and scaled by a factor 2 for readability, (b) Gaussian smoothed velocity field, subsampled by a factor of 16 and scaled by 2. (c) Exemplary PIV image

estimators [29]. Displacement vectors were then validated by means of the universal outlier detection algorithm [39], wherein the eight neighbours were found using a KNN search. Following a top-hat predictor filter [54], displacements were interpolated using a 3rd order polynomial fit, using the CloughTocher interpolation scheme, for image deformation or post-processing[15]. More information on the interpolation scheme and its comparison to structured interpolation can be found in 4.4.4.

Each iteration, k, the budget of correlation windows was set to $2,500 \times k$. The objective function consisted of a combination of the local spatial standard deviation of the flow and seeding density, i.e. $\Phi(x,y) = \phi(x,y) \cdot S_d(x,y)$ [59]. The window size in iteration k was calculated as per (5.22), with the final window size, WS_f, set to $15px^2$, and number of iterations, K, set to four. A fifth and final refinement iteration was subsequently performed, without adjusting the window sizes or locations to minimise residual displacements.

$$WS_k(x,y) = WS_0(x,y) - (WS_0(x,y) - WS_f)\frac{k-1}{K-1} \tag{5.22}$$

To quantify the solution accuracy, the total error was calculated for each pixel, according to (5.23), where δ is the total error, $\overline{U}$ is the measured displacement for each pixel, U_t is the imposed displacement field, σ_U^2 is the variance of U, and $|\cdot|$ represents the euclidean norm. The total error is then spatially averaged for all (x,y) and the value, per iteration, is shown in Figure 5.12.

$$\delta(x,y) = \sqrt{|\overline{U}(x,y) - U_t(x,y)| + |\sigma_U^2(x,y)|} \tag{5.23}$$

The bias magnitude for the Gaussian smoothed flow field is considerably greater than the vortex array due to the increased complexity of the flow. Ideally, this should be sampled by a greater number of correlation windows to properly spatially sample flow structures and minimise interpolation errors. Furthermore, little reduction in total error for the vortex array flow field can be noticed beyond the second iteration, indicating that fewer correlation windows may have been sufficient in this case. For both cases, the AIS and SFD performed almost identically. While the overall computation time for both the PDF and AIS approaches were similar (within 2% of each other), the SFD method was approximately 35%-40% slower compared to the PDF approach.

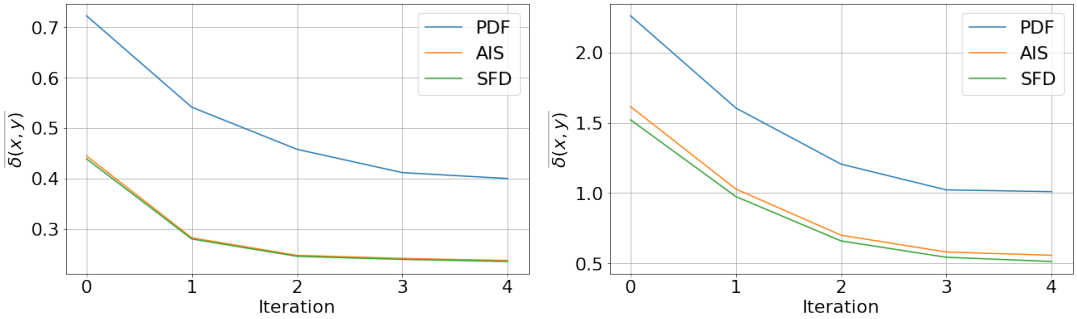


Figure 5.12: Evolution in average total error, $\overline{\delta(x,y)}$, with iteration number. (a) vortex array (b) Gaussian smoothed random velocity field.

5.5.4 Experimental application

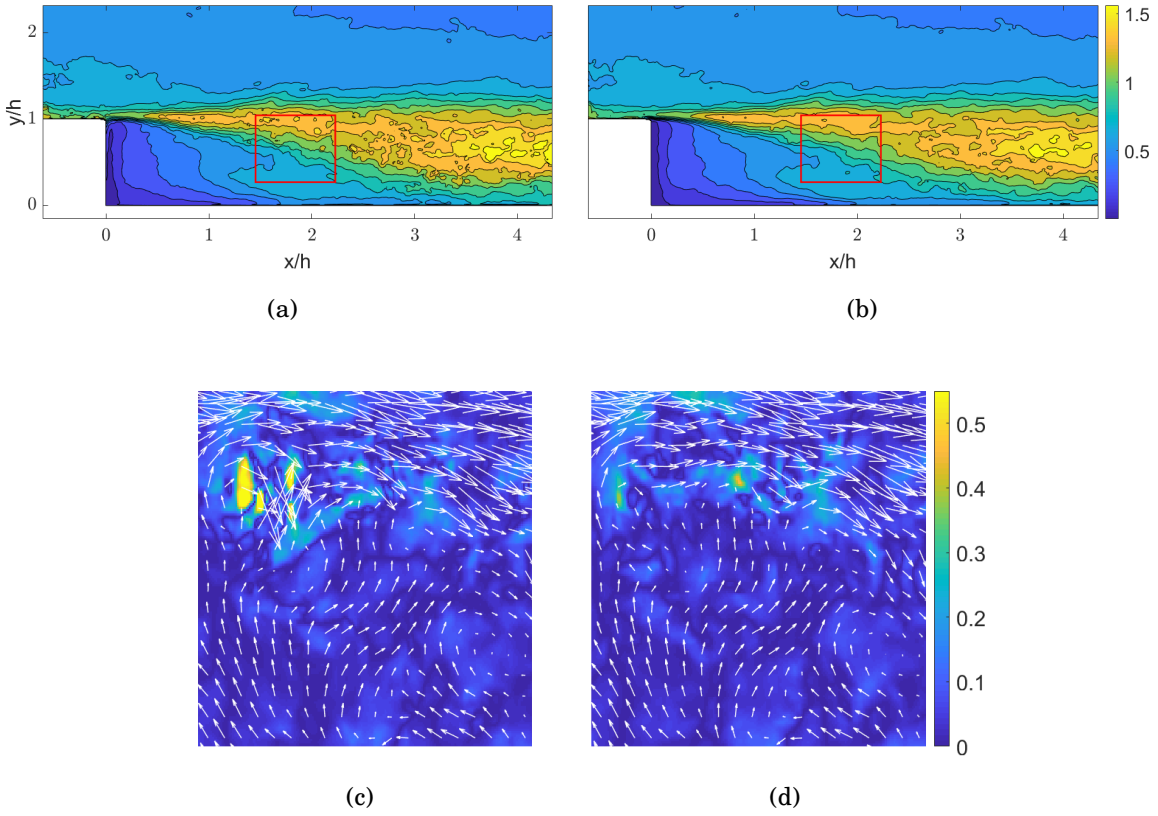
The previous assessments have illustrated the ability to distribute correlation windows by means of adaptive stippling. Distributions reflect the imposed objective function and are void of clustering, contrary to those produced using the PDF method. Compared to the spring-force approach, adaptive stippling represents less computational complexity while maintaining accuracy and thus reveals itself as the favoured sample allocation methodology.

The advantage of using AIS, or indeed SFD, over the PDF transform is lastly demonstrated on the basis of experimental PIV images. The case chosen is the flow over a backwards facing step with an expansion ratio of 1.2 at a step height-based Reynolds number of around 5000. In total 250 images were analysed using the same approach as

above, i.e. a sampling distribution according to the local spatial standard deviation of the flow field and the seeding density. The temporal standard deviation of the u-component of the displacement field is calculated and shown in Figures 5.13(a) and 5.13(b), for the solution with the PDF distribution method and the AIS method, respectively. Since both the stippling and spring-force approaches produced similar results, only the PDF and AIS results are presented hereafter. Despite similar results, the total run-time for the SFD was 60% greater than the stippling approach.

Figure 5.13 shows the standard deviation of the measured horizontal displacement component for both the PDF and AIS approaches, revealing considerably more localised spikes in magnitude, i.e noise, in the PDF results compared to the approach using AIS. A similar story, though not depicted, is observed in the v-component standard deviation. One contributor to this noise is related to interpolation accuracy; a worse interpolation results in greater instantaneous error and thus leads to increased σ in the displacement field. A second, more significant, contributor to these spikes in standard deviation values come from clusters of outliers, which fail to be detected.

An example of the effect of these clusters is presented within Figure 5.13 where a region of one of the instantaneous displacement fields is shown overlaid on the local instantaneous vorticity. Algorithms specifically developed to handle such clusters do exist, such as the works of Masullo *et al*. [41], Higham *et al*. [114], and Wang *et al*. [115], yet these methods are computationally intensive, rely on Proper Orthogonal Decomposition, requiring multiple flow fields, or both, as is the case for Wang *et al*. whereby they recommend their algorithm only for post-processing given it is too computationally intensive for a multi-pass routine. It is possible to apply POD-based outlier detection to an instantaneous flow field by sub-dividing the domain into a number of sub-regions, however, the size and number of sub region become important parameters to be tuned. Furthermore as the results of Higham show, these methods are not totally robust to outliers, particularly as cluster size increases and remains a difficult topic. Nevertheless, the improved distribution quality resulting from AIS and SFD ease this challenge, by reducing the possible size of outlier clusters, ensuring a smoother interpolation, and thus significantly reducing the chance of irrecoverably distorting the underlying image and propagating outliers into the final solution. This is perhaps the key advantage of AIS or SFD over the PDF method, and, since the AIS approach is more computationally efficient and universal, this renders it the distribution method of choice for unstructured sample distribution in PIV.



(a) (b)

(c) (d)

Figure 5.13: Pixel wise standard deviation, σ_u, over 250 displacement fields in horizontal displacement component for the flow over a backwards facing step at $Re_h \approx 5000$, using (a) the PDF transform method and (b) AIS method. (c) and (d) show close-ups of an instantaneous vector field, overlaid on the instantaneous vorticity magnitude, for the regions bounded by the rectangles in (a) and (b) respectively, demonstrating the poor vector validation encountered as a consequence of using the PDF transform approach.

5.5.5 Hybrid Method

Earlier in the chapter it was explained that the SFD method uses a quasi-uniform Halton sequence to generate an initial distribution, which is then perturbed towards the desired sampling density. Importantly, the lack of attractive forces required additional control logic, which periodically assesses whether particular nodes of the distribution are too close or too far apart, and removes or adds samples appropriately. Parameter selection for this aspect of the algorithm, i.e. the threshold for too close/far apart, the frequency at which to impose density control, and the number of points to adjust so as to not significantly alter the overall number of samples, is highly sensitive and is prone to make the algorithm unstable.

One approach which may alleviate the need for such control logic would be to ensure

that the initial distribution for the SFD method is close to the desired final sampling density. Indeed, in their original method, Persson and Strang [90] use a probabalistic rejection, applied to a semi-dense structured grid. In doing so, the initial distribution is representative of the desired density, yet the distribution quality is very poor, as per Figure 5.14. However, the key restriction here is that the distribution quality remains such that control logic is still desirable to accelerate convergence. In other words, the distribution still requires significant work to achieve the final state.

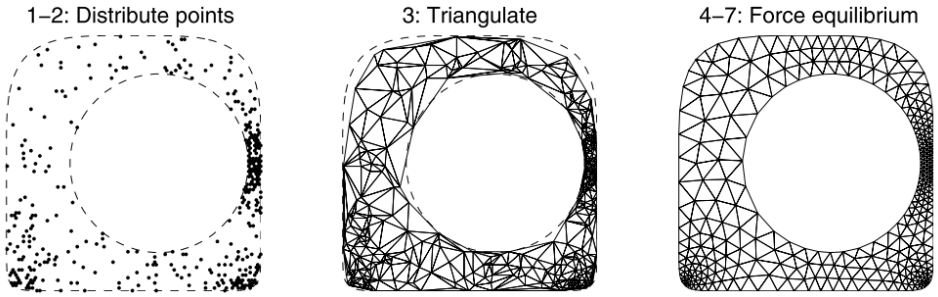


Figure 5.14: Example of the distribution following the rejection method of Persson. Image from [90].

Alternatively, the distribution resulting from the AIS method may be used as input. As has been argued, the quality is already sufficient for PIV, and therefore would not require the additional control logic, greatly simplifying the SFD method. In this sense, the SFD method acts like a mesh smoothing operator, yet, unlike Laplacian smoothing, this operator takes into account the desired distribution density. While this can improve the distribution quality slightly, and with only a few iterations, the relative improvements in solution accuracy are small, as has been demonstrated within this chapter. As such, the additional implementation and computational complexity makes a hybrid method undesirable when compared to AIS by itself.

5.5.6 Comparison to structured

Although the intention of this chapter is to compare unstructured distribution methods, it is worth briefly comparing AIS against structured to demonstrate the potential benefits. To do this, an instantaneous image from the backwards facing step flow is considered, paying particular attention to the displacement gradient as this will amplify the differences in the methods.

To restrict the comparison to only vector locations as much as possible, the WS for both the structured and unstructured analysis are kept the same, reducing from 97px^2

to 25px^2 over 3 iterations, with a subsequent refinement iteration also at 25px^2. Both 50% and 75% overlap are shown for the structured, resulting in ≈5,000 and ≈20,000 final windows, respectively, whereas the final number for the unstructured analysis was chosen to be 2,000. The motivation for this choice is so that the run time is approximately equivalent to the structured analysis with 50% overlap. The run time for the 75% overlap case was ≈3.5x greater.

The unstructured analysis used the spatial standard deviation as an objective function, as per [59]. The resulting unstructured solution is shown in Figure 5.15.

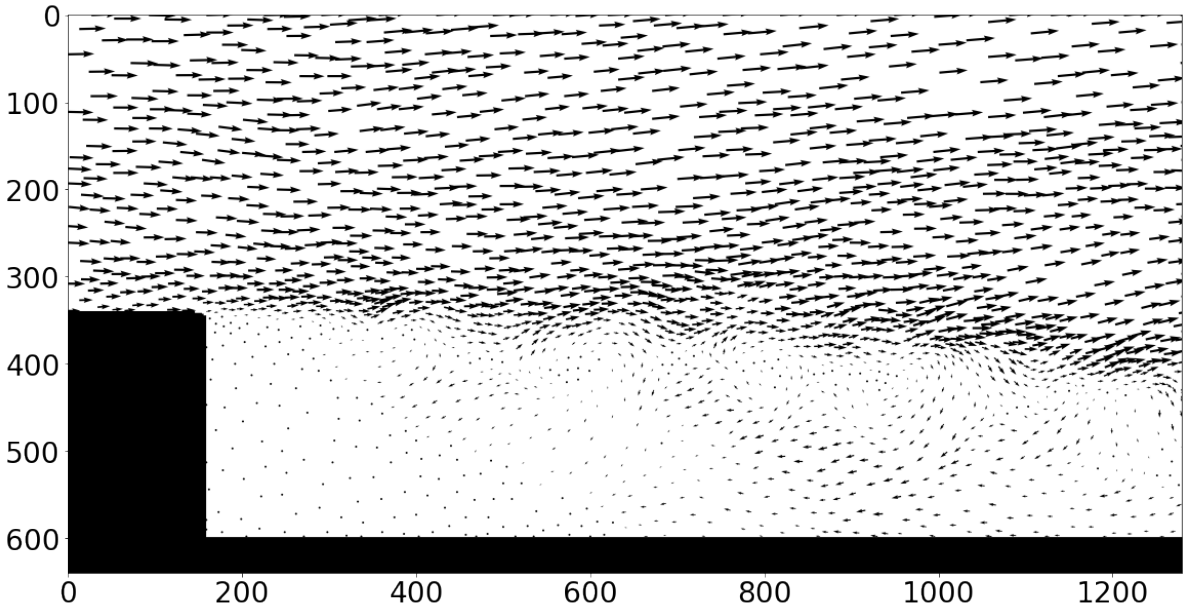


Figure 5.15: Final unstructured displacement field.

Figure 5.16 compares a contour plot of u$d\,y$ for each case, as this is the dominant gradient in the shear layer. Visually, it can be seen that AIS solution is more comparable to the 75% overlap case, which represents better sampling, particularly along the shear layer and in regions of more turbulence. Moreover, this similarity persists even in the freestream despite the lower sampling rate chosen by the algorithm.

To further demonstrate this, slices through u$d\,y$ have been taken at various points along the flow for the three cases, shown in Figure 5.17. While all three cases remain largely similar, there are subtle differences in some regions, such as the shear layer and, to a lesser extent, turbulent regions further downstream. The similarity of the methods shows that, in this case, sampling with 50% overlap is mostly sufficient to capture the flow features properly.

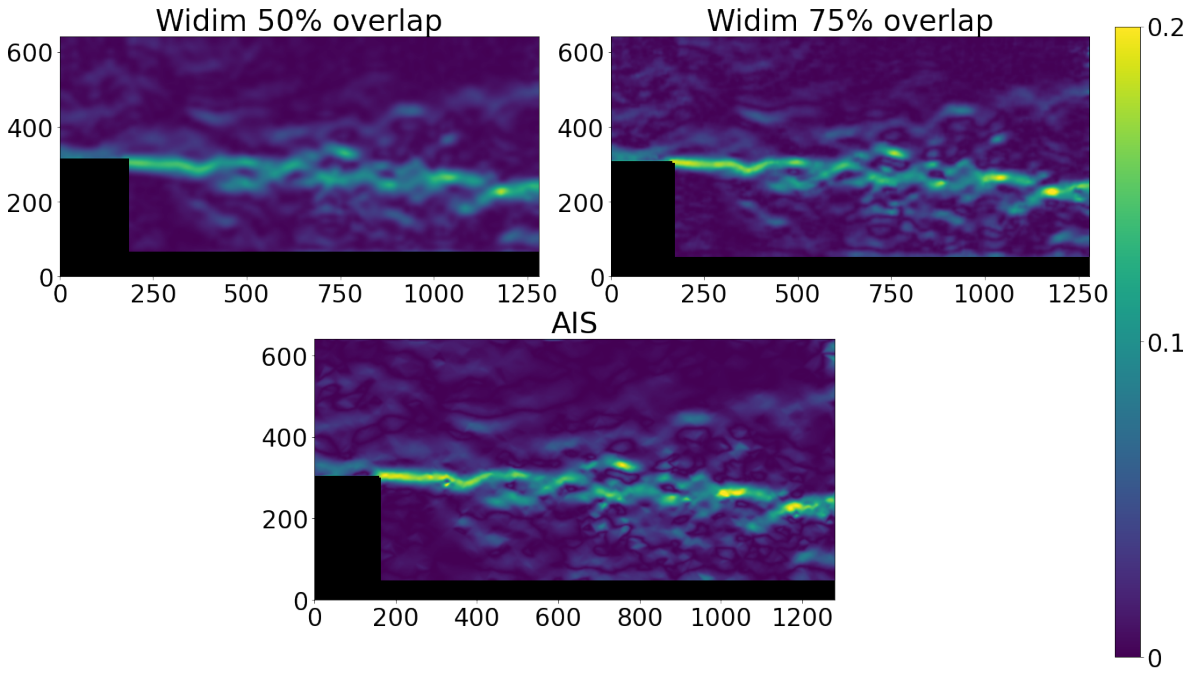


Figure 5.16: Comparison of udy for structured and unstructured analysis.

To highlight the differences in the shear layer, a zoom is provided in Figure 5.18. This elarged region shows a better capturing of the shear layer's strong gradients due to the greater sampling rate, without loss of fidelity in the smoother freestream region.

These results show that the AIS method can be used as a robust distribution method for fully unstructured adaptive sampling. Solution fidelity on par with a structured analysis with 75% overlap ratio is attained with the computational cost comparable to a 50% overlap analysis. Furthermore, the benefits of adaptive window sizing, more easily afforded by the adaptive method, are not present here and should be considered when comparing the relatively subtle differences presented above. With a more optimal objective function, that could more intelligently weight the importance of various regions, and with further optimisations to the implementation[10] this cost could be further reduced, while maintaining solution accuracy.

5.6 Summary

Fully unstructured adaptive PIV algorithms are capable of locally adjusting the sampling locations and parameters according to the image and flow conditions. In doing so, it is

[10]The adaptive approach is somewhat disadvantaged here as the implementation has received less code optimisation than the regular structured approach

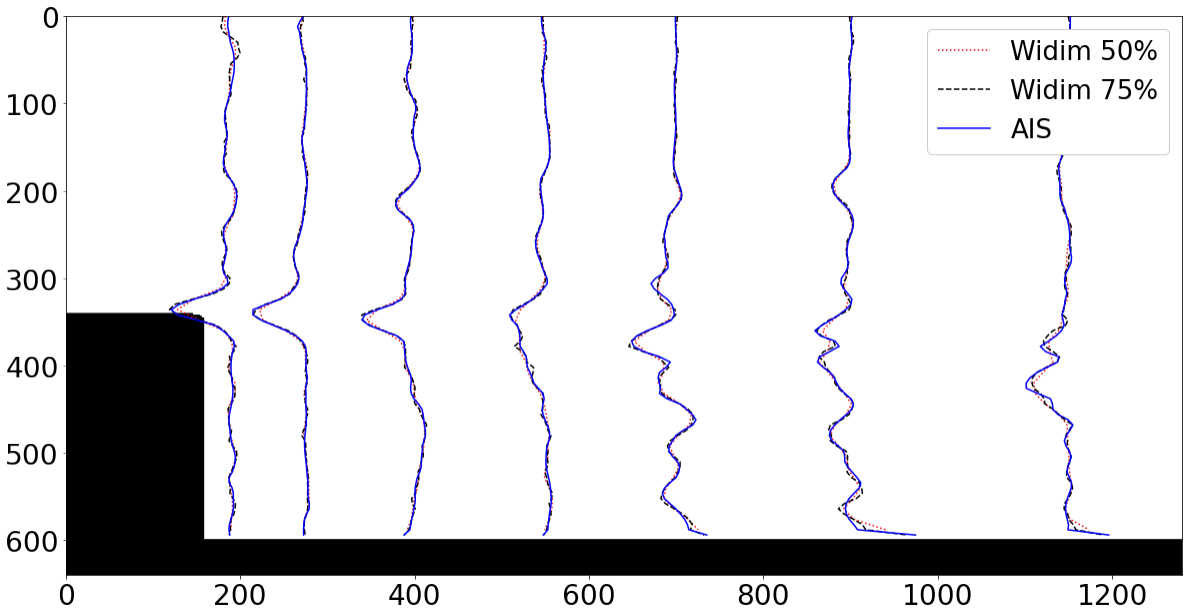


Figure 5.17: Displacement gradient u$d$$y$ along various slices throughout the flow.

possible that they are able to be more efficient, optimal, and automatic - in line with the motivations of this research. However, in adopting a fully unstructured approach, a wide range of algorithmic modifications are required, each of which being required to efficient, optimal, and automatic. Onesuch key aspect, which has been explored in this chapter, is the need to actually create the unstructured distribution of samples according to some objective function.

When used for adaptive sampling PIV, an unstructured distribution method must be fast and robust, while capable of producing sampling distributions faithful to some imposed objective function, without compromising interpolation reconstruction, or hindering vector validation routines. The need for computational efficiency should not be understated as an *optional* motivation; if an adaptive approach is not efficient enough, then it may no longer represent the best approach to analyse the image, c.f. Section 4.3, Figure 4.2.

There are two notable examples in the literature, however, the first, the PDF transform method, leads to excessive clustering of samples. This is shown to lead to poor interpolation accuracy, yet more crucially, to poor vector validation performance. Such degradation in performance is especially critical in adaptive methods, where this may mislead subsequent adaptive methods.

The second method, the SFD methods produces high-quality distributions, yet, is computationally intensive and unstable. In its original form, there is a lack of clarity

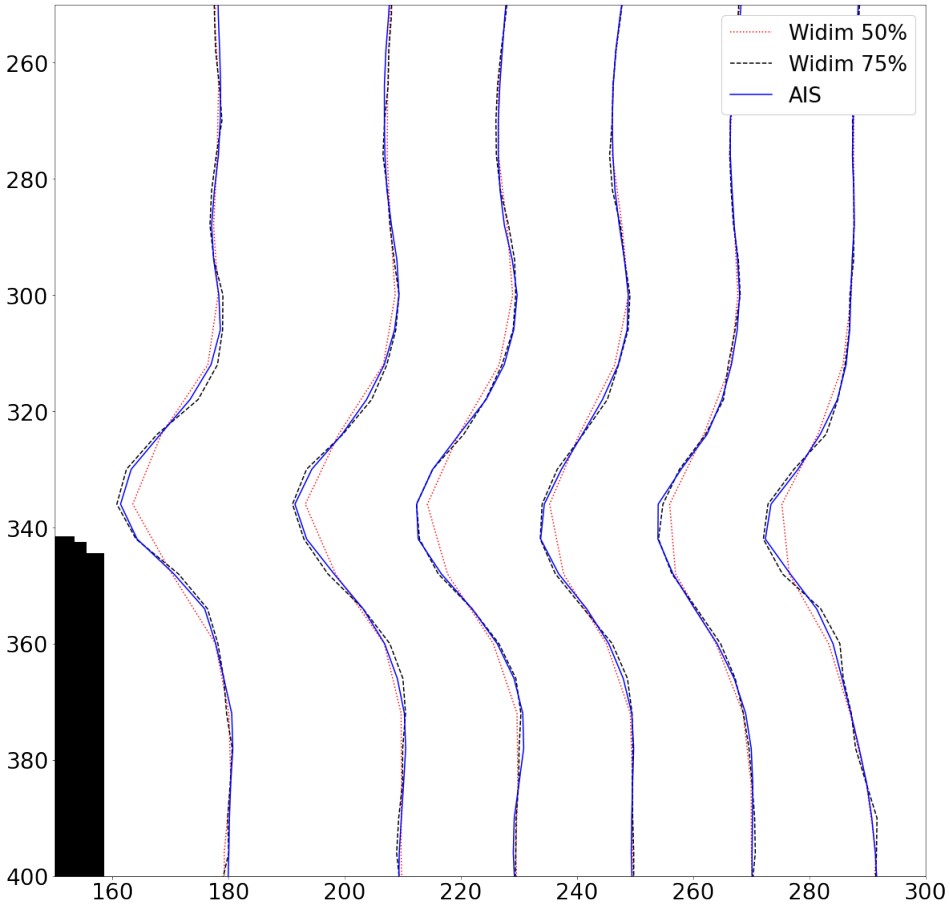


Figure 5.18: Displacement gradient udy along slices through the shear layer.

on suitable parameter settings. Work conducted in this research, presented in Section 5.3, has provided some clarification and improvement on this matter, yet the method remains computationally unviable. Due to the high quality distributions, however, the method remains a good benchmark.

A novel method, called adaptive incremental stippling (AIS), is presented in this chapter, and compared against the benchmark set by the optimised SFD approach. The new distribution approach is extended from a method in computer graphics for creating stippled images, and has been developed in these works to be suitable for sample distribution. The method is efficient, boasting linear time, and produces distributions with quality close to that of the SFD method.

The method was compared against a structured implementation and showed resolution similar to the higher resolution (75% overlap) solution, yet with run time comparable to the lower resolution (50% overlap) solution. With further work to the objective function, so that regions of interest can be more effectively identified and

weighted, along with automatic determination of how many windows to use, and use of adaptive window sizing, the adaptive approach provides a method to automatically analyse PIV images efficiently and accurately, without the user needing to make tradeoffs between accuracy and computational cost.

If such improvements around the wider adaptive approach can be realised, then fully unstructured adaptive sampling, by itself, is likely to represent a more desirable sampling method than fully structured, particularly as image size, and the importance of computational efficiency, increase. Yet, these unstructured sample distribution methods all require randomness to be efficient. Such randomness is essential to obtaining the necessary computational performance to ensure PIV adaptive sampling algorithms remain superior to their structured counterparts, however, adds artificial uncertainty to the solution and hinders algorithm development as a result.

A fully unstructured adaptive routine can still improve the analysis performance of PIV images, however, it is the author's belief that this remains only the case for images with significant spatial variations in image quality or flow scales. Conversely, if some structure can be preserved in an adaptive approach, then efficient data structures may be utilised and deterministic solutions may be obtained - allowing for maximum information extraction, without the associated challenges of unstructured data and distribution methods. The following chapter, therefore, investigates the prospect of a semi-structured AS algorithm.

CHAPTER 6

CUT-CELL APPROACH

It has been shown that fully unstructured adaptive sampling routines can improve the efficiency, accuracy, and automation of PIV image analysis, albeit with a significant amount of algorithmic modifications to facilitate such unstructured approaches. Although improvements are being made in these domains, for example, the novel high quality efficient sample distribution method as presented in the previous chapter, which lie at the heart of adaptive sampling (AS) methods, there are still many unanswered questions. More importantly, all the explored options for sample distribution rely on some element of randomness to attain the required computational efficiency. Of itself, this is undesirable, as it adds uncertainty, however, more importantly it hinders the future development of algorithms. A deterministic, semi-structured, approach is therefore considered in this chapter, which capitalises on the efficiency savings that can be afforded to structured methods. While this limits the flexibility of sampling density variation relative to a fully unstructured technique, this still represents a vast improvement on regular structured grids.

6.1 Introduction

Due to their prevalence among many aspects of engineering and science, the calculation of the Fast Fourier Transform has received significant attention over the past decade and a half resulting in a significant reduction in the physical time required to perform a single cross-correlation operation. This relative cheapness of a single cross-correlation requires

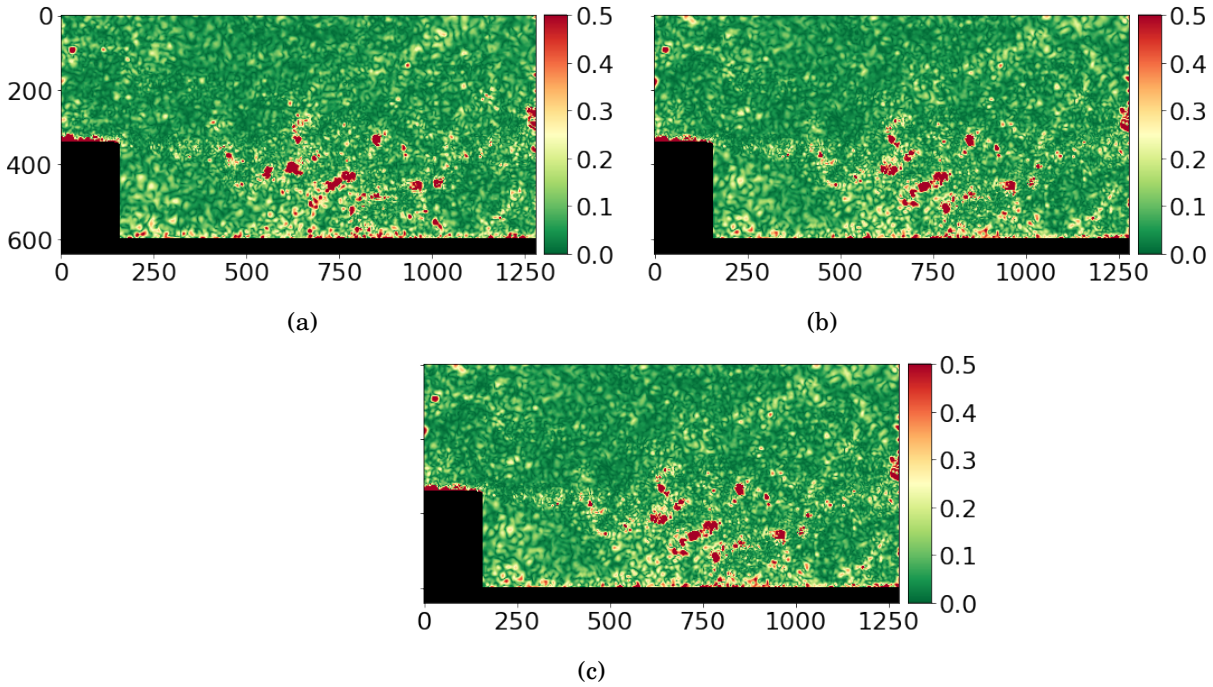


Figure 6.1: Magnitude of vector difference between three analyses of the same PIV image pair, using adaptive sampling with the same sampling parameters. The (approximate) number of windows, distributed using the AIS method according to the local displacement standard deviation, increases from 1,000 to 10,000 over three iterations. A fourth refinement iteration is performed with the same sampling distribution. Showing the difference between runs (a) one and two, (b) one and three, and (c) two and three.

that more windows be avoided by the adaptive approach, such that when coupled with the increases in distribution, interpolation, and validation costs, the overall run-time is still reduced. Additionally, the vast rise in available compute power, even for desktop PCs, further reduces the attractiveness of spatially adaptive techniques relative to structured, since jobs can be left running as a background task, or sent off to a remote cluster, and allow the user to continue unimpeded with other tasks. As such, both the literal and perceived costs of computation are reduced, in turn reducing the perceived benefits of an adaptive sampling approach yet without mitigating the drawbacks, namely algorithmic complexity.

More critical than the additional overhead of implementation is the stochastic nature of existing adaptive sampling methodologies. All three of the previously introduced distribution methods, namely the probability density function (PDF) transform method [59, 103], the spring-force distribution (SFD) method [77, 90], or the adaptive incremental stippling (AIS) method [60, 109], depend on random behaviour. Random number

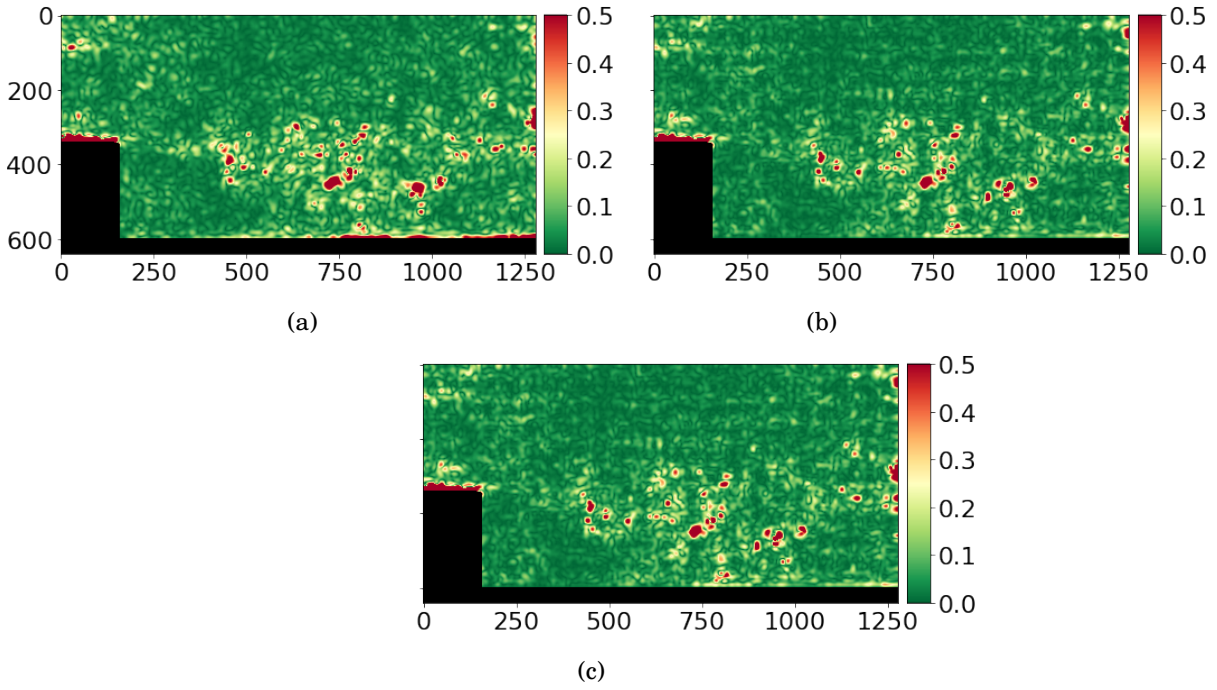


Figure 6.2: Magnitude of vector difference between three analyses of the same PIV image pair, using WIDIM with slightly different paramter settings. The baseline solution uses a WS profile of 97px → 49px → 25px → 25px, using a window overlap ratio of 50%. A second analysis was performed wherein the WS profile was instead 97px → 49px → 23px → 23px, still with 50% overlap. A third analysis was performed using a WS profile equal to the first, 97px → 49px → 25px → 25px, yet with 60% overlap. Showing the difference between analyses (a) one and two, (a) one and three, and (a) two and three.

generator seeds may be fixed to allow reproduction of results, yet it takes only a small divergence of parameters to result in an entirely different sampling distribution, which may or may not result in a different solution. For example, although the use of the non-linear median operator in vector validation adds robustness in the presence of outliers (See Section 2.11), it also allows for a relatively small difference in displacement value, by virtue of sampling a region of the image offset by a few pixels, to have disproportionately large effects on the solution. Figure 6.1 shows the magnitude of such variations for three analyses of the same image with the same set of parameters.

In and of itself, this does not necessarily prohibit the effective use of such implementations, since iterative methods are generally effective at correcting small discrepancies and gradually resolving the residual towards the final solution. While these differences may seem significant, they are comparable to the fluctuations observed for minor changes to a WIDIM analysis, such as changing the final WS by two pixels or

increasing the overlap ratio by a small amount, as shown in Figure 6.2. Although these differences may be comparable to minor parameter differences in a structured approach, the key distinction here is that differences for the adaptive approach occur for the *exact same* analysis parameters.

The implications that different analyses of the same image, with the same settings, can result in different solutions are twofold. Firstly, as with the ideology that the user's opinions, i.e. views on the trade-off of resolution vs robustness, should ideally not influence the solution, the solution should equally not depend on random chance. Secondly, for the development of adaptive sampling techniques, it is challenging to investigate the performance of novel approaches or perform parametric studies, since solution differences from random fluctuations must now be distinguished from differences inherent to the changed configuration. In fact, such distinction is practically impossible in many cases, due to the complexity of the distribution and the inability to parametrise the problem.

For all the aforementioned reasons, fully unstructured adaptive sampling techniques appear to only be of clear and significant benefit when analysing images with a significant range of flow scales. Under these circumstances, the reduction of the number of windows in regions with large flow scales is sufficient to outweigh the additional costs of adaptivity and other less quantifiable drawbacks. Yet the relatively mediocre improvements otherwise are unlikely to sway a user to adopt such stochastic techniques in favour of the less computationally efficient, structured, counterpart.

As has been identified in the previous chapters, more flexibility is desired in PIV image analysis algorithms. This flexibility can allow for better parameter adjustment, adaptive or user-controlled, to improve the amount of information that can be extracted from PIV images. Chapter 4 demonstrated that to achieve this flexibility many other aspects of the algorithm must be adjusted to accomodate the new approach. Developing on this, Chapter 5 investigated a fully unstructured approach, presenting a new distribution method, and found that, while the method works, there are some significant challenges that are yet to be overcome. Most notably of these, is the stochastic nature of the distributions as described above.

One method that may solve many of these challenges, is one that uses a semi-structured sampling approach, composed of a grid of cells wherein a window is placed at each vertex. Cells may then be split, creating four new cells and five new correlation windows, to increase the sampling density in specific regions. By maintaining neighbour information in a prescribed manner, many of the additional computational costs incurred

by a fully unstructured methodology may be avoided and certain data structures may be taken advantage of, notably, quad-trees. Moreover, the sampling locations can be obtained in a deterministic fashion, allowing for better investigation of the cause-and-effect of various parameters and methodologies and easier comparison to structured approaches.

Utilising a semi-structured implementation, can be done in two ways. One option is to use this as a distribution technique, analogously to the AIS or SFD distribution methods, within existing adaptive sampling algorithms. In this manner, it can be used to create a sampling distribution that reflects some objective density function, such as the local standard deviation of displacements. Alternatively, such a sampling grid lends itself to a refinement-based adaptive sampling approach, wherein cells are split only where the solution is deemed unsatisfactory by some criterion. An additional benefit of a refinement-based sampling approach, rather than an objective-based approach, is that the refinement function can be more easily used as a stopping criterion. In this manner, the challenge of determining the optimal number of windows is also eased; the number is automatically decided based upon the refinement criteria.

This chapter explores the feasibility of a semi-structured sampling approach, which claims to resolve many of the unanswered challenges. It is still required to obtain a dense predictor in order to deform the image. Therefore, an interpolation scheme which can take advantage of the semi-structured location information is required, and is presented first. Next, the multigrid architecture is introduced, showing how neighbour information can be efficiently maintained by utilising well-established efficient data-structures and enforcing just two simple rules. Application as an objective-based distribution method, akin to AIS or SFD, is shown, demonstrating that pseudo-arbitrary density distributions can be obtained more easily and quickly than other methods. Finally, the completely new approach to adaptive sampling is presented, wherein sampling density is adjusted on the basis of a refinement criterion. This refinement criterion is based upon the difference between a cubic and linear interpolation of the same control points and acts as a heuristic for the uncertainty of the current solution.

6.2 Multi-level interpolation

When using unstructured interpolation schemes, e.g. CloughTocher, there is an additional cost associated with *constructing* the interpolating function. The cause of this additional cost is often due to constructing a Delaunay triangulation as well as increased complexity

in determining the appropriate control point weights. Additionally, *evaluating* the interpolation also requires significantly more computation, since for each evaluation point, the simplex to which it belongs must first be identified. Conversely, when evaluating a structured interpolant, the local weights which govern the interpolated values can be identified trivially, just from the evaluation points location. While elaborate algorithms may be constructed which ease the task of identifying the local simplex, the process is inherently slower.

In the context of PIV, evaluation performance is as critical as construction performance, since the displacement must be interpolated onto a pixelwise grid each iteration. Considering that images with one million pixels are nowadays considered to be small, the evaluation can not be neglected. A relatively simple solution is to use a two-step interpolation routine (see Section 4.4.4), wherein the unstructured data is first interpolated to a coarse structured grid, which can subsequently be interpolated onto the pixelwise grid through structured interpolation.

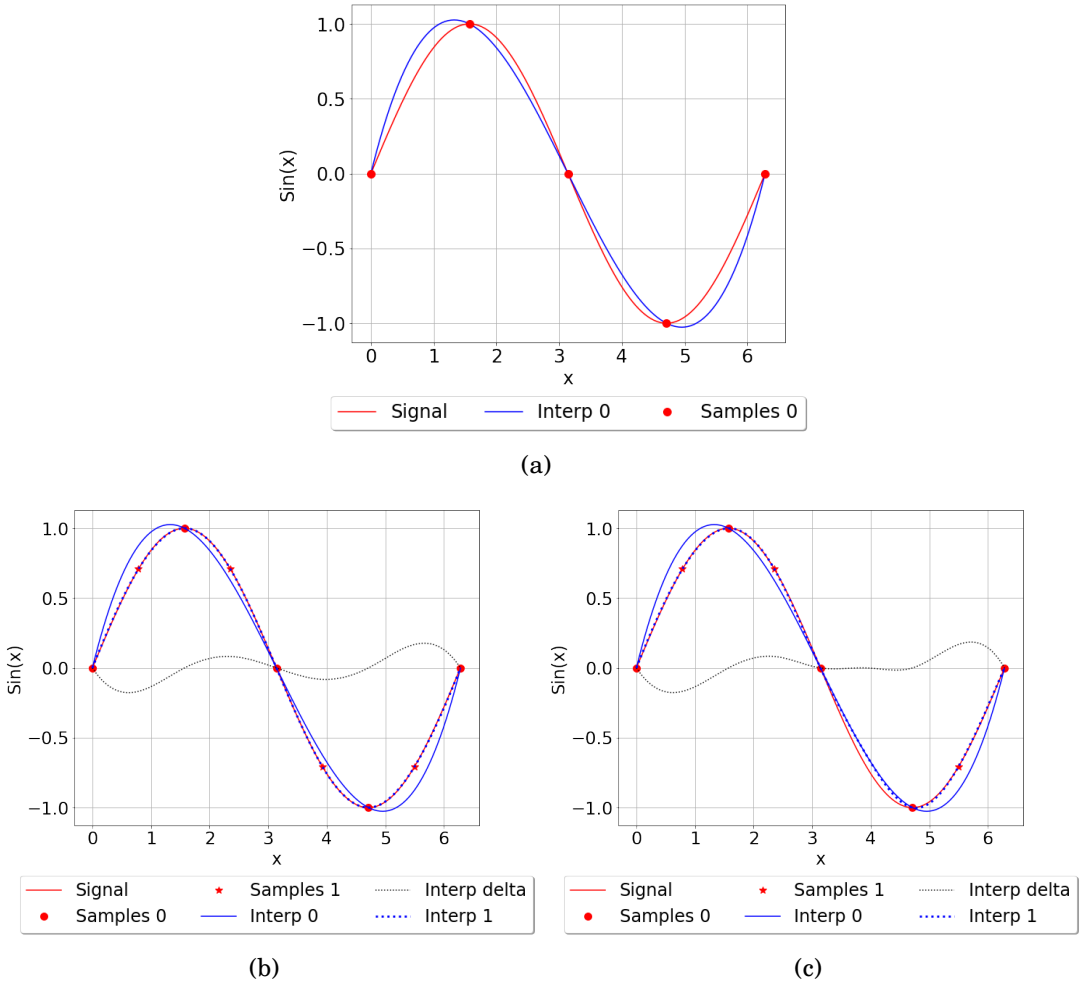
In doing so, the computational requirements can be greatly reduced, but not without limitations. Firstly, exact recovery of control points is no longer guaranteed due to the secondary interpolation. Secondly, the intermediate grid spacing must be less than the smallest intersample spacing, otherwise a low-pass filtering effect will be observed.

Alternatively, a semi-structured interpolation scheme allows the sampling structure to be relaxed, making sampling more flexible, while still allowing evaluation to be performed efficiently. First, the coarsest grid of datapoints, which must be regular and full, is interpolated. At the grid points of the tier above, with precisely half the spacing, the residual to the interpolation is calculated, being zero by definition for any grid point in the coarse mesh and any location in the finer mesh without a data point. The residual is then interpolated and summed with the first interpolation, and the process repeated until all samples are interpolated.

Algebraicly, let $f_0(x)$ be the function which interpolates sampled values in the coarse grid, $y(x_0)$, where the samples are spaced by a uniform distance, h_0, with exact recovery of values such that $f_0(x_0) = y(x_0)$. Given an additional set of sampling locations, x_1, with spacing $h_0/2$, the residual at each of these locations can be calculated as (6.1)

$$\Delta y_1(x_1) = \begin{cases} y(x_1) - f_0(x_1) & x_1 \neq x_0,\ \exists\ y(x_1) \\ 0 & x_1 = x_0,\ \nexists\ y(x_1) \end{cases} \tag{6.1}$$

Additionally, a sampled value, $y(x_1)$, is not required for all locations in x_1; in the absense of a sampled value, $\Delta y_1(x_1) = 0$. From this, another interpolating function,



(a)

(b)

(c)

Figure 6.3: Multi-level interpolation in one dimension for a sinusoidal signal. (a) Signal sampled at five equispaced locations and cubic spline interpolation. (b) Signal additionally sampled with four equispaced intermediate locations and the interpolated delta, which sums to produce the overall interpolation. (c) As previously, yet with a sample location ommitted to highlight how the scheme performs with non-unifom sampling locations.

$\Delta f_1(x)$, is then created, again using a cubic spline, which interpolates the values of Δy_1 over the domain. The total interpolant can then be constructed as a summation of the base interpolation, $f_0(x)$ and residual interpolation, $\Delta f_1(x)$.

$$f(x) = f_0(x) + \Delta f_1(x) \tag{6.2}$$

The process can be repeated for as many times as required to obtain the total interpolating function, which considers all points.

Figure 6.3 demonstrates a one dimensional example, without loss of generality. A sinusoidal signal is sampled at five equispaced locations and interpolated using a cubic spline, as shown in Figure 6.3(a). Now consider that, due to grid refinements, additional samples are added at the midpoints of the five original locations, as in Figure 6.3(b). The difference between the sampled values and the interpolated value, shown by the black dotted line, then combines with the base interpolant to obtain the overall function. Finally, Figure 6.3(c) shows the case where one of the samples in the finer grid is missing. It can be seen that the total resulting interpolation remains very similar to the original interpolating line, i.e. $f_0(x)$, labelled as 'Interp 0', within this region, with minor fluctuations being observed due to continuity preservation.

For a final grid spacing of h_k, and $h_{k-i} = 0.5^i \cdot h_k$, the total number of evaluations is $N^2 + (N/h_k)^2 + (N/h_{k-1})^2 + (N/h_{k-2})^2 + \cdots + (N/h_{k-k})^2$. Therefore, for even moderate h_k, the number of additional evaluations, beyond the minimum N^2, becomes small and thus performance is not much slower traditional interpolation of a full structured grid, yet is able to handle semi-structured data.

6.3 Multigrid Architecture

6.3.1 Multigrid structure

The initial grid of sample locations is created using a regular spacing in both x and y, wherein each grid location represents a vertex of a grid cell. To restrict refined grids to integer pixel locations, the initial spacing should be some power-of-two multiple of the smallest anticipated grid spacing. Additionally, to facilitate refinement of the multigrid near the boundaries of the image, the grid should extend one spacing beyond the domain of the image, as shown by Figure 6.4. These externel samples will not be correlated or considered in any stage of PIV analysis, i.e. interpolation or vector validation, however, the resulting cells, often referred to as halo cells, behave identically to any regular cell.

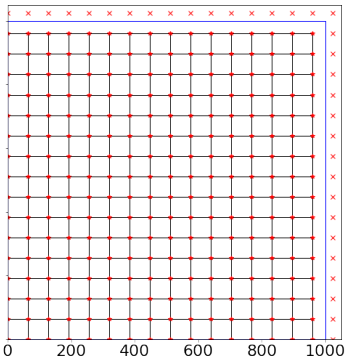


Figure 6.4: Extension of the grid beyong the image domain to facilitate boundary refinement. The blue line indicates the perimeter of the image. Red dots indicate valid correlation windows within the domain, whereas red crosses indicate halo-cell vertices.

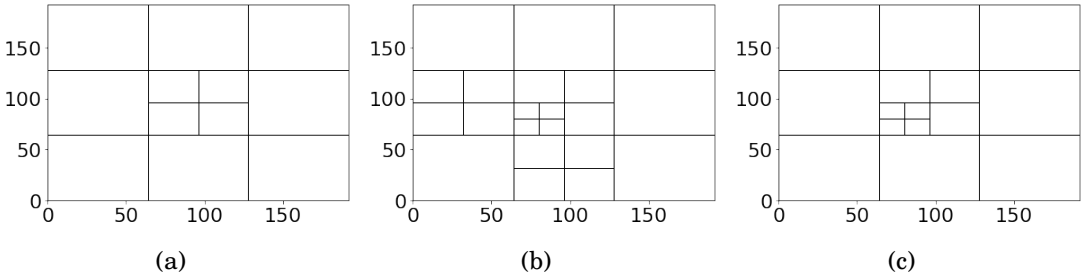


Figure 6.5: Demonstration of allowable cell configurations, (a) and (b), and an illegal configuration, (c).

Interrogation windows, and associated properties, e.g. displacement values, are stored in a regular grid structure such that gridded data can easily be obtained where necesssary. Throughout the life of the multigrid, it is important to keep track of neighbour information. During initialisation, it is straightforward to associate the relavent neighbours for each cell, noting that a null value is stored where no neighbour exists.

Refining, or splitting, a cell relies upon two fundamental principles.

- Neighbouring cells are not permitted to be more than one refinement tier apart, as illustrated by Figure 6.5.
- To qualify as a neighbour, the cell must belong to the same tier.

Consider Figure 6.6 wherein cells are numbered to aid the discussion, note that cell 3 still exists beneath cells 4-7 but has been omitted for clarity. Cells 5 and 6 are neighbours of 4, whereas cell 2 is *not* a neighbour. Instead, the western neighbour of 4 is null. Likewise, neither 4 nor 6 are eastern neighbours of 2 - which remains to be cell 3.

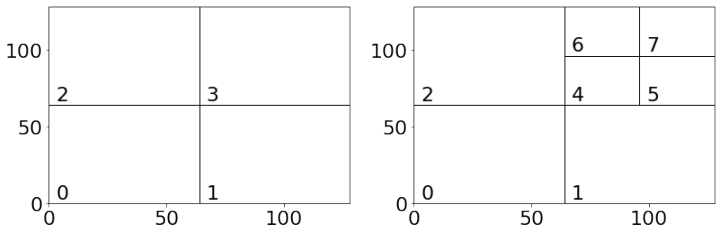


Figure 6.6: Visual aid for cell refinement algorithm.

Storing neighbours in this manner allows the first criteria to be easily enforced, without the need for handling arbitrary quanities of neighbours. To begin, the algorithm for splitting a cell is presented in Algorithm 1, and discussed herein, using cell 3 as in Figure 6.6(a) as an example. The first step is to create the five new correlation windows which will represent the vertices of the new grid cells, or, in the case that a neighbouring cell has already been split, identify such windows. In the current example, cell 3 is the first cell in the current max tier, tier 0, to be split, and therefore a new data grid must be created to maintain windows which exist in tier 1. The four new cells, 4-7, see Figure 6.6(b), may now be created, each containing references to the four vertices which make the cell and a reference to the parent cell, cell 3. Additionally, cell 3 contains a reference to each of these four new cells.

The cell refinement process is finished by updating neighbour information for each cell. Each cell has two known neighbours which can be trivially identified, e.g. cells 6 and 5 are the north and east neighbours of cell 4, respectively. Additional neighbours can be identified by checking whether the parent's (3) neighbour (1 or 2) has children. If the parent's neighbour has children, the references to the appropriate children can be copied as neighbours. An example is given in lines 6-11 in Algorithm 1. By ensuring that adjacent cells are never permitted to be more than one tier different, only the immediate parent and its neighbours need to be searched for neighbours, preventing the search domain growing significantly with increasing number of tiers.

To complete the algorithm, a check must be made to ensure that the first principle, i.e the maximum tier difference between neighbouring cells, is not violated. If the cell in question has neighbours in all directions, or exists in tier 0, then it may be safely split according to Algorithm 1. Alternatively, if the cell's parent has neighbours in a given direction, these too must be split. Using cell 5 in Figure 6.6 for reference, the four neighbours are queried. To the north and west, the presence of cells 7 and 4, respectively,

Algorithm 1 Cell splitting algorithm

```
Identify, if neighbours are refined, or create new correlation windows at vertices.
Define four Cells, bl, br, tl, tr, using new correlation windows.
Define known neighbours
bl.North ← tl
bl.East ← br

Search for neighbours
if self.West.has_children then
    bl.West ← self.West.br
    self.West.br.East ← bl
else
    bl.West ← NULL
Repeat neighbour search in other three directions
```

indicate that refining the current cell would be acceptable, with respect to these two cells. On the other hand, null values to the south and east suggest that refining cell 5 *may* violate this principle, and thus adjacent cells in lower tiers may require splitting. The parent's neighbour to the east is null, which indicates that this cell is at the edge of the domain and neighbours are not required. However, the parent's neighbour to the south, cell 1, must be refined before cell 5 may be refined. This process is allowed to recurse as much as required, splitting cells in lower tiers before cells in higher tiers can be split.

6.3.2 Boundary interface refinement

The ability to place samples arbitrarily close to an interface or image boundary, is a desriable characteristic of fully unstructured distribution methodologies. Due to the equispaced nature of the initial structured distribution, alignment of the grid with such interfaces is not straightforward, or necessarily possible for arbitrary mask shapes. Figure 6.7(a) demonstrates the lack of grid-mask alignment, for the flow around two uneven structures representing urban buildings, for an initial grid spacing of 64px.

Despite this, it is possible to pre-process the multigrid such that resulting semi-structured grid contains nodes arbitrarily close to various interfaces. Leaf cells, i.e. those which do not contain any children, which contain any masked pixels should be refined, and likewise halo cells, which contain nodes outside of the domain, should also be split. Figure 6.7 shows the gradual refinement of the sampling grid near the interfaces. Note that this pre-processing step simply serves to reduce the distance from the mask interface to the nearest sample, and does *not* represent grid refinement near these interfaces. As

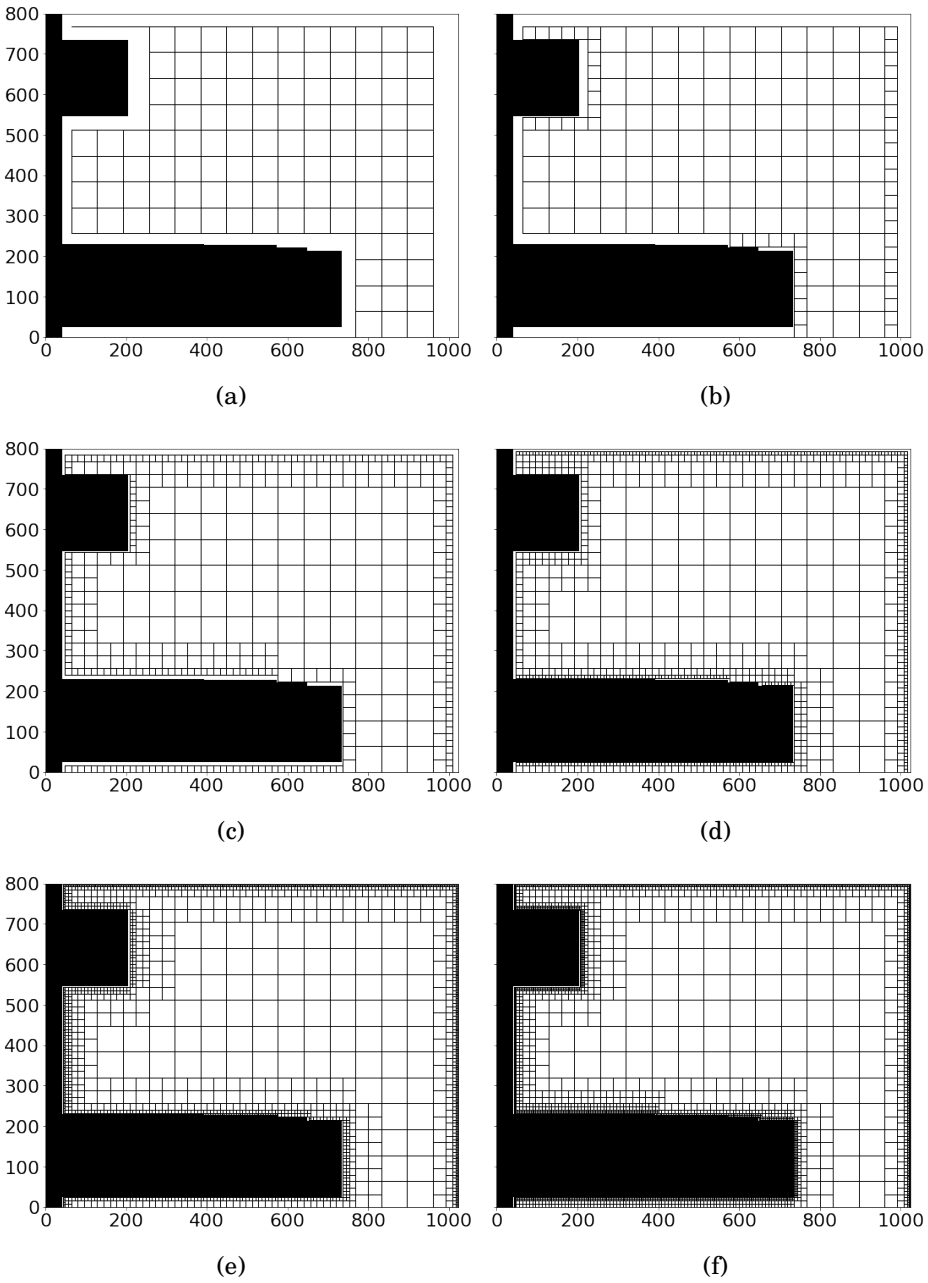


Figure 6.7

can be seen in the lower right corner of the domain, cells wherein the vertices coincide with the limit of the domain are not refined.

6.3.3 Arbitrary sampling distributions

It is possible to use the multigrid sampling methodology to create psuedo-arbitrary distributions of samples, much like the fully unstructured methods described in the previous chapter. The first step is to convert the objective density function into a distance function, as per Section 5.3.2. In the aforementioned section, the distance function was obtained such that a disk of diamter $h(x,y)$, centered on (x,y), with constant local objective density function, $\phi(x,y)$, would contain a pre-defined amount of the objective density function, F, as per (5.8).

$$\frac{\pi h^2(x,y)}{4}\phi(x,y) = F \qquad \text{(5.8 revisited)}$$

As the distributions in the current approach remain semi-structued, we can, instead of disks, use squares to obtain the objective function, simplifying $h(x,y)$ into

$$h(x,y) = \sqrt{\frac{F}{\phi(x,y)}} \tag{6.3}$$

Where F is found by dividing the cumulative sum of the objective density function, which should be unity by definition, by the desired number of points.

The process to obtain a desired density distribution is then remarkably simple. An initial grid is created with a spacing equal to the next power of two bigger than the largest value in $h(x,y)$, i.e as in (6.4), and the multigrid object defined as per Section 6.3.1.

$$h_0 = 2^{\beta} \qquad \text{with} \qquad \beta = \lceil \log_2(\max(\text{h})) \rceil \tag{6.4}$$

Subsequently, the average of $h(x,y)$, within every leaf cell[1] in the multigrid is evaluated, and compared to the current cell spacing. If the average desired spacing is smaller than some fraction, η, of the current spacing it should be split. Leaf cells are repeatedly refined until none require splitting, at which point the final distribution is obtained. While it may seem intuitive to set $\eta = 0.5$, this results in an underproduction of windows, arising from situations where the average desired spacing may be just slightly greater than half, and thus not refined. Conversely, setting η too high will result in too many windows, as cells get refined beyond what is necessary. A good balance has been found by settings $\eta = 0.65$.

Figure 6.8 compares the distributions from the multigrid technique (6.8(c)/(d)), with distributions from the AIS method (6.8(b)), presented in the previous chapter, for an

[1] A leaf cell is one which does not contain any children.

objective function (6.8(a)), obtained from Franke's function (5.3). The desired total number of samples was set to 14,000, with AIS resulting in 13,888 whereas the multigrid approach distribution contains 13,582 locations. Notably, the ability of the multigrid technique to represent a gradual change in sampling density is limited, due to the discretised options of inter-sample spacings.

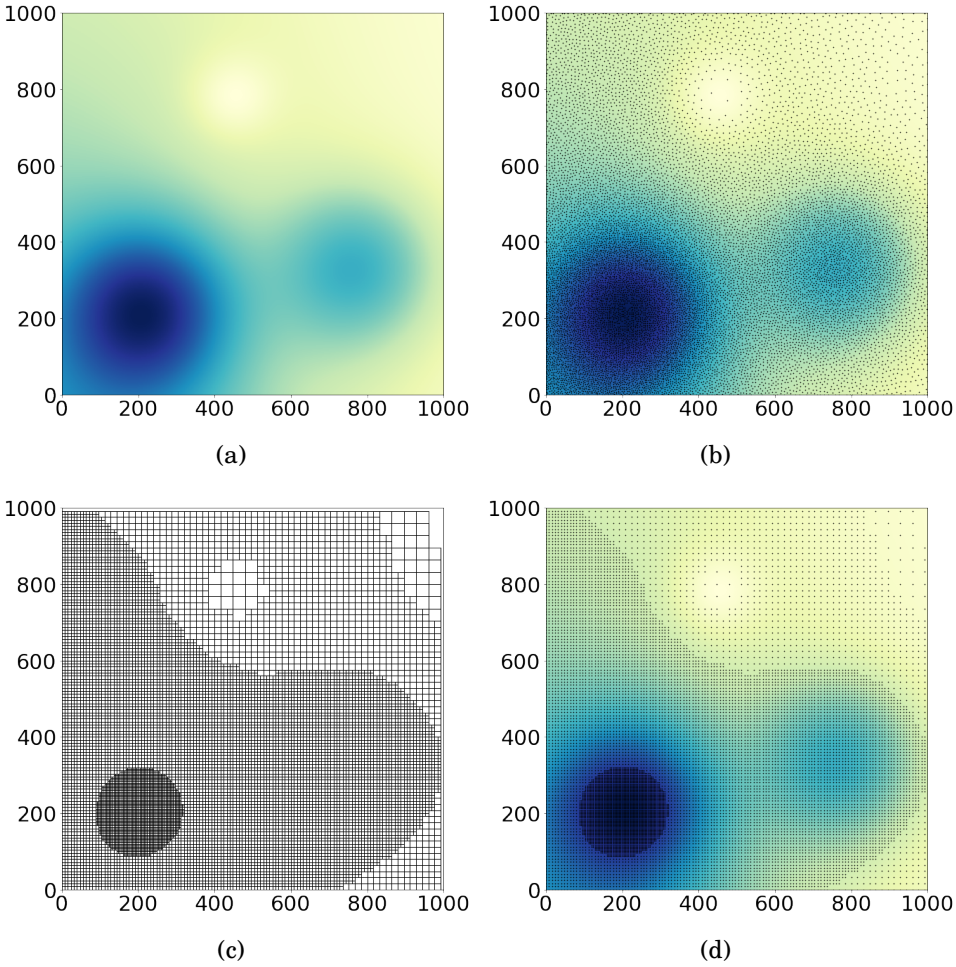


Figure 6.8: Comparison of semi- and un-structured distribution methods for an imposed objective density function (a) obtained from Frankes function, as per (5.3). (b) samples distributed using adaptive incremental stippling (AIS). (c) and (d) show the cells and the sample locations, respectively, obtained from the multigrid method.

To assess how the distribution method would perform within a real adaptive implementation, a more representative objective density function is obtained. The flow over a backwards facing step, as introduced in the previous chapter, is analysed using an adaptive routine, utilising adaptive initial window sizing (AIW) (see Section 4.6), and distributing according to a combination of the seeding density and the local standard

deviation of the flow (see Section 4.4.1). The initial number of windows is 2,500, which increases to 5,000 for the second iteration, followed by a refinement iteration, also with 5,000 windows. Displacement information is interpolated onto a dense predictor using an unstructured CloughTocher cubic interpolation scheme. From the final displacement field, the local standard deviation of the flow is obtained, using 4.5, and used as the objective density function, shown in 6.9(a). A total of 14,000 windows are desired from both distribution methods, resulting in 14,305 and 14,889 locations from the AIS and multigrid technique, respectively. As before, the multigrid distribution is shown in Figures 6.8(c) and (d), whereas the distribution from the AIS method is shown in 6.8(b). Due to the greater variations in desired sampling density, the multigrid distribution method is relatively well able to reflect the imposed sampling density, however, there are a few notable regions, e.g. (950, 475), wherein the local peak in objective density function can not be discerned from the resulting distribution.

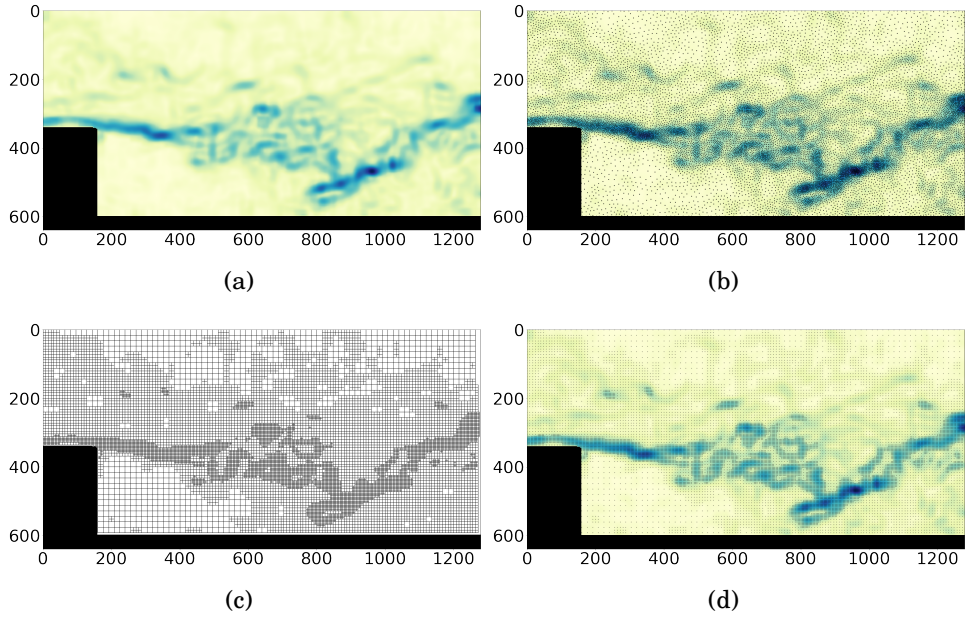


Figure 6.9: Comparison of semi- and un-structured distribution methods for an imposed objective density function (a) obtained from the local standard deviation of the displacement field of flow over a backwards facing step. (b) samples distributed using adaptive incremental stippling (AIS). (c) and (d) show the cells and the sample locations, respectively, obtained from the multigrid method.

Finally, it is worth mentioning the computational requirements of the multigrid technique relative to the AIS approach. Figure 6.10 shows the wall time, averaged over five repeats, required by each of the methods for a range of desired number of locations,

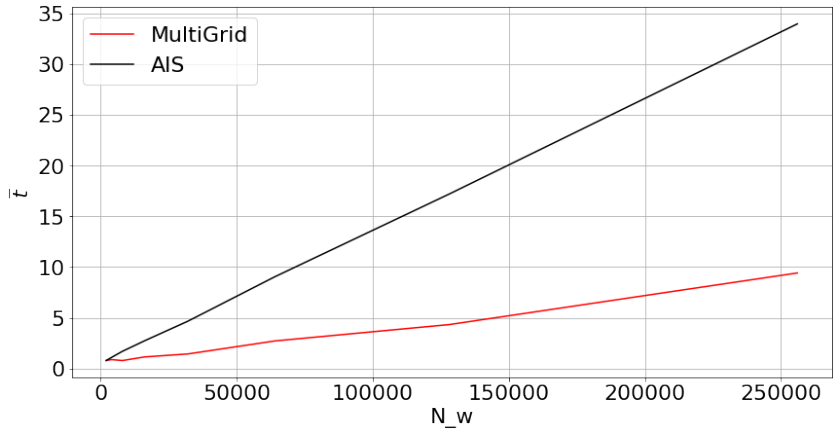


Figure 6.10: Run time comparison

N_w. While both are seen to exhibit linear cost with N_w, the gradient for the multigrid method is significantly shallower, taking approximately one quarter of the time relative to AIS.

While extensive investigation into the use of multigrids for semi-structured sample distribution remains to be conducted, preliminary signs are encouraging. The proposed multi-level interpolation, as per 6.2, allows for data collected at the resulting distribution locations to be more rapidly interpolated onto the dense predictor Furthermore, the distribution method also offers significant computational savings, due to the simplicity and structured nature of the algorithm, which could be directed towards additional image sampling. The distributions produced can be seen to reasonably reflect the input objective density function, however, it is noted that, due to the limited options available for inter-sample spacing, the ability to represent gradual changes in sampling density is diminished.

6.4 Refinement based sampling

Existing adaptive sampling methodologies create a fresh distribution for each iteration, based on the current understanding of the image and displacement field. The new distribution is used to interrogate the image and update the displacement field, and in turn, update the objective function. The updated objective function leads to an entirely different, although likely similar, distribution, and so on. As discussed in Section 4.4.3, the full re-distribution of available sampling budget is due to the fact that adding to an existing distribution imposes various challenges and constraints on the distribution -

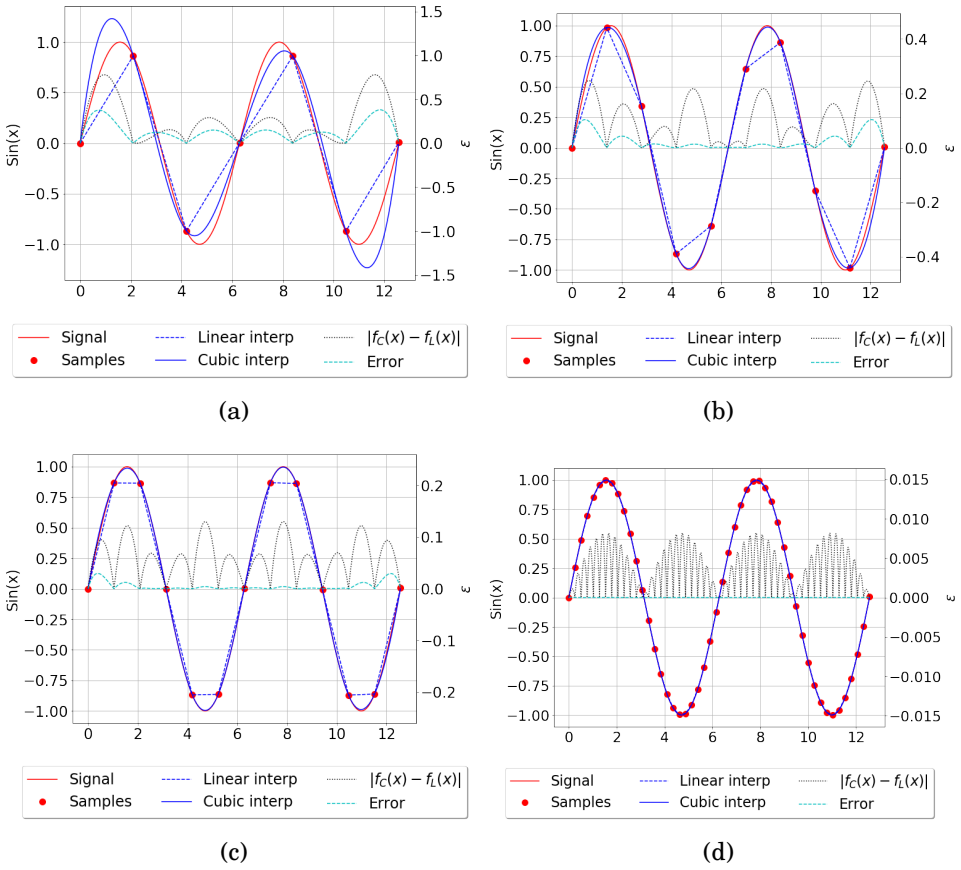


Figure 6.11: Reconstruction uncertainty approximated by the difference between a linear and cubic interpolant for (a) 7 samples, (b) 10 samples, (c) 13 samples, and (d) 50 samples. $f_C(x)$ is the cubic spline interpolation and $f_L(x)$ is the linear interpolation. The error is defined as $|f_C(x) - f(x)|$, where $f(x) = sin(x)$.

notably the ability to remain faithful to the imposed objective function and the ability to maintain distribution quality.

A multigrid architecture, however, does not face the same difficulties; when splitting a cell, new windows remain locally equidistant from each other, and by limiting the tier difference between adjacent cells, i.e. Figure 6.5, overall distribution quality is preserved. Furthermore, the ability of a multigrid structure to remain faithful to the imposed objective function was demonstrated in the previous section. Therefore, rather than requiring full re-distribution according to an updated objective function, a multigrid architecture facilitates the use of a refinement-based AS algorithm, wherein the local sampling density can be increased in regions that meet certain criteria, for example, some metric of local uncertainty or error estimate. By refining the grid in this way, the notable challenges associated with determining the appropriate number of windows (see

Section 4.4.2) can be avoided.

One option for a simple estimate of the local curvature is to compare the difference between a cubic interpolation and a linear interpolation, of the same sample data. Regions with low curvature, i.e. small second-order derivatives, will be well approximated by a linear interpolant, and therefore should be similar to a cubic interpolant in this region. On the other hand, regions with larger second derivatives will be poorly interpolated by a linear function, and so the interpolated value is likely to be significantly different from the cubic interpolant. This behaviour is demonstrated for a simple one dimensional sinusoid in Figure 6.11, where $|f_C(x) - f_L(x)|$ is the magnitude of the difference between a cubic spline interpolation, $f_C(x)$, and a linear interpolation, $f_L(x)$, with Figure 6.11(d) clearly showing the influence of curvature on this term. It is possible to show that the reconstruction error of an interpolating polynomial is proportional to the local curvature [104], and, therefore, the term $|f_C(x) - f_L(x)|$ effectively serves as a heuristic for the error, which can also be observed in Figure 6.11.

While other measures of uncertainty could be used, considering for example [116], the major advantage of this approach is simplicity, both in terms of algorithmic complexity and computational. An objective *refinement* function can, therefore, be constructed by computing simply a cubic and linear interpolation, without the need to determine derivatives, or solve complex systems of equations. Considering that the cubic interpolant is a necessary requirement for the algorithm anyway, since pixel-wise displacement information is required for image deformation, the only additional computation is, therefore, a linear interpolation. The threshold can then be easily be set by considering the amount of uncertainty in a single sample, e.g. 0.1px or less, as below this level the reconstruction error will be dominated by sampling noise.

6.4.1 Application to PIV images

As mentioned previously, the attractiveness of this approach is simplicity. This section will cover the application of this method to an experimental flow over a backwards facing step. The first step is to create an initial sampling grid, with a spacing of 48px, as shown in Figure 6.12, which is nothing more than a regular grid. The choice of 48px spacing is to more closely mimic a standard analysis approach, which is discussed later in this section, wherein a window size profile of $99 \rightarrow 49 \rightarrow 25$ is adopted, along with a window overlap ratio (WOR) of 50%, resulting in an initial grid spacing of 48px. On this grid, the adaptive initial window size algorithm, presented in Section 4.6, can be applied, and the resulting displacement field validated using the nomalised median threshold (NMT)

validation approach. Both the initial WS and validated displacement field are shown in Figure 6.13.

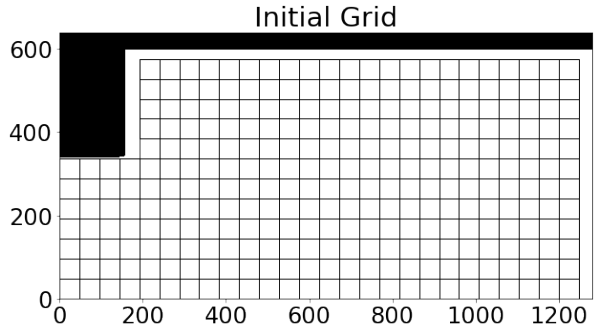
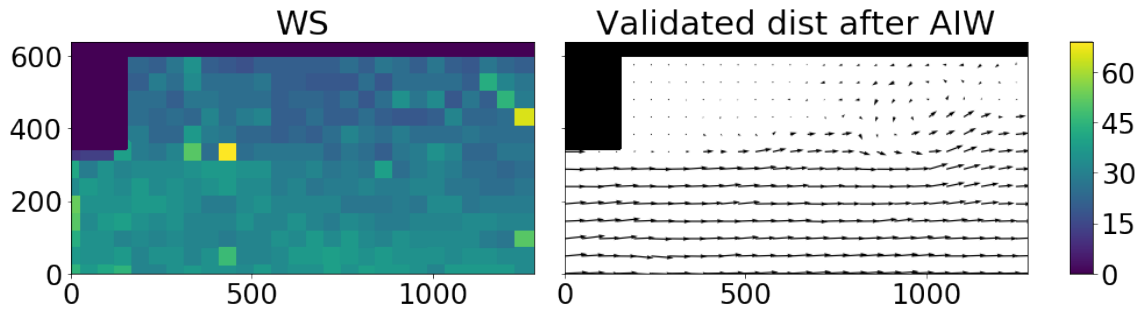


Figure 6.12: Initial sampling grid for the adaptive multigrid refinement algorithm. Note that correlation windows are centered at the grid vertices.



(a) (b)

Figure 6.13: Example of the (a) initial window size, obtained using the AIW algorithm from Section 4.6, and (b) the resulting (validated) displacement field from the first iteration.

From the coarse sampling grid, a pixelwise cubic and linear interpolation can be computed using the methodology presented earlier in this chapter (Section 6.2). The difference between the two interpolants can be calculated for each velocity component, and subsequently combined using the euclidean norm, to obtain a pixelwise defined objective refinement function (ORF) as shown in Figure 6.14(a). Cells which contain a peak ORF value above some threshold, currently set to 0.07px, can then be split, resulting in a sampling grid that looks like Figure 6.14(b). The vertices of this grid represent the central location of correlation windows. The threshold of 0.07px has been chosen considering the typical uncertainty of correlation to be approximately 0.1px or less, and that the reconstruction error is typically less than the ORF, as seen in

Figure 6.11, and therefore the threshold of the ORF should be somewhat *larger* than the assumed inherent uncertainty.

Afterwards the image is deformed, using a quintic B-spline image interpolation scheme [88, 52], before the process is allowed to repeat with the following minor

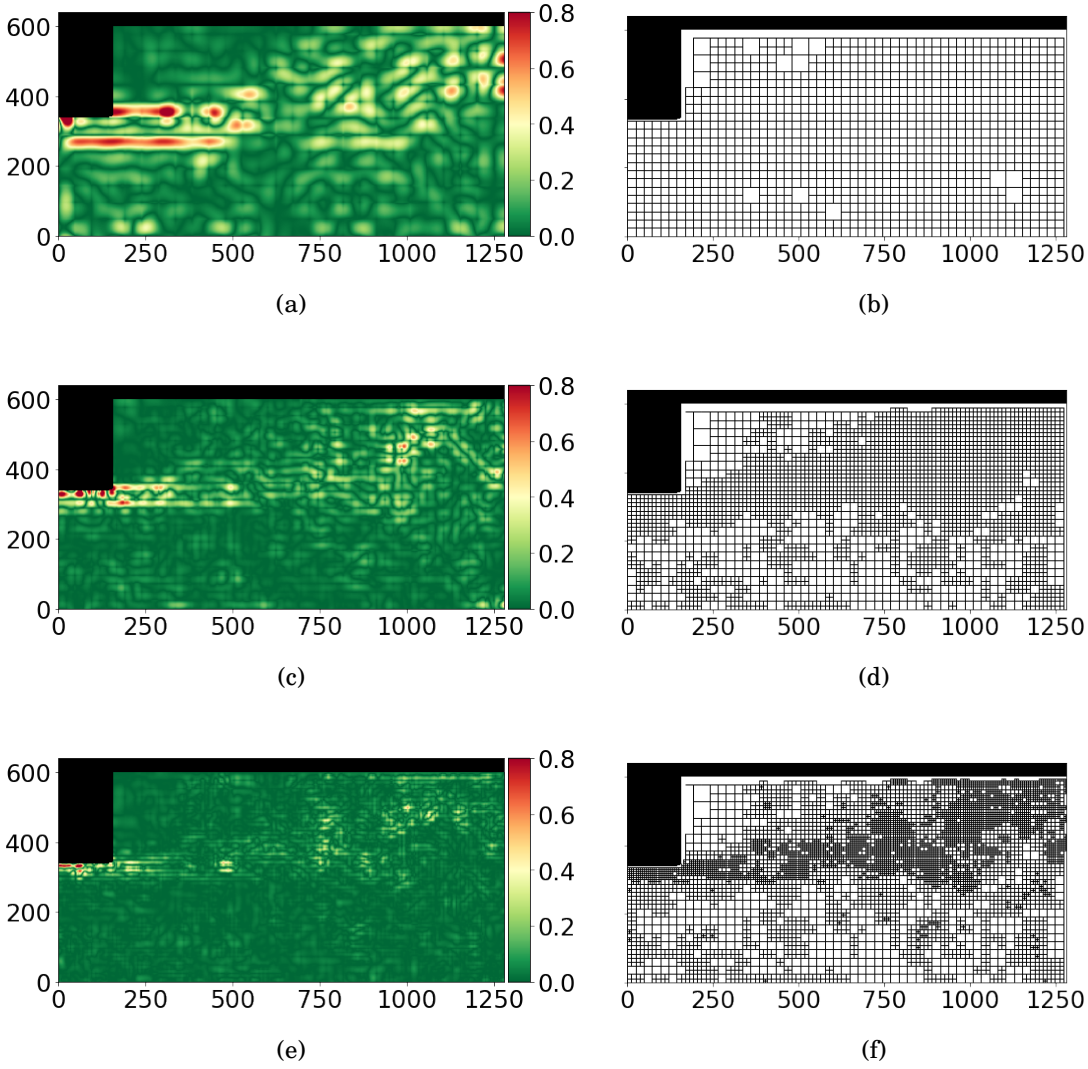


Figure 6.14: (a), (c), (e) Objective refinement functions after the first, second, and third iterations, respectively, calculated as the euclidean norm of the delta between a bicubic and bilinear interpolation of the same data points. Units in px.
(b), (d), (f) Sampling grids following refinement according to (a), (c), and (e), respectively, where the peak cell value is greater than 0.07px. Note that correlation windows are centered at the vertices. Furthermore, the sampling grid in (f) is for visualisation only, and was not used to sample the domain, as only three refinement iterations were applied.

adjustments. Firstly, the window size is reduced linearly towards a final WS, calculated from the seeding density such that the window contains approximately 15 particle images. Secondly, to allow for the variation in inter-sample distance imposed by the non-uniform grid, inverse distance weighting is applied when validating vectors, as per [40] (see Sections 2.11 and 4.4.5). Once again, both the cubic and linear interpolations are computed, from which the ORF is calculated, shown in Figure 6.14(c), and the grid refined where the peak value exceeds 0.07px, as shown by Figure 6.14(d). In the third and fourth iterations, the WS is set to the final WS, i.e. based on seeding, and the image interrogated at the new grid locations. Repeating the final main iteration, i.e. the third iteration, mimics a WIDIM approach and allows for additional convergence to be reached.

The simplicity and robustness is well domonstrated by the buildings flow case as introduced earlier. Figure 6.15 shows the final displacement magnitude when analysing two image pairs with the same analysis configuration. For Widim, this was an initial window size of $97px^2$ reducing to $25px^2$ over three primary iterations, followed by a single refinement iteration with fixed window size and grid spacing. The adaptive multigrid analysis was configured to also have three primary iterations, where grid refinements and window size reduction is allowed, followed by a single fixed refinement iteration with a final WS of $25px^2$ to match the Widim analysis. The initial window size was obtained using the AIW algorithm from Chapter 4. Both 50% and 75% overlap were investigated for Widim, while the multigrid initial grid spacing was configured to match the grid spacing.

While this Widim configuration works well for one image, it performs poorly for another, particularly at higher overlap settings, as seen by the spikes in displacement caused by undetected outliers in the final displacement field. The cause of these outliers is due to poor seeding in this region, which does not cause an issue for the adaptive method for two reasons. Firstly, windows are more correctly sized from the offset, and, secondly, fewer windows are placed in this region of reasonably uniform flow. Adding more windows to regions of poor image quality can lead to clusters of outliers, which are notoriously difficult to resolve. Furthermore, the flow structures in the adaptive method remain almost identical to those in the Widim analysis. This is the expected behaviour as in regions where the AMG grid is fully refined the sampling is equivalent to Widim in this case.

In addition to the improved robustness, reduced user-dependence, and equivalent performance in regions of interest, the AMG method is notably quicker than its Widim equivalent. A comparison of run times is shown in Table 6.1 where the times were

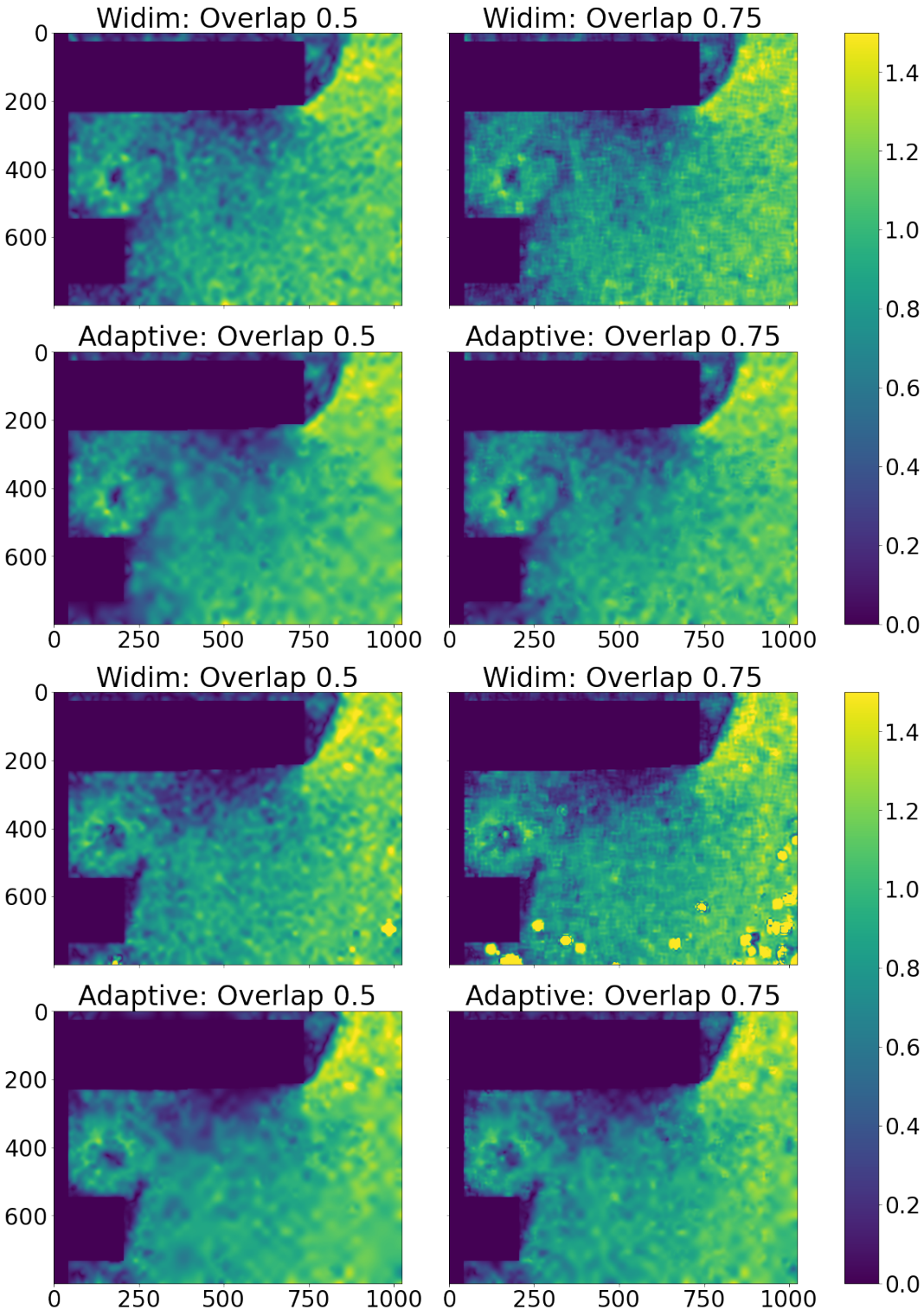


Figure 6.15: Comparison of robustness when analysing two different image pairs using the same configuration settings for regular Widim vs Semi-Structured multigrid with AIW. Shown here is the displacement magnitude.

	50%	75%
AMG	1.00	2.04
Widim	1.20	3.99

Table 6.1: Comparison of run time for Adaptive MultiGrid and Widim for 50% and 75% overlap. Normalised by the run time for AMG with 50% equivalent overlap.

obtained by averaging over seven repeats and then normalised by the run time for AMG with 50% equivalent overlap. The larger the desired overlap, the more efficient AMG is likely to be, relative to Widim, since the approximately fixed cost to determine the seeding density becomes a smaller fraction of the run time, and more correlations can be avoided in uniform regions.

While comparing the displacement magnitude in Figure 6.15 helps to demonstrate the improvements in robustness, another important aspect of PIV analyses is gradient information. One common concern is the noise in the gradient field, which may be exacerbated by polynomial interpolation. Considering the interpolation methodology, as presented in Section 6.2, it is expected that in the event that all cells in the AMG method are refined the interpolation tends towards an equivalent, regular, full, cubic interpolation. Accordingly, noise in the gradients will be no worse than a regular Widim analysis with equivalent final grid spacing.

Instantaneous gradient magnitudes are shown in Figure 6.16 for both Widim (top) and AMG (bottom) [2]. Gradients for Widim are obtained by finite differencing of the sampling grid. Gradients for the AMG method are also obtained from the same size sampling grid. As some of these grid locations do not have a sample, these locations have a displacement interpolated using interpolation as per Section 6.2. In both cases, other than the top-hat mean-filter applied at the predictor stage (as per Schrijer and Scarano [54]), no additional smoothing has been applied.

When looking at the main regions of interest, the measured gradients for both methods are practically indistinguishable. However, it can be seen that there is a generally lower level of background fluctuations in the relatively freestream region. This is due to the lower frequency sampling in these regions resulting in a small low-pass filtering effect. This is best observed when looking at the instantaneous vorticity, as shown in Figure 6.17.

[2]The absolute value of gradients is shown here to aid visualisation and comparison

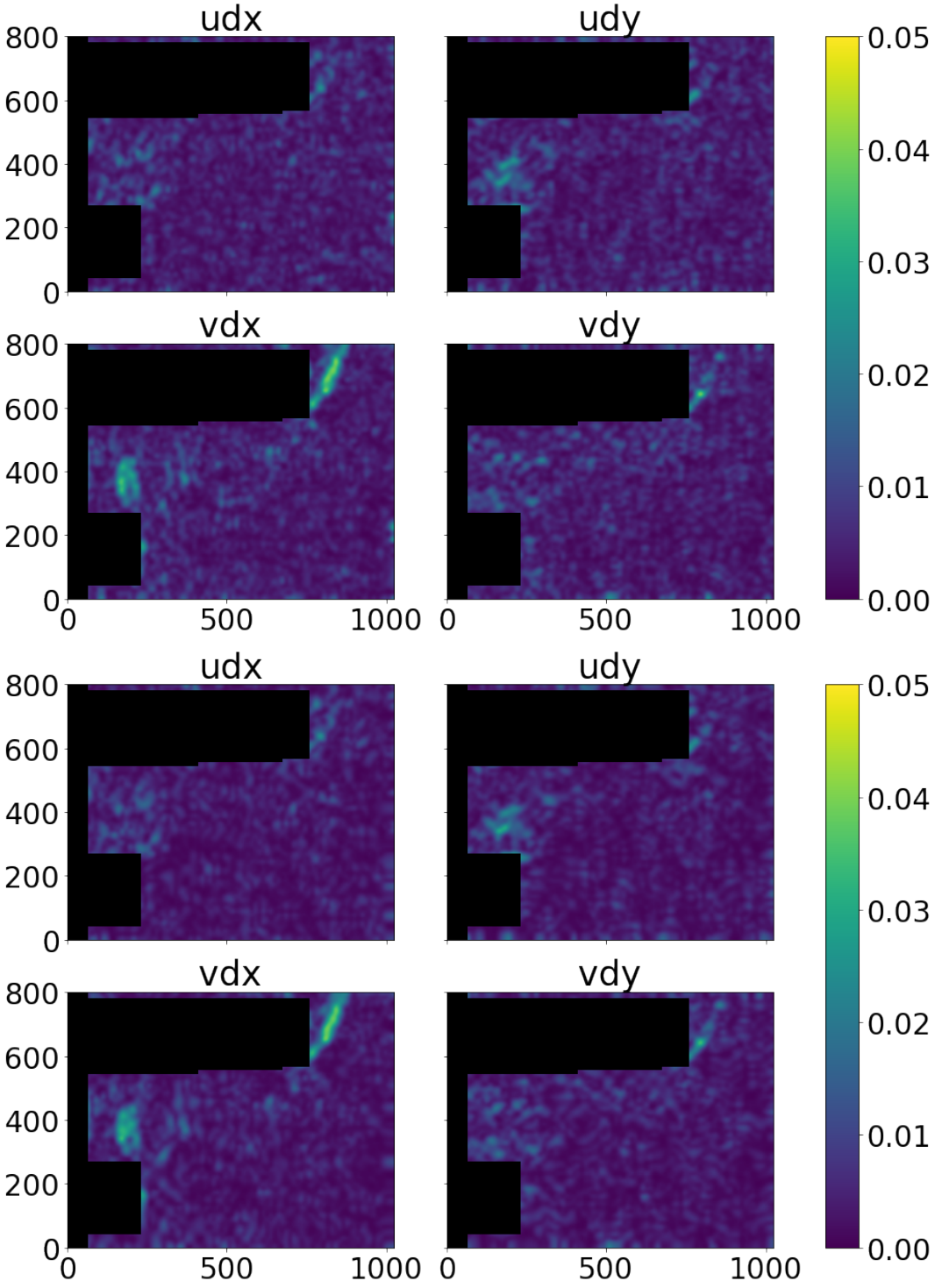


Figure 6.16: Comparison of instantaneous displacement gradients for Widim (top) and AMG (bottom). Absolute values plotted to aid visualisation.

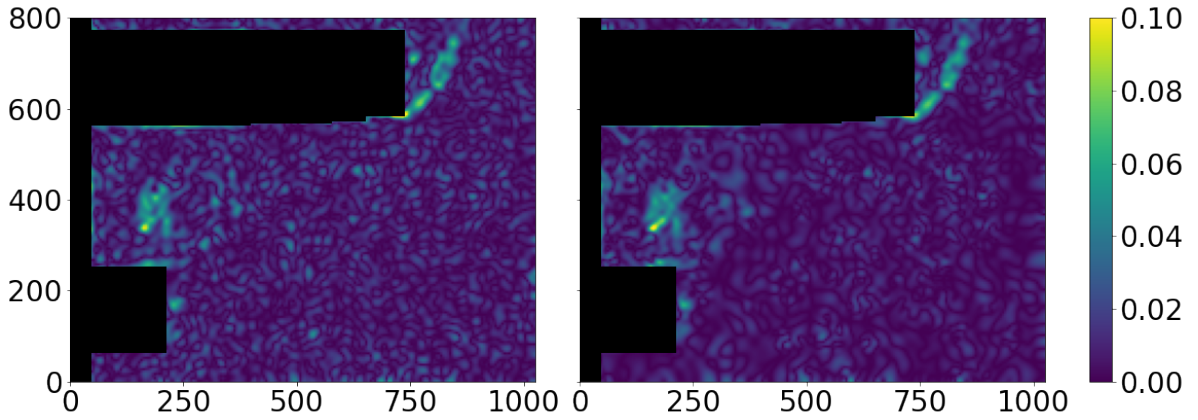


Figure 6.17: Comparison of instantaneous vorticity magnitude for Widim (left) and AMG (right). Absolute values plotted to aid visualisation.

6.4.2 Ensemble comparison

Beyond the simple nature of this approach, particularly relative to unstructured adaptive approaches seen previously, this method is able to be run almost entirely independently of user input, and this advantage should not be overlooked. When considering that the threshold for refinement may be based upon known cross-correlation uncertainty, i.e. in the region of 0.05-0.1px, which remains reasonably constant in differing conditions, the only remaining non-universal parameters are the initial grid spacing and the number of iterations. Yet even these parameters can be set to values that are close to universal.

The initial grid spacing should be chosen to allow for sufficient refinement of the sampling grid over several iterations, where at each refinement the central location must be an integer pixel location, suggesting that a power of two multiple should be used. Coupling this requirement with general experience of initial grid spacing leads to an initial grid spacing of 64px or 128px. In fact, when using either of these initial spacings, it is common that nearly all cells are refined following the first iteration, indicating insufficient sampling. On the other hand, a grid spacing of 32px represents quite intensive analysis conditions which may not be necessary for all regions of the image, therefore, a suitable compromise can be found by setting the initial spacing to 48px. Although not a power of two, this value allows a final grid spacing of 3px to be achieved if desired, which, for PIV, is approximately the upper limit for spatial resolution anyway[3]. Further research of this method could determine if an optimal, adaptive, initial

[3]A sample spacing below three pixels is unlikely to yield improvements in spatial resolution due to the spatial averaging of cross-correlation. At this grid spacing it is highly probable that neighbouring windows sample the same combination of particles, and thus would yield the same displacement value, and hence

spacing could be determined based on, for example, the particle density or spacing.

While the number of iterations must still be decided by the user, a value of three is often conducive to good results. As a result, this method can be applied to almost any image and yield reliable, efficient, results without any user-tuning. Two avenues of research on iteration count remain appealing. Firstly, an automatic stopping/refinement criterion could be investigated, for example, based on the domain-wide peak value of the ORF, the number of cells with ORF value above the threshold, or some combination of the two. Secondly, grid refinement is currently limited to once per iteration, which therefore couples the initial/final grid spacing with the number of iterations. Allowing multiple refinements per iteration may allow for greater flexibility in the AS approach.

In addition to user-independence, this method offers a reduction in computational cost for a given peak sampling intensity. Whether this cost reduction is used to accelerate the analysis process, or to allow more intensive sampling in an allotted amount of time is left to the user, however, the former will be demonstrated in the current works, due to the absence of higher fidelity results to act as a reference solution when comparing difference sampling intensities.

To demonstrate this reduction in cost, an ensemble of experimental images will be analysed by the adaptive multigrid refinement (AMR) algorithm presented above and the results juxtaposed with a comparable conventional interrogation solution. The conventional WIDIM analysis used an initial window size profile of 97px → 49px → 25px → 25px, with a window overlap ratio of 50% such that the grid spacing is equivalent to that of the AMR approach. In addition to this conventional analysis, a structured adaptive *interrogation* (SAI) approach, wherein the locations remain fixed but windows' sizes are adaptively determined from the start, was also applied to the same set of images to demonstrate the influence of window size. The sampling grid for this SAI method is equivalent to the WIDIM method.

The motivation for this additional SAI analysis is to allow separation of the impact of window size and of sample locations. Regions where the SAI solution differs from the Widim solution are driven purely by window size variations, however, regions where the SAI differs from the adaptive method are driven by sample location differences. The intention is to show that differences in regions of interest due to sampling density are negligible, since the adaptive method should naturally add more windows here. Moreover, differences in lesser regions of interest should be limited despite the use of fewer sampling resources. In this way, it is shown that this method's sampling will,

such a grid spacing may act, counter-intuitively, as a low pass filter in some regions.

automatically without the input of a user, tend towards the already accepted state-of-the-art result where it matters without incurring the same computational penalty to do so, additionally with more optimally sized windows which are known to be beneficial to the solution.

Figure 6.18 compares the AMR ensemble mean displacement field with both ordinary WIDIM as well as the SAI. As would be expected, differences between AMR and SAI, c.f. Figure 6.18(a), remain small over the domain since differences are driven only by variations in sample spacing. In the recirculation region window size and density are largely the same, and hence differences are negligible, while in the freestream differences are slightly larger yet this is achieved with fewer correlation windows, as shown in Figure 6.14(d) which represents a typical sampling grid for the AMR method. On the other hand, differences in the recirculation region for the standard WIDIM approach are larger, due to a difference in window size. Without a higher fidelity reference solution against which to compare, it is not possible to quantify the effects of the adaptive window sizing, however, as demonstrated in Section 2.10.3 smaller, more appropriately sized windows will reduce the error associated with cross-correlation displacement evaluation, particularly in regions of high gradients. This can be observed in the shear layer, where the AMR solution's displacement is larger near the freestream and smaller near the stagnation, suggesting an improvement of spatial resolution and a reduction in low-pass filtering effect.

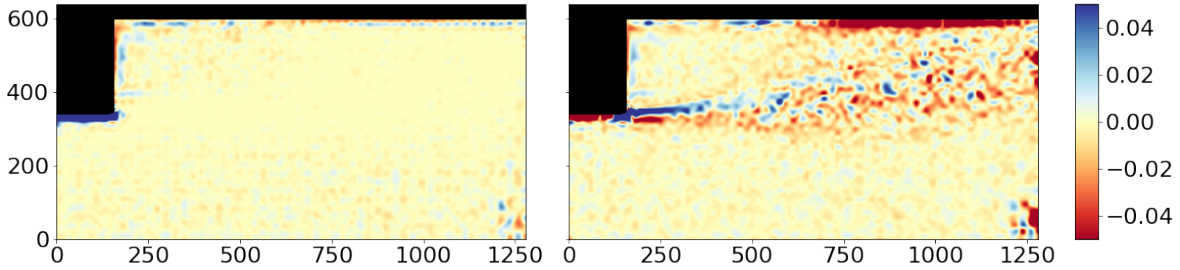


Figure 6.18: Comparison of the mean displacement field delta for (a) $|U_{SAI}| - |U_{AMR}|$, and (b) $|U_{WIDIM}| - |U_{AMR}|$. Units in px.

Another impact of using smaller windows is increased uncertainty, due to fewer particle images contributing to the correlation map. Noticeable differences can, therefore, be observed in the standard deviation (std) of the solution, as shown in Figure 6.19, where Figure 6.19(a) represents the standard WIDIM approach, 6.19 represents the AMR, and 6.19(c) represents the SAI. The spikes in std for the WIDIM approach are indicative

of outliers that have not been corrected for and suggest incorrectly sized windows. To highlight the differences, Figure 6.20 shows the difference between the AMR standard deviation and both the structured approaches. Figure 6.20(b) demonstrates the increased background level of uncertainty, for the SAI method relative to ordinary WIDIM, in the turbulent region due to the smaller windows. Also clear are the localised peaks, caused due to the increased presence of outliers, which are testified by Figure 6.20(a) to be almost exclusively due to window sizing. However, Figure 6.20(a) also helps justify the lack of refinement in the freestream, since with or without window size adaptation, the level of fluctuations remain approximately constant.

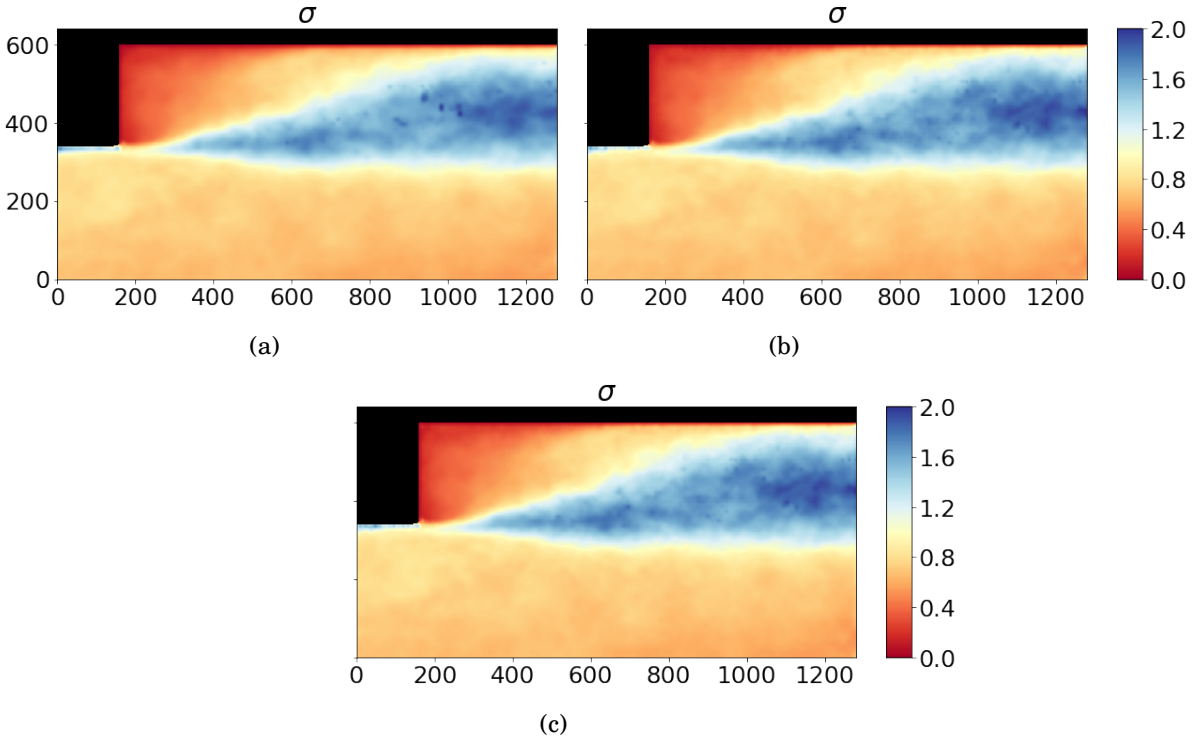


Figure 6.19: Comparison of the temporal standard deviation of the displacement field, showing the euclidean norm of both components, for (a) conventional WIDIM analysis, (b) adaptive multigrid refinement, and (c) conventional grid as per (a) with adaptive window sizing. Units in px.

Also noticeable is a difference near the top of the step, this region of the image suffers from a bright reflection, often impeding cross-correlation, and no attempt to adjust window eccentricity or relocate vectors, as can be beneficial near flow interfaces [117]. Accordingly, the displacement in this region is likely to be biased towards the freestream value. Such a bias may be alleviated by the use of more appropriately sized windows. It

is unclear as to exactly what is causing the differences in this region, since there exists non-negligible differences for similar sampling density and window sizing, as per Figure 6.20(a). One theory is that a cascading effect is observed in the vector validation process. Near the vertex of the step, the vector spacing in the slow-moving recirculation region is typically lower than for the fully structured approach. Since this will lead to a different neighbour composition and different validation behaviour, the classification of the vector may change. In such an event, neighbouring vectors can no longer use this for vector replacement, and therefore the replaced value of the neighbour will be different, causing a different image deformation, and so on.

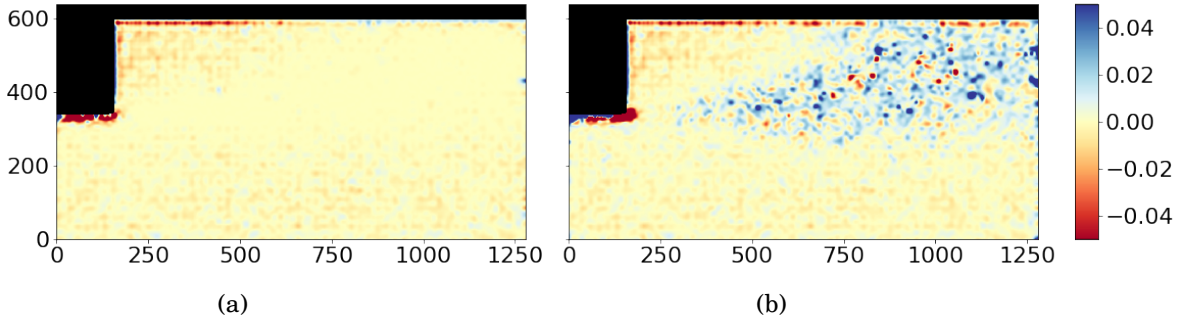


Figure 6.20: Comparison of the standard deviation delta for (a) $|\sigma_{SAI}| - |\sigma_{AMR}|$, and (b) $|\sigma_{WIDIM}| - |\sigma_{AMR}|$, where $|\sigma_{(.)}|$ represents the euclidean norm of the std in both component directions for the given methodology, as shown in Figure 6.19. Units in px. Note that the scale has been selected to better highlight the differences, and does not correspond to peak values.

Notably, however, the differences between the AMR and SAI approaches remain small, despite the use of fewer correlation windows in the freestream for the spatially adaptive approach. In fact, each image required 20% fewer correlations on average. Furthermore, this 20% is a relatively lower bound on the reduction in correlations - a further refinement iteration quadruples the number of correlations for WIDIM, whereas typically only a doubling is observed for the AMR approach in this case, resulting in an overall reduction of about 50% of the correlations for this higher resolution analysis.

6.5 Summary

This chapter presents a step away from traditional adaptive sampling approaches. While fully unstructured distribution methods offer the most flexibility in sampling density and allow for easy handling of arbitrary domain shapes, they are associated with a

notable increase in computational cost and complexity which make such approaches less attractive in light of the ever-decreasing cost of computational power. Moreover, all existing distribution methods, as presented in Chapter 5, contain some element of random behaviour. Random number generators can be seeded with a fixed value to allow for re-production of results, however, it only takes a small divergence of input parameters to result in entirely different random sequences. Without fixing the random number seed, the analysis of a PIV image pair will result in differing solutions despite using the exact same analysis configuration, leading to increased uncertainty in the solution and difficulty in performing parametric studies, or testing of new methodologies, since these fluctuations cloud the effect of the varied parameter.

The completely novel semi-structured approach presented in this chapter avoids many of these issues, and, crucially, remains conceptually simple. Notably, adopting a semi-structured sampling grid in this manner enables the use of a multi-level interpolation scheme, which greatly reduces the time required to interpolate the data. The multigrid architecture is presented which maintains a grid of cells, with interrogation windows located at the vertices, which can be split to result in four more cells, at a higher tier, and five more correlation windows. Neighbour information is continually maintained by enforcing two simple criteria; firstly, neighbours are only tracked in the tier to which they belong, and secondly, adjacent cells may be no more than one tier different. These criteria ensure that only the immediate parent cell and it's neighbours must be referenced when adding new cells, resulting in good computational efficiency.

Sampling locations are, therefore, restricted to certain positions, reducing the available flexibility sampling density variations, relative to previous fully unstructured methods, yet ensuring that the locations are deterministic. Despite the reduced flexibility, it was shown that this architecture can be used as a stand-alone distribution method akin to the AIS or SFD methods, and, in fact, is considerably quicker, albeit with a less gradual change in sampling density under some circumstances. However, the real benefit of this architecture is that it more easily facilitates a refinement-based, rather than objective based, adaptive sampling methodology.

Using the magnitude difference between a cubic and linear interpolation of the same sample data as a heuristic for the local curvature and uncertainty, cells can be split by comparing such difference with an imposed threshold. This threshold should be somewhat larger than the anticipated correlation uncertainty, and reasonable values are between 0.05 and 0.1px. Not only does this allow for the sampling grid to persist over all iterations, i.e. it does not need to be re-created each iteration, it also relaxes the

yet-unanswered question of the required number of correlation windows to distribute.

Further research into the relationship between the aforementioned heuristic and the actual error of the solution should be conducted to allow for better thresholding and performance. Furthermore, although not investigated here due to time constraints, this approach should allow for a reasonable stopping criterion to be defined based on the global values of the refinement criterion, indicating whether another iteration is required or not.

Finally, when coupled with the improved seeding density estimation procedure and novel adaptive initial window sizing (AIW) routine introduced in Chapter 4, this method can be evaluated with practically no user-input, addressing one of the key motivators for this research - autonomy.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

PIV has presented itself as one of the primary methods for experimental analysis of fluid motion, mainly due to its non-intrusiveness and ability to capture information from a wider field-of-view simultaneously, unlike similar pointwise techniques, e.g. laser doppler anemometry (LDA) or hot-wire anemometry (HWA). Advances in hardware continue to improve the attainable spatial resolution of PIV, with 4 megapixel (MP) cameras now being considered standard, if not small, and with sensors up to 50MP now available. Additionally, improvements in both camera and laser hardware allow for even higher framerates to be achieved, improving the temporal resolution, and the amount of data to be processed. Both of which significantly increase the computational requirements to process the resulting data.

PIV image analysis routines, on the other hand, have remained largely the same over the past decade and a half, instead of relying on increased availability of computational resources to match the increased data production. Moreover, such standardised routines are strongly user-dependent and force the user to trade-off between robustness, resolution, and computational cost. It is worth stressing that this is not solely experience-dependent, neither, since notable solution differences were observed during the PIV challenges, where the same PIV image datasets were analysed by many groups of PIV experts [3]. Not only do these approaches depend significantly on the user, but they are also known to highly couple numerous interrogation parameters, and impose spatially homogeneous analysis conditions, often resulting simultaneously in regions of under-sampled flow and regions of over-sampled flow.

The objective of this research, therefore, was to develop image analysis routines that can maximise the information extracted from PIV images, autonomously and efficiently. Furthermore, such analysis should, ideally, be able to adapt to the data at hand to provide more optimal analysis settings. To achieve this, more flexibility in the analysis process was desired, specifically, de-coupling of various components of the algorithm. This research has made good progress towards these objectives.

One manner in which more data can be extracted *for a finite amount of resource*, is to restrict the usage of such resources once the solution is sufficiently resolved. Such an approach may use significantly fewer resources for a given configuration, allowing a higher fidelity analysis to be performed for the same total resource consumption. This is the focus of Chapter 3, wherein confidence intervals are used to identify regions with sufficient convergence, which can subsequently be neglected by the analysis process. An important finding from this work, however, is that the analysis process has a fairly large spatial footprint. In other words, a change in the solution at one location, due to the absence of a now-skipped data point, may cause differences to the solution some distance away due to the spatial extent of interpolation, window size, and vector validation. It was shown how this spatial extent can be determined, and the method validated on an ensemble of 115 images, where the number of correlations was shown to be reduced by 44%. Determination of the sampling masks is relatively simple, and adjusting the analysis routine to respect such maps is trivial, making the method easy to implement. The method is best suited for when quick, approximate, results are desired, for example on-the-fly calibration of wind-tunnel experiments where tunnel-time is at a premium, or when the region of interest (ROI) is limited to a small subset of the image.

Undesirably, such an approach is still limited to the structured, rigid, coupled implementations that this research set out to address. Chapter 4 investigated the de-coupling of such implementations, exploring the implications of attempting to incorporate adaptivity to PIV image analysis. It was found that even de-coupling seemingly simple parameters, for example, trying to allow the window size to vary over the domain, caused a cascading effect, necessitating many algorithmic changes, highlighting many previously-unanswered questions. While these questions, such as "how many windows are optimal?", "how many iterations should be performed?", or "given the flexibility, where should windows be placed for maximum efficiency?", may be answered in time, they are not questions that have received a great deal of attention until now - primarily because they have been hidden away by the aforementioned coupling.

Towards the objective of autonomous image processing, Chapter 4 also presents a

novel approach to automatically determining initial correlation window size. The method is robust and effective, and promotes overall analysis efficiency, by virtue of selecting smaller, more appropriately sized windows. This technique requires no user-input and, when coupled with existing adaptive final window sizing algorithm, truly allows for a PIV image to be analysed, and provide good results, completely free from user input.

Chapter 5 addresses another of the questions identified by Chapter 4 - fully unstructured, arbitrary, sample distribution. The key finding of this chapter is that a good distribution quality, wherein samples are locally equispaced, changes in sampling density are more gradual, and there are no excessive voids, is necessary for two reasons. Firstly, it impacts upon the interpolation accuracy, and, secondly, it can hinder the ability of vector validation routines to filter erroneous data. Existing approaches to unstructured sample distribution were investigated and found to either lack distribution quality, or be unacceptably computationally intensive. Attempts to optimise the latter method, yielded improvements, but to an insufficient standard. Therefore, a method inspired by computer graphics was adapted for PIV sample distribution, called adaptive incremental stippling (AIS), and shown to possess similar efficiency to the first alternative, yet with comparable quality to the latter method.

Despite the benefits that unstructured adaptive sampling can bring, another key finding of this research is that randomness, required by all investigated distribution methods in order to attain the desired efficiency, leads to increased solution uncertainty and obfuscates the development process. While random number generators can be seeded with a fixed value, the relationship between input parameters and sample distribution is highly non-linear; reducing the number of samples by one may completely alter the resulting distribution. Such non-linear behaviour drastically inhibits the ability to perform parametric studies, and hence also inhibits further investigation into unstructured adaptive methods, since now natural fluctuations must be isolated from artificially created fluctuations. Such behaviour effectively opposes an ethos of one of the key drivers of this research: autonomy and user-independence; the solution quality should no less depend on the user's experience than it should depend on random chance.

It is still believed that a fully unstructured sampling routine can be beneficial in situations where the ROI represents only a small subset of the image, with easily resolvable freestream elsewhere. This is particularly true where a high window overlap ratio would otherwise be required to properly sample the flow, wherein the computational savings afforded by the efficient locations of samples can be upwards of 3-4x. Despite this belief, work is still required to develop an optimal objective function to accurately

detect such regions of interest and balance their sampling with the need to explore the entire domain sufficiently. Given the complexity of the methods involved, the challenges involved with unstructured sampling, and the rising availability of High Performance Computing (HPC), the attractiveness of *fully-unstructured* adaptive sampling remains uncertain.

Building on this finding, an attempt for a deterministic adaptive sampling approach has been investigated in Chapter 6, with very promising results. By restricting sample locations to a semi-structured grid, operations such as interpolation and vector validation are able to remain computationally efficient, with a relatively minor overhead relative to their fully structured counterparts. Furthermore, when used to create sample distributions it is shown to be approximately four times faster than the previously proposed AIS method. The approach is capable of producing distributions with semi-arbitrary sampling density variations, although, owing to the discrete nature of the semi-structured grid, the smoothness of density variations is somewhat diminished.

However, while the multigrid approach offers a significant improvement when used as a standalone distribution methodology, the real benefit is that it more readily facilitates a refinement-based adaptive sampling approach. Due to the newfound efficiency of interpolation, a heuristic for both the curvature and uncertainty is proposed by determining the magnitude of difference between a bicubic and bilinear interpolation of the same data. By refining cells wherein the uncertainty heuristic is above some threshold, the sampling density is naturally refined where it is required, yet only to a sufficient extent and no more.

The *pièce de résistance* of this approach is how well it achieves all of the original motivations of this research. It is able to spatially adapt the analysis conditions to reflect the underlying data, especially when windows are automatically sized using the novel method proposed in Chapter 4. It is more efficient than a conventional structured approach since many fewer correlation windows are required to produce a result of equivalent status, yet remains a fundamentally simple approach. Most notably, since the heuristic thresholded can be pseudo-universally set, this approach enables an almost entirely automatic approach to PIV image analysis, free from user input. This method provides a solid platform to build on when developing methods which extract all of the available information in a PIV image.

Since this method tends towards a regular structured grid in the event that all areas need further refinement, with a slight computational penalty in this case, there is little penalty to using this method as the de-facto approach for image analysis. The majority

of the time, the reduced analysis of smoother regions will outweigh the small overhead associated with the adaptive methodology. Furthermore, it is important to consider the fact that user-time is more valuable than compute-time, and that this method requires very little input from the user nor the laborious trial-and-error of parameter selection.

The key findings from this work can therefore be summarised with the following list:

- It is possible to significantly optimise the analysis of an ensemble of images by restricting analysis to unconverged regions using masks.
  - It is important to consider the spatial footprint of the analysis to prevent artificial artefacts.
  - In a representative example, a reduction of 44% of the number of correlations was observed.
- The current standard method, WIDIM, contains highly-coupled parameters.
  - This restricts optimal image analysis and requires undesirable trade-offs between resolution and robustness.
  - De-coupling parameters triggers a cascading effect, requiring many other parameters to also be de-coupled.
  - This cascading opens up many unanswered questions, suitable for future research.
- A fully unstructured image analysis routine can provide the necessary flexibility to analyse images more optimally, but may not offer enough of an improvement to justify the complexity
  - Accurate seeding density information is required to guide window sizing
    - Particle detection should be optimised to improve binary detection of particles - the precise centroid location is of less importance.
    - Seeding density estimation can be optimised by choosing the averaging kernel according to the global mean seeding density.
  - It must be possible to distribute samples quickly and robustly, while ensuring good interpolation and vector validation performance
    - Sample spacing variations should be gradual - localised clusters of samples significantly inhibit vector validation routines.

        * Necessary computational efficiency is obtained through randomness, which leads to additional artificial measurement uncertainty.
    - Unstructured interpolation is possible, but is approximately an order of magnitude slower than structured counterparts
    - Consensus on the best objective function, i.e. broadly where to place samples, has not been reached.
    - Fully unstructured adaptive sampling is still beneficial where the range of flowscales in an image is large and inhomogeneous.

- A semi-structured image analysis routine can provide much of the flexibility of fully-unstructured, with far fewer disadvantages
    - A regular grid can be refined only where it is deemed necessary.
    - A simple, yet highly-effective, refinement criterion is the difference between linear and cubic interpolation of the same data.
    - Multi-level semi-structured interpolation is significantly faster than fully unstructured.
    - Localised clustering of data is not possible, facilitating high quality interpolation and good vector validation performance.

- In order to maximise automatic information extraction from PIV images, it must be possible to quantify the amount of information available.
    - This could be through simple heuristics, such as linear vs cubic, or through more elaborate uncertainty quantification techniques, gaining popularity in the community.

7.1 Future work

The obvious next steps for this research is development of the semi-structured adaptive sampling approach presented in Chapter 6. Amongst the numerous interesting topics of work, three stand out as key steps towards a fully autonomous, optimal, PIV image analysis algorithm. Firstly, in the current research, a semi-universal initial grid spacing is suggested, and while this is known to produce adequate results under a wide range of circumstances, the optimality of this initial spacing is as yet unknown.

The effect of initial grid spacing on the final solution should be investigated, with the goal of automatically determining this spacing in mind, perhaps from the observed seeding density. Secondly, the final grid spacing of this method is inherently linked to the number of iterations, as it is in the conventional image analysis routines. Currently, only a single refinement is permitted per iteration, yet this limits the smallest sample spacing to $0.5h_0^K$ px, where h_0 is the initial grid spacing and K is the number of iterations, which conflicts the desire to have as few iterations as possible with the desire for high sampling frequency where necessary. Thirdly, since a heuristic for the uncertainty is now available, this offers itself as a candidate for automatic stopping criteria. In other words, the algorithm can automatically determine whether to perform another refinement and another iteration or not. When all combined, this work should represent a truly fully-independent PIV image analysis program. Additionally, although the selective spatial analysis approach from Chapter 3 may not be feasibly applicable to an unstructured data-dependent algorithm, it is supposed that a variant could be applied to this semi-structured approach, yet further improving the efficiency of image analysis.

In order to fully realise the objectives of this research, i.e. maximum automatic information extraction, it will be necessary for the resulting method to be fundamentally aware of how much information is *available.* For this, it is anticipated that developments on the semi-structured method proposed in Chapter 6 should incorporate some of the flavours of the growing topic in PIV; uncertainty quantification.

BIBLIOGRAPHY

[1] M. Raffel, C. Willert, S. Wereley, and J. Kompenhans, *Particle Image Velocimetry - A Practical Guide*, vol. 53.
Springer Science \& Business Media, 1989.

[2] M. Stanislas, K. Okamoto, C. J. Kähler, and J. Westerweel, "Main results of the Second International PIV Challenge," *Experiments in Fluids*, vol. 39, no. 2, pp. 170–191, 2005.

[3] M. Stanislas, K. Okamoto, C. J. Kähler, J. Westerweel, and F. Scarano, "Main results of the third international PIV Challenge," *Experiments in Fluids*, vol. 45, pp. 27–71, jun 2008.

[4] A. Melling, "Tracer particles and seeding for particle image velocimetry," *Measurement Science and Technology*, vol. 8, no. 12, pp. 1406–1416, 1997.

[5] J. Hong, M. Toloui, L. P. Chamorro, M. Guala, K. Howard, S. Riley, J. Tucker, and F. Sotiropoulos, "Natural snowfall reveals large-scale flow structures in the wake of a 2.5-MW wind turbine," *Nature Communications*, vol. 5, no. May, 2014.

[6] M. Stanislas and J. C. Monnier, "Practical aspects of image recording in particle image velocimetry," *Measurement Science and Technology*, vol. 8, no. 12, pp. 1417–1426, 1997.

[7] A. Boillot and A. K. Prasad, "Optimization procedure for pulse separation in cross-correlation PIV," *Experiments in Fluids*, vol. 21, no. 2, pp. 87–93, 1996.

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

[9] D. G. Abdelsalam, M. Stanislas, and S. Coudert, "PIV camera response to high frequency signal: Comparison of CCD and CMOS cameras using particle image

simulation," *Measurement Science and Technology*, vol. 25, no. 8, p. 084007, 2014.

[10] B. Lecordier and J. Westerweel, "The EUROPIV Synthetic Image Generator (S.I.G.)," *Particle Image Velocimetry: Recent Improvements*, pp. 145–161, 2004.

[11] M. Marxen, P. E. Sullivan, M. R. Loewen, and B. Jähne, "Comparison of Gaussian particle center estimators and the achievable measurement density for particle tracking velocimetry," *Experiments in Fluids*, vol. 29, no. 2, pp. 145–153, 2000.

[12] R. J. Adrian, "Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow: speckle velocimetry vs particle image velocimetry," *Applied Optics*, vol. 23, p. 1690, jun 1984.

[13] A. K. Prasad, "Stereoscopic particle image velocimetry," *Experiments in Fluids*, vol. 29, no. 2, pp. 103–116, 2000.

[14] P. H. Biwole, *Large scale particle tracking velocimetry for 3-dimensional indoor airflow study*.
PhD thesis, National Institude of Applied Sciences of Lyon, 2009.

[15] F. Scarano and M. L. Riethmuller, "Advances in iterative multigrid PIV image processing," *Experiments in Fluids*, vol. 29, pp. S051–S060, dec 2000.

[16] E. A. Cowen and S. G. Monismith, "A hybrid digital particle tracking velocimetry technique," *Experiments in Fluids*, vol. 22, no. 3, pp. 199–211, 1997.

[17] S. T. Wereley, L. Gui, and C. D. Meinhart, "Advanced algorithms for microscale particle image velocimetry," *AIAA Journal*, vol. 40, pp. 1047–1055, jan 2002.

[18] U. Shavit, R. J. Lowe, and J. V. Steinbuck, "Intensity Capping: A simple method to improve cross-correlation PIV results," *Experiments in Fluids*, vol. 42, no. 2, pp. 225–240, 2007.

[19] J. Westerweel, *Digital paticle image velocimetry - Theory and application*.
PhD thesis, jan 1993.

[20] D. Dussol, P. Druault, B. Mallat, S. Delacroix, and G. Germain, "Automatic dynamic mask extraction for PIV images containing an unsteady interface, bubbles, and a moving structure," *Comptes Rendus - Mecanique*, vol. 344, no. 7, pp. 464–478, 2016.

[21] J. Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679–698, nov 1986.

[22] A. Sanchis and A. Jensen, "Dynamic masking of PIV images using the Radon transform in free surface flows," *Experiments in Fluids*, vol. 51, no. 4, pp. 871–880, 2011.

[23] A. Masullo and R. Theunissen, "Automated mask generation for PIV image analysis based on pixel intensity statistics," *Experiments in Fluids*, vol. 58, no. 6, pp. 1–11, 2017.

[24] J. Westerweel, "Fundamentals of digital particle image velocimetry," *Measurement Science and Technology*, vol. 8, no. 12, pp. 1379–1392, 1999.

[25] J. Westerweel, D. Dabiri, and M. Gharib, "The effect of a discrete window offset on the accuracy of cross-correlation analysis of digital PIV recordings," *Experiments in Fluids*, vol. 23, no. 1, pp. 20–28, 1997.

[26] R. D. Keane and R. J. Adrian, "Theory of cross-correlation analysis of PIV images," *Applied Scientific Research*, vol. 49, pp. 191–215, jul 1992.

[27] L. Lourenco and A. Krothapalli, "On the accuracy of velocity and vorticity measurements with PIV," *Experiments in Fluids*, vol. 18, no. 6, pp. 421–428, 1995.

[28] T. Roesgen, "Optimal subpixel interpolation in particle image velocimetry," *Experiments in Fluids*, vol. 35, no. 3, pp. 252–256, 2003.

[29] H. Nobach and M. Honkanen, "Two-dimensional Gaussian regression for sub-pixel displacement estimation in particle image velocimetry or particle position estimation in particle tracking velocimetry," *Experiments in Fluids*, vol. 38, no. 4, pp. 511–515, 2005.

[30] R. D. Keane and R. J. Adrian, "Theory of cross-correlation analysis of PIV images," *Applied Scientific Research*, vol. 49, no. 3, pp. 191–215, 1992.

[31] A. Masullo and R. Theunissen, "On dealing with multiple correlation peaks in PIV," *Experiments in Fluids*, vol. 59, no. 5, pp. 1–15, 2018.

[32] R. D. Keane Adrian, R.J., "Optimisation of particle image velocimeters Part 1: Double pulsed lasers," *Measurement Science and Technology*, vol. 1, no. 1, pp. 1202–1215, 1990.

[33] J. Nogueira, A. Lecuona, P. A. Rodríguez, J. A. Alfaro, and A. Acosta, "Limits on the resolution of correlation PIV iterative methods. Practical implementation and design of weighting functions," *Experiments in Fluids*, vol. 39, pp. 314–321, aug 2005.

[34] R. Theunissen, "Theoretical analysis of direct and phase-filtered cross-correlation response to a sinusoidal displacement for PIV image processing," *Measurement Science and Technology*, vol. 23, no. 6, p. 065302, 2012.

[35] R. Theunissen and M. Edwards, "A general approach to evaluate the ensemble cross-correlation response for PIV using Kernel density estimation," *Experiments in Fluids*, vol. 59, no. 11, pp. 1–17, 2018.

[36] D. P. Hart, "PIV error correction," *Experiments in Fluids*, vol. 29, no. 1, pp. 13–22, 2000.

[37] J. Westerweel, "Efficient detection of spurious vectors in particle image velocimetry data," *Experiments in Fluids*, vol. 16, no. 3-4, pp. 236–247, 1994.

[38] A. M. Shinneeb, J. D. Bugg, and R. Balachandar, "Variable threshold outlier identification in PIV data," *Measurement Science and Technology*, vol. 15, no. 9, pp. 1722–1732, 2004.

[39] J. Westerweel and F. Scarano, "Universal outlier detection for PIV data," *Experiments in Fluids*, vol. 39, no. 6, pp. 1096–1100, 2005.

[40] J. Duncan, D. Dabiri, J. Hove, and M. Gharib, "Universal outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data," *Measurement Science and Technology*, vol. 21, no. 5, p. 057002, 2010.

[41] A. Masullo and R. Theunissen, "Adaptive vector validation in image velocimetry to minimise the influence of outlier clusters," *Experiments in Fluids*, vol. 57, no. 3, pp. 1–21, 2016.

[42] A. M. Fincham and G. R. Spedding, "Low cost, high resolution DPIV for measurement of turbulent fluid flow," *Experiments in Fluids*, vol. 23, pp. 449–462, dec 1997.

[43] F. Scarano and M. L. Riethmuller, "Iterative multigrid approach in PIV image processing with discrete window offset," *Experiments in Fluids*, vol. 26, no. 6, pp. 513–523, 1999.

[44] H. T. Huang, H. E. Fiedler, and J. J. Wang, "Limitation and improvement of PIV II," *Experiments in Fluids*, vol. 15-15, pp. 263–273, sep 1993.

[45] H. T. Huang, H. E. Fiedler, and J. J. Wang, "Limitation and improvement of PIV I," *Experiments in Fluids*, vol. 15, pp. 168–174, aug 1993.

[46] K. Jambunathan, X. Y. Ju, B. N. Dobbins, and S. Ashforth-Frost, "An improved cross correlation technique for particle image velocimetry," *Measurement Science and Technology*, vol. 6, no. 5, pp. 507–514, 1995.

[47] L. Gui and S. T. Wereley, "A correlation-based continuous window-shift technique to reduce the peak-locking effect in digital PIV image evaluation," *Experiments in Fluids*, vol. 32, no. 4, pp. 506–517, 2002.

[48] T. Astarita, "Analysis of velocity interpolation schemes for image deformation methods in PIV," *Experiments in Fluids*, vol. 45, pp. 257–266, aug 2008.

[49] T. Blu, P. Thévenaz, and M. Unser, "Linear interpolation revitalized," *IEEE Transactions on Image Processing*, vol. 13, no. 5, pp. 710–719, 2004.

[50] S. T. Wereley and C. D. Meinhart, "Second-order accurate particle image velocimetry," *Experiments in Fluids*, vol. 31, pp. 258–268, sep 2001.

[51] T. Astarita and G. Cardone, "Analysis of interpolation schemes for image deformation methods in PIV," *Experiments in Fluids*, vol. 38, pp. 233–243, feb 2005.

[52] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. I. Theory," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 821–833, 1993.

[53] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing. II. Efficiency design and applications," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 834–848, 1993.

[54] F. F. Schrijer and F. Scarano, "Effect of predictor-corrector filtering on the stability and spatial resolution of iterative PIV interrogation," *Experiments in Fluids*, vol. 45, no. 5, pp. 927–941, 2008.

[55] C. D. Meinhart, S. T. Wereley, and J. G. Santiago, "A PIV Algorithm for Estimating Time-Averaged Velocity Fields," *Journal of Fluids Engineering*, vol. 122, no. 2, p. 285, 2002.

[56] U. Ullum, J. J. Schmidt, P. S. Larsen, and D. R. McCluskey, "Statistical analysis and accuracy of PIV data," *Journal of Visualization*, vol. 1, pp. 205–216, jun 1998.

[57] Federation Internationale de l'Automobile, "FIA Sporting Regulations," 2019.

[58] J. Westerweel, "On velocity gradients in PIV interrogation," *Experiments in Fluids*, vol. 44, no. 5, pp. 831–842, 2008.

[59] R. Theunissen, F. Scarano, and M. L. Riethmuller, "An adaptive sampling and windowing interrogation method in PIV," *Measurement Science and Technology*, vol. 18, pp. 275–287, jan 2007.

[60] M. Edwards and R. Theunissen, "Adaptive incremental stippling for sample distribution in spatially adaptive PIV image analysis," *Measurement Science and Technology*, mar 2019.

[61] A. Masullo and R. Theunissen, "On the applicability of numerical image mapping for PIV image analysis near curved interfaces," *Measurement Science and Technology*, vol. 28, no. 7, 2017.

[62] L. Gui, S. T. Wereley, and Y. H. Kim, "Advances and applications of the digital mask technique in particle image velocimetry experiments," *Measurement Science and Technology*, vol. 14, no. 10, pp. 1820–1828, 2003.

[63] O. Ronneberger, M. Raffel, and J. Kompenhans, "Advanced evaluation algorithms for standard and dual plane particle image velocimetry," *Proc. 9th International Symposium on Applied laser techniques to fluid mechanics*, pp. 13–16, 1998.

[64] J. Neyman, "Outline of a Theory of Statistical Estimation Based on the Classical Theory of Probability," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 236, no. 767, pp. 333–380, 1937.

[65] T. F. Chan, G. H. Golub, and R. J. Leveque, "Algorithms for Computing the Sample Variance : Analysis and Recommendations," *American Statistician*, vol. 37, no. 3, pp. 242–247, 2017.

[66] L. H. Benedict and R. D. Gould, "Towards better uncertainty estimates for turbulence statistics," *Experiments in Fluids*, vol. 22, no. 2, pp. 129–136, 1996.

[67] B. Efron and R. Tibshirani, "Bootstrap Methods for Standard Errors, Confidence Intervals, and Other Measures of Statistical Accuracy," *Statistical Science*, vol. 1, pp. 54–75, feb 1986.

[68] R. Theunissen, A. Di Sante, M. L. Riethmuller, and R. A. Van Den Braembussche, "Confidence estimation using dependent circular block bootstrapping: Application to the statistical analysis of PIV measurements," *Experiments in Fluids*, vol. 44, no. 4, pp. 591–596, 2008.

[69] A. Sciacchitano and B. Wieneke, "PIV uncertainty propagation," *Measurement Science and Technology*, vol. 27, no. 8, 2016.

[70] A. Sciacchitano, D. R. Neal, B. L. Smith, S. O. Warner, P. P. Vlachos, B. Wieneke, and F. Scarano, "Collaborative framework for PIV uncertainty quantification: comparative assessment of methods," *Measurement Science and Technology*, vol. 26, p. 074004, jul 2015.

[71] A. Sciacchitano, B. Wieneke, and F. Scarano, "PIV uncertainty quantification by image matching," *Measurement Science and Technology*, vol. 24, no. 4, 2013.

[72] F. Scarano, "Iterative image deformation methods in PIV," *Measurement Science and Technology*, vol. 13, jan 2002.

[73] R. van den Boomgaard and R. van Balen, "Methods for fast morphological image transforms using bitmapped binary images," *CVGIP: Graphical Models and Image Processing*, vol. 54, pp. 252–258, may 1992.

[74] R. Theunissen, F. Scarano, and M. L. Riethmuller, "Spatially adaptive PIV interrogation based on data ensemble," *Experiments in Fluids*, vol. 48, no. 5, pp. 875–887, 2010.

[75] H. Wendland, *Scattered Data Approximation*.
Cambridge University Press, dec 2004.

[76] W. Elsahhar, S. A. Showkat Ali, R. Theunissen, and M. Azarpeyvand, "An Experimental Investigation of the Effect of Bluff Body Bluntness Factor on Wake-vortex Noise Generation," in *2018 AIAA/CEAS Aeroacoustics Conference*,

(Reston, Virginia), American Institute of Aeronautics and Astronautics, jun 2018.

[77] K. Yu and J. Xu, "Adaptive PIV algorithm based on seeding density and velocity information," *Flow Measurement and Instrumentation*, vol. 51, pp. 21–29, 2016.

[78] M. Stanislas, K. Okamoto, and C. Kähler, "Main results of the First International PIV Challenge," *Measurement Science and Technology*, vol. 14, pp. R63–R89, oct 2003.

[79] M. Stanislas, K. Okamoto, C. J. Kähler, J. Westerweel, and F. Scarano, "Main results of the third international PIV Challenge," *Experiments in Fluids*, vol. 45, pp. 27–71, jul 2008.

[80] B. Wieneke and K. Pfeiffer, "Adaptive PIV with variable interrogation window size and shape," *Measurement Science and Technology*, vol. 18, pp. 275–287, 2008.

[81] D. Di Florio, F. Di Felice, and G. P. Romano, "Windowing, re-shaping and re-orientation interrogation windows in particle image velocimetry for the investigation of shear flows," *Measurement Science and Technology*, vol. 13, no. 7, pp. 953–962, 2002.

[82] R. D. Keane, R. J. Adrian, and Y. Zhang, "Super-resolution particle imaging velocimetry," *Measurement Science and Technology*, vol. 6, 1995.

[83] Y. G. Guezennec and N. Kiritsis, "Statistical investigation of errors in particle image velocimetry," *Experiments in Fluids*, vol. 10, no. 2-3, pp. 138–146, 1990.

[84] A. Susset, J. M. Most, and D. Honoré, "A novel architecture for a super-resolution PIV algorithm developed for the improvement of the resolution of large velocity gradient measurements," *Experiments in Fluids*, vol. 40, no. 1, pp. 70–79, 2006.

[85] D. P. Hart, "Super-resolution PIV by Recursive Local-correlation," *Journal of Visualization*, vol. 3, no. 2, pp. 187–194, 2000.

[86] F. Scarano, "Theory of non-isotropic spatial resolution in PIV," *Experiments in Fluids*, vol. 35, no. 3, pp. 268–277, 2003.

[87] F. Becker, B. Wieneke, S. Petra, A. Schröder, and C. Schnörr, "Variational adaptive correlation method for flow estimation," *IEEE Transactions on Image Processing*, vol. 21, no. 6, pp. 3053–3065, 2012.

[88] T. Astarita, "Analysis of interpolation schemes for image deformation methods in PIV: Effect of noise on the accuracy and spatial resolution," *Experiments in Fluids*, vol. 40, no. 6, pp. 977–987, 2006.

[89] A. Secord, W. Heidrich, and L. Streit, "Fast primitive distribution for illustration," *Computer Graphics Forum (Proceedings of the Eurographics Symposium on Rendering 2002)*, pp. 215–226, 2002.

[90] P.-O. Persson and G. Strang, "A Simple Mesh Generator in MATLAB," *SIAM Review*, vol. 46, no. 2, pp. 329–345, 2005.

[91] S. Beresh, "The Influence of Velocity Gradients on PIV Measurements of Turbulence Statistics: A Preliminary Study," no. 505, pp. 1–12, 2012.

[92] V. Sabri, A. Tahan, X. Pham, H. Radvar-Esfahlan, B. Louhichi, and A. M. Tahvilian, "Fixtureless Profile Inspection of Non-rigid Parts," 2013.

[93] R. Sibson, "A brief description of natural neighbor interpolation (chapter 2)," in *Interpreting Multivariate Data*, pp. 21–36, Wiley, 1981.

[94] M. Miozzi, "Particle Image Velocimetry using Feature Tracking and Delaunay tessellation," *12th International Symposium on Applications of Laser Techniques to Fluid Mechanics*, pp. 1–12, 2004.

[95] P. M. Lloyd, P. K. Stansby, and D. J. Ball, "Unsteady surface-velocity field measurement using particle tracking velocimetry," *Journal of Hydraulic Research*, vol. 33, no. 4, pp. 519–534, 2010.

[96] K. Ohmi and H. Y. Li, "Particle-tracking velocimetry with new algorithms," *Measurement Science and Technology*, vol. 11, no. 6, pp. 603–616, 2000.

[97] K. Takehara and T. Etoh, "A Study on Particle Identification in PTV - Particle Mask Correlation Method," *Journal of Visualization*, vol. 1, no. 3, pp. 313–323, 1999.

[98] R. Theunissen, – *Application to compressible flows and interfaces* –. 2010.

[99] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. C, no. 1, pp. 62–66, 1979.

[100] A. P. Garcia Sagrado, J. Van Beeck, P. Rambaud, and D. Olivari, "Numerical and experimental modelling of pollutant dispersion in a street canyon," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 90, no. 4-5, pp. 321–339, 2002.

[101] J. Westerweel, "Theoretical analysis of the measurement precision in particle image velocimetry," *Experiments in Fluids*, vol. 29, no. 7, pp. S003–S012, 2002.

[102] K. Yu, J. Xu, L. Tang, and J. Mo, "New adaptive sampling method in particle image velocimetry," *Measurement Science and Technology*, vol. 26, no. 3, p. 037002, 2015.

[103] A. Secord, "Weighted Voronoi stippling," *Proceedings of the second international symposium on Non-photorealistic animation and rendering - NPAR '02*, p. 37, 2004.

[104] R. Theunissen, J. S. Kadosh, and C. B. Allen, "Autonomous spatially adaptive sampling in experiments based on curvature, statistical error and sample spacing with applications in LDA measurements," *Experiments in Fluids*, vol. 56, no. 6, p. 116, 2015.

[105] L. Kocis and W. J. Whiten, "Computational Investigations of Low-Discrepancy Sequences," *ACM Transactions on Mathematical Software*, vol. 23, no. 2, pp. 266–294, 1997.

[106] Q. Cai and X. Yang, "Multiwalled Carbon nanotubes/hydroxyapatite nanoparticles incorporated GTR membranes," *Emerging Nanotechnologies in Dentistry: Second Edition*, pp. 181–209, 2017.

[107] D. A. Field, "Qualitative measures for initial meshes," *International Journal for Numerical Methods in Engineering*, vol. 47, no. 4, pp. 887–906, 2000.

[108] R. Franke, "A critical comparison of some methods for interpolation of scattered data," tech. rep., 1980.

[109] I. Ascencio-Lopez, O. Meruvia-Pastor, and H. Hidalgo-Silva, "Adaptive Incremental Stippling using the Poisson-Disk Distribution," *Journal of Graphics, GPU, and Game Tools*, vol. 15, no. 1, pp. 29–47, 2011.

[110] W. Huang and T. Ye, "Greedy vacancy search algorithm for packing equal circles in a square," *Operations Research Letters*, vol. 38, no. 5, pp. 378–382, 2010.

[111] K. J. Nurmela and P. R. Östergård, "Packing up to 50 equal circles in a square," *Discrete and Computational Geometry*, vol. 18, no. 1, pp. 111–120, 1997.

[112] D. W. Boll, J. Donovan, R. L. Graham, and B. D. Lubachevsky, "Improving dense packings of equal disks in a square," *Electronic Journal of Combinatorics*, vol. 7, no. 1 R, pp. 1–9, 2000.

[113] K. He, M. Dosh, and S. Zou, "Packing Unequal Circles into a Square Container by Partitioning Narrow Action Spaces and Circle Items," 2016.

[114] J. E. Higham, W. Brevis, and C. J. Keylock, "A rapid non-iterative proper orthogonal decomposition based outlier detection and correction for PIV data," *Measurement Science and Technology*, vol. 27, no. 12, 2016.

[115] L. Feng, J. Wang, H. Wang, Q. Gao, and R. Wei, "Proper orthogonal decomposition based outlier correction for PIV data," *Experiments in Fluids*, vol. 56, no. 2, pp. 1–15, 2017.

[116] A. Sciacchitano, "Uncertainty quantification in particle image velocimetry," *Measurement Science and Technology*, vol. 30, p. 092001, sep 2019.

[117] R. Theunissen, F. Scarano, and M. L. Riethmuller, "On improvement of PIV image interrogation near stationary interfaces," *Experiments in Fluids*, vol. 45, no. 4, pp. 557–572, 2008.