High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

Roth, Adrian

2023

Citation for published version (APA):
Roth, A. (2023). *High-speed 3D imaging of liquid jets, surfaces and respiratory droplets*. [Doctoral Thesis (compilation), Department of Physics]. Department of Physics, Lund University.

High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

ADRIAN ROTH
DEPARTMENT OF PHYSICS | FACULTY OF ENGINEERING | LUND UNIVERSITY

High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

by Adrian Roth

LUND
UNIVERSITY

Thesis for the degree of Doctor of Engineering
Thesis advisors: Dr. Edouard Berrocal and Dr. Elias Kristensson
Faculty opponent: Dr. Daniel Schanz

To be presented, with the permission of the Faculty of Engineering at Lund University, for public criticism in the Rydberg hall at the Department of Physics on Friday, the 20th of October 2023 at 09:15.

Organization
LUND UNIVERSITY

Department of Physics
Box 118
SE–221 00 Lund
Sweden

Document name
DOCTORAL DISSERTATION

Date of disputation
2023-10-20

Sponsoring organization

Author(s)
Adrian Roth

Title and subtitle
High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

Abstract
Sprays are commonly found in, among other, combustion, agriculture and food processing. For each of these applications, the understanding of spray liquid dynamics is crucial for optimization of efficiency, accuracy, and robustness of the spray-system in use. Sprays are also found as a collection of respiratory droplets ejected when people are speaking, yelling, coughing etc. that is one of the main transmission routes for viral disease in the recent COVID-19 pandemic. The experimental research performed on these sprays is often in 2D and not seldom on average data. However, the spray dynamics of interest acts in 3D space, during very short timescales and are stochastically unique. Here, instantaneous high-speed 3D imaging is required to fully characterize these events.

This thesis applies and analyses three different laser-based instantaneous high-speed 3D imaging techniques on three different liquid dynamics. These include, (1) volumetric Laser Induced Fluorescence (LIF) imaging of liquid jets, (2) LIF structured illumination for surface 3D reconstruction of a liquid hollow cone sheet and (3) stereoscopic particle tracking velocimetry of respiratory droplets. The volumetric imaging was found to be challenging because of refractive effects at the liquid-air interface. The structured illumination 3D reconstruction technique managed to reconstruct a transient 3D event where liquid breakups, ruptures, surface-waves, and ejection angles were extracted. Simulations found that the used reconstruction was accurate to below 1% of the structure and could resolve small surface waves with a height up to 65% of the theoretical limit. Finally, the stereoscopic imaging extracted 3D tracks of respiratory droplets with found experimental average speed uncertainties around 0.3 m/s. In addition, this experiment enabled simultaneous estimation of speed and size of respiratory droplets that give valuable information on the risks of disease spreading.

The presented instantaneous high-speed 3D reconstruction techniques can provide data that paves the way towards a deeper understanding of liquid dynamics in general and sprays in particular. The data is advantageous partly since it can be directly applied by modellers to improve and validate their simulations. In the future, both more validation and application of the presented techniques are required which is enabled by the open-sourced software and data that this thesis provides.

Key words
Laser Induced Fluorescence; Volumetric imaging; Structured Illumination; 3D Reconstruction; Stereoscopy; 3D Particle Tracking Velocimetry; COVID-19; sprays; respiratory droplets;

Classification system and/or index terms (if any)

Supplementary bibliographical information

Language
English

ISSN and key title

ISBN
978-91-8039-803-9 (print)
978-91-8039-804-6 (pdf)

Recipient's notes

Number of pages
217

Price

Security classification

I, the undersigned, being the copyright owner of the abstract of the above-mentioned dissertation, hereby grant to all reference sources the permission to publish and disseminate the abstract of the above-mentioned dissertation.

Signature

Date 2023-09-04

DOKUMENTDATABLAD enl SIS 61 41 21

High-speed 3D imaging of liquid jets, surfaces and respiratory droplets

by Adrian Roth

LUND
UNIVERSITY

Cover illustration front: A merged illustration of all three experimental setups applied in this thesis. On the right is the liquid jet experiment, on the left is the fringe projection onto a hollow cone liquid sheet and in the centre is a person ejecting droplets through an illumination volume. In the foreground are also four particles with illustrations of Corona viruses inside.

Cover illustration back: The so-called virusship that is unrealistic in respect to how viruses actually spread but illustrates the similarities between viruses and pirates in an ironic way. One of the virus pirates is saying ”Psst, note that no Large Language Models were used to write this thesis!”. This refers to artificial intelligence text generators that are the basis of chatbots such as ChatGPT and Bard.

Funding information: The thesis work was financially supported by the Swedish Research Council (project no. 2016-03894 and 2021–04542) and the Crafoord Foundation (project no. 20210942)

Faculty of Engineering, Department of Physics

ISBN: 978-91-8039-803-9 (print)
ISBN: 978-91-8039-804-6 (pdf)
LRCP: LRCP-247
ISSN: 1102-8718
ISRN: LUTFD2/TFCP-247-SE

Printed in Sweden by Media-Tryck, Lund University, Lund 2023

Dark and difficult times lie ahead. Soon we must all face the choice between what is right and what is easy.

- Albus Dumbledore

Asking for help is the opposite of failure. It is the only way to succeed.

- Beatrix Franklin

Contents

Author's contributions

List of publications

This thesis is based on the following publications:

I **A. Roth**, D. Frantz, W. Chaze, A. Corber, and E. Berrocal
High-speed imaging database of water jet disintegration Part I: Quantitative imaging using liquid laser-induced fluorescence
International Journal of Multiphase Flow, vol. 145, p. 103641 (2021).
`https://doi.org/10.1016/j.ijmultiphaseflow.2021.103641`

II **A. Roth**, E. Kristensson, and E. Berrocal
Snapshot 3D reconstruction of liquid surfaces
Optics Express, vol. 28, no. 12, p. 17906 (2020).
`https://doi.org/10.1364/OE.392325`

III **A. Roth**, M. Sapik, E. Kristensson, J. Jedelsky, and E. Berrocal
Analysis of liquid surface deformation and breakups using three-dimensional high-speed data recorded with a single camera
Physics of Fluids, vol. 34, p. 123324 (2022).
`https://doi.org/10.1063/5.0130498`

IV **A. Roth**, M. Stiti, D. Frantz, and E. Berrocal
High-speed scattered-light imaging for sizing respiratory droplets
Journal of Aerosol Science, vol. 174, p. 106257 (2023).
`https://doi.org/10.1016/j.jaerosci.2023.106257`

V **A. Roth**, M. Stiti, D. Frantz, A. Corber, and E. Berrocal
Exhaled aerosols and saliva droplets measured in time and 3D space: Quantification of pathogens flow rate applied to SARS-CoV-2
Natural Sciences, vol. 3, no. 3 (2023).
`http://doi.org/10.1002/ntls.20230007`

All papers are reproduced with permission of their respective copyright agreement.

Related Work

vi A. Karlsson, S. Török, **A. Roth**, and P-E. Bengtsson
Numerical scattering simulations for estimating soot aggregate morphology from nephelometer scattering measurements
Journal of Aerosol Science, vol. 159, p. 105828 (2022).
https://doi.org/10.1016/j.jaerosci.2021.105828

vii S. Ek, V. Kornienko, **A. Roth**, E. Berrocal, and E. Kristensson
High-speed videography of transparent media using illumination-based multiplexed schlieren
Scientific reports, vol. 12, no. 1, p. 19018 (2022).
https://doi.org/10.1038/s41598-022-23198-6

viii D. Sanned, J. Lindström, **A. Roth**, M. Aldén, and M. Richter
Arbitrary position 3D tomography for practical application in combustion diagnostics
Measurement Science and Technology, vol. 33, no. 12 p. 125206, (2022).
https://doi.org/10.1088/1361-6501/ac92a1

ix D. Sanned, S. Nilsson, **A. Roth**, E. Berrocal, A. Ehn, and M. Richter
3D-Tomographic Reconstruction of Gliding Arc Plasma
Applied Physics Letters, vol. 123, no. 7, p. 071104 (2023).
https://doi.org/10.1063/5.0161361

Conference contributions

x **A. Roth**, and E. Berrocal
3D surface reconstruction of liquid structures in sprays using structured illumination and phase demodulation
ILASS–Europe 2019, 29th Conference on Liquid Atomization and Spray Systems, 2-4 September 2019, Paris, France
Oral presentation and conference proceedings paper

xi **A. Roth**, M. Sapik, J. Jedelsky, and E. Berrocal
High-speed 3D imaging of liquid sheet surfaces using the FP-LIF technique
ICLASS 2021, 15th Triennial International Conference on Liquid Atomization and Spray Systems, Edinburgh, UK, 29 Aug. - 2 Sept. 2021
Oral presentation and conference proceedings paper

xii **A. Roth**, M. Stiti, A. Matamis, D. Frantz, M. Richter, M. Aldén and E. Berrocal
Analysis of coughed droplets using stereoscopic high-speed imaging
ICLASS 2021, 15th Triennial International Conference on Liquid Atomization and Spray Systems, Edinburgh, UK, 29 Aug. - 2 Sept. 2021
Oral presentation and conference proceedings paper

xiii **A. Roth**, M. Stiti, D. Frantz, and E. Berrocal
Stereoscopic high-speed imaging for 3D tracking of coughed saliva droplets in the context of COVID-19 spreading
20th International Symposium on Application of Laser and Imaging Techniques to Fluid Mechanics - Lisbon, Portugal, July 11-14, 2022
Oral presentation and conference proceedings paper

Popular science contributions

PhD students in Teaching (Doktorander i lärande)
2018-10-17 and 2019-03-20, Ängelholm, Sweden
Presentations and interactions with pupils to show what a PhD student does on a daily basis.

Culture Night (Kulturnatten) - Open house
2019-09-21, Lund, Sweden
Developed and was in charge of the activity *Take a 3D-selfie*
Using the fringe projection 3D reconstruction described in this thesis together with the general public.

The Research Tour (Forskarturnén)
2019-09-12 and 2019-11-07, Lund, Sweden
Visiting two libraries to present my research.

Researchers' Grand Prix, Science Slam
2022-10-13, Helsingborg, Sweden
A competition where you are challenged to present your research in the most captivating, inspiring and educational way possible – in just four minutes!

Acknowledgements

Five years have passed and I have a bunch of people to thank for all of my achievements. But I am running out of time so I will keep this short. First off, I would like to thank Per-Erik Bengtsson for being the reason that I ended up at the division, guiding me through my Master's project and then encouraging me in the work I have been doing for the division. My supervisors, Edouard Berrocal and Elias Kristensson, thank you for all your help and guidance whenever I needed it that have made me the scientist I am today.

Thanks to all awesome colleagues that I have had the luxury of working with! For all the work that enabled my research I thank Minna Ramkull, Emelie Nihlén, Cecilia Bille and Igor Buzuk. You are essential! A special thanks to David Frantz, Joakim Jönsson and Mehdi Stiti. I have had great collaborations with all of you. Thanks to Henrik Feuk for guiding me through the stormy seas that is the thesis admin stuff. Thanks to David Sanned and Sebastian Nilsson for letting me join an awesome project. Yue Qiu, Zhiyong Wu, David Sanned and Sabrina Gericke, I just love all the time we have put into awesome/weird trivselgruppen activities. Danke Sabrina für deine Hilfe mit meine Deutsch und alle andere schöne stunden im Büro und im Erdbeeren Land. Ich hoffe dass unsere Deutsch und Schwedisch fika geht weiter.

Thanks to Erik, Johan, Ludwig, Ulrica, Kajsa, Samuel and Nelly for just being awesome people to hang around.

At last I would like to thank my family:

Mamma och Pappa, ni är mina klippor,
Anton, du är min förebild,
Sara och Olof, ni är mitt syfte!

Popular summary

On November 12, 1732, a man named Henri Pitot showed his new invention to the Royal Academy of Sciences in Paris. He showed two tubes. One tube was straight while the other was shaped like an L. What was it supposed to do? It was developed to measure the flow speed of water in rivers. At the time, the standard methodology was to put an object (typically an orange for its excellent floating abilities) in the river and observe its speed. It worked like a charm but there was a fundamental limitation. One could only measure the flow speed at the surface. Pitot's tube worked in a unique way that made it possible to measure the velocity at different depths in the river. Suddenly a new dimension was available. With his new measurement device, Pitot challenged the prominent theory at the time that flow speed in a river increase with depth. He correctly found that the opposite was the case. This basic knowledge of fluid dynamics was discovered because of the groundbreaking measurement tool.

This thesis has quite some similarities with the novelty of the tube. Here, measurement techniques are presented that unlocks another dimension of liquids. This other dimension is the third dimension. The third dimension that we are all living in, the third dimension that you perceive in the distance, the third dimension in which all liquid phenomena take place. Currently many measurements on liquids use imaging which is a 2D measurement approach that has given significant insight into fluid dynamics. However, it can never give the complete picture since, as mentioned, all liquid phenomena take place in 3D. Therefore, 3D measurements are important. I present three different kind of 3D measurements in this thesis that have been performed on three different liquid phenomena: a water jet, a hollow cone water sheet and respiratory droplets. The first two are phenomena with major industrial applications. They are both closely connected to sprays and sprays are used in more places than you might think. Examples are painting, cooling, food processing, firefighting, combustion engines, and more. An improved understanding of the water jet and hollow cone water sheet can give a higher precision and efficiency for both applications in jets and sprays.

The third phenomenon was respiratory droplets who are connected to the way too well known COVID-19 pandemic. We can all probably say where we were when the world shut down (probably at home learning how to use zoom). To lower the risk of overfull hospitals, it was important to delay the spreading of the virus. The first suspect of this spreading was direct contact between people and contaminated surfaces such as keyboards and measures were taken to lower the probability of spreading the virus in this way. Later, research suggested that this path was not as probable as first suggested since a virus cannot survive on surfaces long enough. The next spreading suspect was then the infamous aerosols. Aerosols are small particles that can float for a long time in the air to then be inhaled and infect when containing viruses. In the fall of 2020, we wanted to help so we employed our laboratory skills in spray imaging to extract information of 3D velocity and size of these small droplets. This is essential information in understanding what measures should be taken to prevent the spreading of the virus.

In this thesis I present the 3D measurement techniques we have developed that are unlocking the third dimension of all these phenomena. Just like the Pitot tube did in its time, I hope that these measurement techniques will produce knowledge previously beyond our grasp and both improve sprays for specific applications and our ability to fight future pandemics.

Populärvetenskaplig sammanfattning

Den 12e november 1732 visade en man vid namn Henri Pitot sin nya uppfinning för den Kungliga franska vetenskapsakademin i Paris. Han visade två rör. Det ena röret var rakt medan det andra var format som ett L. Vad var poängen med uppfinningen? Jo, den utvecklades för att mäta flödeshastigheten av vatten i floder. På den tiden var det standard att mäta flöde genom att placera ett föremål (vanligtvis en apelsin för dess utmärkta flytförmåga) i floden och observera dess hastighet. Det fungerade bra men det fanns en grundläggande begränsning. Man kunde bara mäta flödeshastigheten vid ytan. Pitots rör fungerade på ett annat sätt som gjorde det möjligt att mäta hastigheten på olika djup i floden. Plötsligt öppnades en ny dimension. Med sin nya mätanordning utmanade Pitot den då framstående teorin att flödeshastigheten i en flod ökar med djupet. Han fann, med rätta, att det motsatta var fallet. Denna mycket grundläggande kunskap om vätskedynamik upptäcktes som ett resultat av det banbrytande mätverktyget.

Denna avhandling har en hel del likheter med Pitot's rör. Jag presenterar här mättekniker som låser upp ytterligare en dimension av vätskor. Denna dimension är den tredje dimensionen. Den tredje dimensionen som vi alla lever i, den tredje dimensionen som du uppfattar i fjärran, den tredje dimensionen där alla vätskefenomen äger rum. För närvarande appliceras främst 2D mätningar på vätskor som har gett betydande insikter i vätskedynamik. Det kan dock aldrig ge den fullständiga bilden eftersom alla vätskefenomen som sagt sker i 3D vilket gör 3D-mätningar viktiga. Jag presenterar tre olika typer av 3D-mätningar i denna avhandling som har utförts på tre olika vätskefenomen: en vattenstråle, en vätskeyta formad som en ihålig kon och respiratoriska droppar. De första två är fenomen med stora industriella tillämpningar. De är båda nära förbundna med sprayer som används på fler ställen än man kan tro. Exempel är målning, kylning, livsmedelsförädling, brandbekämpning, förbränningsmotorer med mera. En förbättrad förståelse av vattenstrålen och den ihåliga konens vattenskiva kan ge en högre precision och effektivitet för både applikationer i jetstrålar och sprayer.

Det tredje fenomenet var respiratoriska droppar som är kopplade till den alldeles för välkända COVID19-pandemin. Vi kan nog alla säga var vi var när världen stängdes av (mest troligt hemma försökte förstå sig på zoom). För att minska risken för överfulla sjukhus var det viktigt att fördröja spridningen av viruset. Den första misstänkta för spridningen var direktkontakt mellan människor och förorenade ytor som exempelvis tangentbord och styrande vidtog åtgärder för att minska sannolikheten för att viruset skulle spridas på detta sätt. Senare antydde forskning att denna väg inte var så trolig som först antyddes eftersom virus inte kan överleva på ytor tillräckligt länge. Nästa misstänkt för spridning en var då de ökända respiratoriska dropparna. Dessa droppar skjuts ut när vi pratar, hostar eller liknande och de kan sväva länge i luften där de sedan kan andas in och infektera en annan människa. Hösten 2020 ville vi hjälpa till att förstå denna spridningsväg så vi använde våra laboratoriekunskaper i spraybildbehandling för att extrahera information om 3D-hastighet och storlek på dessa små droppar. Hastighet och storlek är viktig information för att förstå vilka åtgärder som bör vidtas för att förhindra spridning av viruset.

I denna avhandling presenterar jag de tre 3D-mättekniker vi har utvecklat som låser upp den tredje dimensionen av alla dessa fenomen. Jag hoppas att dessa mättekniker kommer att producera kunskap tidigare utom vårt räckhåll, precis som Pitot-röret gjorde, och både förbättra sprayer för specifika applikationer och vår förmåga att bekämpa framtida pandemier.

Abbreviations

2D	Two Dimensional
3D	Three Dimensional
APS	Aerodynamical Particle Sizer
BFS	Breadth First Search
COVID-19	CoronaVirus Disease 2019
CT	Computed Tomography
CWT	Continuous Wavelet Transform
DOE	Diffractive Optical Element
FFBE	Four Frame Best Estimate
FM	Figure of Merit
FP-LIF	Fringe Projection - Laser Induced Fluorescence
GMRF	Gaussian Markov Random Field
HWR	Height/Width Ratio
LIF	Laser Induced Fluorescence
PIV	Particle Image Velocimetry
PTV	Particle Tracking Velocimetry
RMSE	Root Mean Square Error
SARS-CoV-2	Severe Acute Respiratory Syndrome CoronaVirus 2
VLIF	Volumetric Laser Induced Fluorescence

Chapter 1

Introduction

The ever-present liquid dynamics in nature and the importance of liquid dynamics to civilization cannot be understated. A liquid dynamic that is both practically important and one of the most challenging to study is spray formation. In nature, sprays can be found in for example waterfalls, clouds, fog and rain. Artificial sprays can be found in fuel injection systems for automobiles and aircraft, pesticide and herbicide delivery in agriculture, pharmaceutical delivery, drying in food processing or painting in manufacturing to name a few [1]. A spray is a collection of small spherical droplets distributed in 3D space and characterized by various parameters. These include the number density of droplets as well their size and velocity distributions. Different spray applications have unique optimal characteristics of the spray and deviations from this optimum can greatly reduce the effectiveness and increase environmental contamination. For example, too small droplet sizes in agriculture pesticide spray increase the drift of pesticides to areas beyond the crops that comes with a negative impact on residents and ecosystems [2]. To produce the optimal spray for each application, a detailed understanding of the underlying processes that produce the spray is essential. This understanding will assist in predicting what spray-system and boundary parameters are likely to create the sought characteristics of the spray.

Another spray example are the droplets exhaled when people are active in respiratory events such as speaking, yelling, singing, coughing, and sneezing. These respiratory droplets have been found to be one of the main drivers in the spread of COVID-19 via the virus SARS-CoV-2 [3]. The pandemic has had a substantial impact on societies all over the world, partly due to the millions of fatalities caused by the disease, but mainly through the massive restrictions that were put in place to limit the spreading of the virus. Initially, it was thought that the spreading of COVID-19 mainly went through direct contact between people or indirect via surfaces that were contaminated. However, the fast spreading together with research on how the surfaces were unlikely to host viruses for longer periods of times led to the conclusion that small respiratory droplets who can endure in the air for longer periods of time was the more probable spreading agent [4, 5]. The increased understanding of the production and spreading characteristics of these droplets thereafter became a high priority. In the context of spreading disease, the relevant characteristics of respiratory droplets are their quantity, velocity, ejection-angles, and size.

The understanding of liquid dynamics, such as sprays and respiratory droplets, require detailed experimental work that can discover new phenomena and provide accurate data for liquid dynamic modelling research. In the modelling, this data is typically used to validate and improve simulation models. These simulation models are then used to explore the often large parameter spaces for specific applications that aid in the development of improved spray apparatus or, in the context of COVID-19, in what restrictions are efficient. There are, however, multiple challenges in the experimental characterization of liquid dynamics that are obstacles on the path towards fully understanding these processes. The main challenges are:

- The deformations and breakups of liquid bodies are intrinsically stochastic where each event is unique as a result of chaotic creation of instabilities and turbulence.
- The turbulence processes found in liquid flows injected at relatively high pressures, are occurring at very short time scales and are therefore hard to capture. Typically, high-speed cameras of kHz frame rates are used even though faster recording would be preferred. Cameras with MHz recording rate exist, but they can only record a few hundred frames that corresponds to short periods of time. In addition, such cameras are expensive and therefore less accessible.
- The visualization of liquid structures is usually performed using a diffuse white light or a collimated laser beam illumination. As many liquids are transparent, photons are mainly refracted and specularly reflected from the surfaces and complex light effects are created. Typical effects include light focusing and beam trapping. In the case of spherical curvature, commonly observed with drops, a strong focusing occurs that leads to the appearance of glare points which does not correctly represent the liquid shape.
- Unwanted artefacts leading to some image blurs are occurring when photons interact multiple times with several droplets, prior to image detection. This is called multiple light scattering which is significant for liquids injection systems, such as atomizing sprays, that create a droplet cloud of high density. Light that is sent through a dense spray is scattered multiple times, creating a blur of the imaged object, just like the situation of imaging through a fog.

The use of experimental data for validation and improvement of simulation models is also challenging which is partly caused by how experiments and simulations have a gap in output quantities. The output of experiments are often quantities such as scattering intensity, Laser Induced Fluorescence (LIF) intensity and extinction. In simulation models the quantities are instead 3D positions, sizes, and velocities. One can bridge this gap by converting the simulation quantity to the experimental one that enable direct comparison [6], however, this quantity gap remains to be an obstacle on the path towards understanding liquid processes [1].

Laser measurement techniques are most commonly employed to characterize liquid processes. These include point measurement techniques such as Phase Doppler Anemometry (PDA) [7] that can simultaneously measure droplet size and velocity. Such point measurements require scanning to extract spatial variation in the liquid phenomenon. To avoid scanning, imaging techniques have been extensively used that give 2D information. Shadowgraphy [8] is a basic technique where a background illumination is obstructed by the liquid and the imaged shadow give information of the liquid structure. Schlieren measurements [9] are employed to image gradients of refractive index typically used in hot air flow measurements but can also be used for analysing liquid fuel combustion.

Planar Laser Imaging have been extensively used since it can extract information from cross-section slices of the liquid. It can be used to extract fluid flow velocities using Particle Image Velocimetry (PIV) [10, 11] and Particle Tracking Velocimetry (PTV) [12]. Both techniques can be applied in the same setting where droplets or tracer particles are tracked over two or more frames. Planar laser imaging has also been extensively applied with Lorenz-Mie scattering and LIF [1]. The LIF/MIE ratio of such images is often employed to estimate droplet size (Sauter Mean Diameter) [13]. These planar laser techniques are all affected by multiple scattering for larger droplet densities and some advanced techniques can reduce these effects. One such technique is called Structured Laser Illumination Planar Imaging (SLIPI) that create a spatially modulated intensity pattern that look like the teeth on a comb. With post-processing, the single scattered or fluorescence intensity can be extracted from the blurred images [14, 15].

Measurements in the 2D domain can only increase our understanding of liquid processes up to a limit. For full characterization, 3D measurements are required where there are fewer examples found in the literature. A common 3D measurement approach is to use Computed Tomography (CT). Then, multiple views (typically ~ 10 or more) are used and a line-of-sight inversion is employed to extract 3D information. CT examples applied on sprays have used visible light and SLIPI to reduce multiple scattering [16], and x-rays [17]. 3D can also be produced through stacking of multiple 2D slices [18]. All these examples use average imaging to improve signal to noise ratio in the recorded data.

Averaged data can give general liquid properties, but the stochastic nature of these liquid phenomena require instantaneous measurements to be fully understood. In addition, high-speed (>kHz) temporal resolution is required to better resolve complex 3D flow features such as turbulence and small surface waves. Examples that fulfil these requirements use CT with X-rays on spray where multiple high-speed cameras are employed [19]. Plenoptic imaging has been applied to droplet fields where an array of microlenses is added in front of the camera [20, 21]. The array enables post-processing to refocus the images and particle locations are estimated by finding the refocusing distances where objects are optimally in focus. Holography has been applied on a swirl spray that can reconstruct object 3D location and shape from the interference pattern created when a laser interacts with droplets [22]. High-speed shadowgraphy imaging has been used on a dilute spray where the degree of droplet blurring due to out of focus is used to estimate the 3D position [23]. Shape from refraction was applied on short wind waves where a pattern is placed at a known location below a liquid surface. The distortion of this pattern caused by refraction of light is then used to estimate the surface topology [24]. 3D-PTV has also been employed with multiple views to resolve turbulence of a water jet injected into a liquid environment impinging on a cylinder [25]. Here, the advanced 3D-PTV technique called Shake-The-Box is used that enables tracking of high particle densities [26]. The high particle density is of great importance because of the improved spatial resolution and increased details of complex flow features it enables. The research presented in this paragraph are limited to specific working conditions. Either many high-speed cameras are required as in tomography, or the 3D reconstruction is only possible for separated droplets which is the case for the plenoptic imaging and digital off-axis holography.

Respiratory droplets have also been analysed through experiments where used techniques largely overlap with the previous examples. Velocity measurements of respiratory droplets have often been indirect by means of measuring air flow. Examples include the spirometer, a point measurement device that estimate air volume per unit time [27]. Other work have used schlieren imaging [28, 29, 30], PIV [31, 32, 33, 34, 35, 36, 37, 38] and CO2 imaging [39]. Subjects that exhale smoke, or flour have also been applied in measurements to follow air flow [27, 40, 41, 42]. Measurements on the actual droplets have among other been performed with shadowgraphy [43, 38], and droplet scattering of laser light [42, 44, 45, 46]. To measure the size and concentration of droplets there are a few common apparatus' that have been used. The Aerosol Particle Sizer (APS) is a point measurement device that size particles through a time of flight measurement [47, 48, 49, 50, 51, 52] which have a a sizing range of 0.5 to 20 µm. Sizing techniques using scattering intensity in point measurement devices, including different commercial devices such as the Optical Particle Sizer (OPS), is also commonly used [53, 54, 55, 56] with size ranges around 0.3 to 10 µm. The Scanning Mobility Particle Sizer (SMPS) is a device that can measure very small droplet sizes between 0.01 and 0.4 µm [53, 54, 52]. Not seldom is the combination of devices with different sizing ranges used to be able to detect a larger span of particle sizes. Imaging approaches for sizing respiratory particles have also been used where shadowgraphy [57], Interferometric Laser Imaging for Droplet Sizing (ILIDS) [32], holography [53, 58] can be found. In the context of respiratory droplets, 3D measurements are rare. However, Shake The Box has been used in a 3D-PTV measurement to track the air flow by means of helium filled soap bubbles inside a room where synthesized breathing and thermal heat convection flow is created by a manikin [59].

The aim of this thesis is to apply and validate new combinations of optical techniques that enable instantaneous high-speed 3D measurements of liquid phenomena. The methodology to achieve this aim is focused on the image post-processing to extracts 3D information where both estimation performance and computational optimization is important. The new use of these techniques will diversify the toolbox of which liquid phenomena can be imaged in 3D. In addition, the measurement techniques will provide experimental data directly compatible for validation, tuning and development of liquid simulation models. The thesis contains three different 3D imaging techniques applied to three different liquid phenomena each with its own specific aim.

- The aim of the first technique is to estimate liquid thickness from LIF intensity applied on water jets (Chapter 2 and Paper I).
- The aim of the second technique is to use structured illumination and a single camera view to 3D reconstruct the surface of a hollow cone liquid sheet (Chapter 3 and Papers II and III).
- The aim of the third technique is to use stereoscopic PTV to find 3D velocities, sizes and number of respiratory droplets ejected when speaking, yelling and coughing (Chapter 4 and Papers IV and V).

The chapter for each of these three imaging techniques is outlined similarly. The first section describes the liquid phenomenon that is followed by a description of the 3D imaging approach. Thereafter, the experimental setup and image post-processing routines are detailed. When applicable, there are so-called ”Software boxes” that informs the reader of python scripts, modules and functions connected to the described post-processing and analysis of data. Each chapter ends with results and analysis of the measurement technique. After the chapters on the techniques, Chapter 5 summarises this thesis together with an outlook of future considerations.

Chapter 2

High-speed volumetric imaging of liquid jets

2.1 Characteristics of liquid jets

Liquid jets have applications in multiple areas such as cooling, cutting, cleaning, medicine, agriculture painting, food drying and delivery of fuel for combustion. The most common

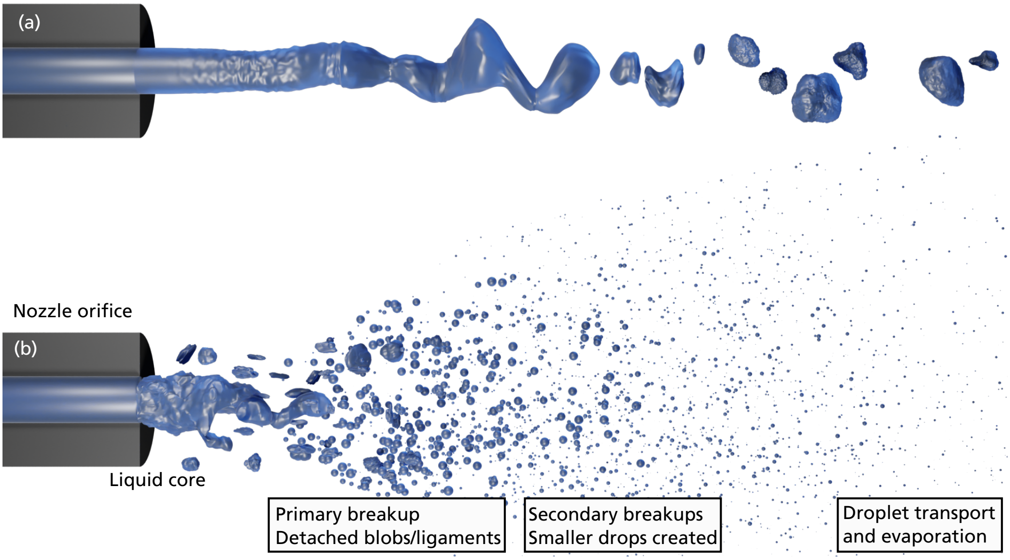


Figure 2.1: Liquid jet breakup. Two different liquid injection pressures are illustrated where (a) is relatively low and (b) is high. The higher pressure corresponds to faster breakup and smaller produced droplets.

application is internal combustion engines where the fuel is injected under high pressure into a combustion chamber to produce a spray of small droplets. The small fuel droplets evaporate and burn more efficiently and cleaner than their liquid counterpart.

Liquid jets are inherently unstable and will breakup [60]. Depending on the application, it is either advantageous to hasten or suppress this breakup process. Two parameters that largely affect the breakup is the geometry of the nozzle that injects the liquid and the injection pressure. Figure 2.1 illustrate two liquid jets with different injection pressure. The higher pressure gives relatively both faster breakups and smaller resulting droplets. The process of these breakups involves the following. Initially, the jet is continuous as a liquid core and instabilities are created further downstream from the nozzle. The instabilities grow until they eventually induce the primary breakup where large drops, blobs and ligaments are created. The created particles might interact through collision and merging but with large enough injection velocity, the so-called secondary breakups will eventually divide the liquid into small spherical droplets, a spray. Here, the regions involving primary and secondary breakups is commonly called the spray formation region which is followed by the spray region. Jet nozzles are commonly used in for example diesel injection systems where injection pressures of hundreds of bar are required to quickly vaporize the injected fuels before ignition [61, 62]. For high-pressure injection systems, one relevant effect to mention is cavitation that is connected to the design of the nozzle. Cavitation occurs when there is a local drop in pressure below the liquid vapor pressure. The vaporized liquid is then thought to induce ripples in the liquid that can hasten the breakup of the liquid core. The local drop of pressure is common when there are fast changes of the cross-section area for the liquid inside the nozzle.

2.2 Volumetric imaging approach

The fluorescence intensity emitted using LIF is connected to the concentration of the fluorescent molecule. By imaging the fluorescence from multiple views together with a proper calibration, the 3D variation of this concentration can be extracted, called Volumetric LIF or VLIF. VLIF has been used in gas flows to image for example turbulence and mixing processes [63] and it can also be used in high-speed measurements [64]. VLIF is here applied to liquid processes which has also been applied in the literature [65, 66].

To extract the volume of the liquid, a first step is to extract the liquid thickness d for an imaged pixel intensity. This liquid thickness is defined as the distance of liquid a beam would cross when sent out from the camera in the direction the camera is viewing. The model of imaged intensity I_{LIF} can be realised through the following equation plotted in

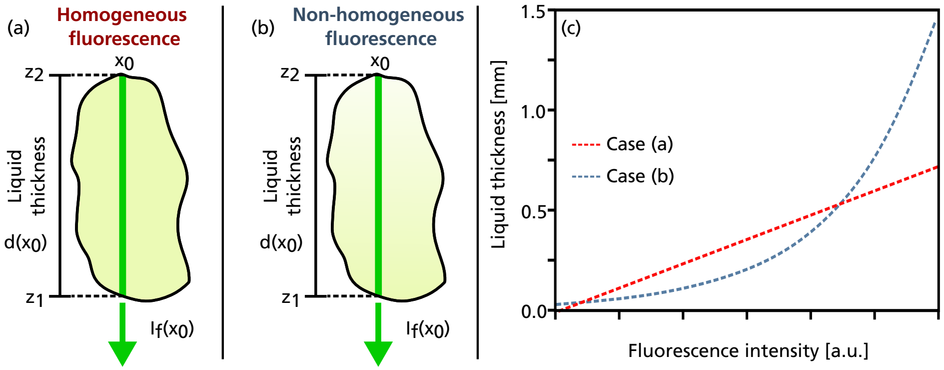


Figure 2.2: Simplified models of how the imaged fluorescence intensity is connected to liquid thickness. The models either assume a homogeneous fluorescence throughout the liquid body (a) or a non-homogeneous fluorescence (b). Curves of liquid thickness as a function of fluorescence intensity for both models are shown in (c). The function for model (a) and (b) is described by equation (2.1) and (2.5) in the text, respectively.

Figure 2.2(c),

$$I_{LIF} = d \cdot K_{\text{opt}} \cdot I^0_{\text{laser}} \cdot c \cdot \sigma_a(\lambda_{\text{laser}}) \int_{\lambda_1}^{\lambda_2} F(\lambda) d\lambda, \tag{2.1}$$

where K_{opt} is a constant taking the collection angle and the response of the camera in a single pixel into account, I^0_{laser} is the illumination intensity, c is the concentration of fluorescent dye in the liquid, $\sigma_a(\lambda)$ is the absorption cross-section probability of the dye at a specific wavelength and $F(\lambda)$ is the emission spectrum. The emission spectrum is integrated over the wavelengths that are transmitted by the bandpass filters. In this work, $\lambda_1 = 465.5$ nm and $\lambda_2 = 554.5$ nm. The resulting simplified model give a linear dependence between the fluorescent signal and liquid thickness d.

The two effects, attenuation of the laser and reabsorption of the fluorescence are not taken into account by equation (2.1) even though both effects are significant in this experiment. The attenuation of the laser is caused by the absorption of light by the fluorescence tracer as the laser light travels through the liquid. The laser intensity is here assumed to be relatively low compared to the concentration of fluorescent dye which means that the reduced laser intensity will also reduce the intensity of fluorescent light. This is also known as the fluorescence dye is unsaturated. The laser attenuation is described using the Beer-Lambert law and the extinction coefficient $\mu_e(\lambda_{\text{laser}}) = c\sigma_a(\lambda_{\text{laser}})$ where the laser intensity is reduced after travelling from the liquid interface z_1 to position z as,

$$I_{\text{laser}}(z) = I^0_{\text{laser}} e^{-\mu_e(\lambda_{\text{laser}}) \cdot (z - z_1)}. \tag{2.2}$$

Reabsorption of the fluorescent light is caused by a relatively high concentration of the fluorescent dye used in this experiment. The reabsorption will in a similar way to the laser attenuation reduce the probability of fluorescence deep in the liquid to reach the camera because of extinction as follows,

$$\frac{I_{\text{fluorescence}}(z)}{I_{\text{laser}}(z)\mu_e(\lambda_{\text{laser}})} = \int_{\lambda_1}^{\lambda_2} F(\lambda)e^{-\mu_e(\lambda)\cdot(z-z_1)}d\lambda, \tag{2.3}$$

where $I_{\text{fluorescence}}$ is the amount of fluorescent light that reach the camera for each position z.

The attenuation and reabsorption are illustrated in Figure 2.2(b) and can be modelled by updating equation (2.1) to,

$$I_{LIF} = K_{\text{opt}} \cdot \int_{z_1}^{z_2} I^0_{\text{laser}} e^{-\mu_e(\lambda_{\text{laser}})\cdot(z-z_1)} \cdot \mu_e(\lambda_{\text{laser}}) \int_{\lambda_1}^{\lambda_2} F(\lambda)e^{-\mu_e(\lambda)\cdot(z-z_1)}d\lambda dz. \tag{2.4}$$

Rearranging and solving the z integral,

$$\begin{aligned}
I_{LIF} &= K_{\text{opt}} \cdot I^0_{\text{laser}} \cdot \mu_e(\lambda_{\text{laser}}) \int_{\lambda_1}^{\lambda_2} F(\lambda) \int_{z_1}^{z_2} e^{-(\mu_e(\lambda_{\text{laser}})+\mu_e(\lambda))\cdot(z-z_1)} dz d\lambda \\
&= K_{\text{opt}} \cdot I^0_{\text{laser}} \cdot \mu_e(\lambda_{\text{laser}}) \int_{\lambda_1}^{\lambda_2} F(\lambda) \left[\frac{e^{-(\mu_e(\lambda_{\text{laser}})+\mu_e(\lambda))\cdot(z-z_1)}}{-(\mu_e(\lambda_{\text{laser}}) + \mu_e(\lambda))} \right]_{z_1}^{z_2} d\lambda \\
&= K_{\text{opt}} \cdot I^0_{\text{laser}} \cdot \mu_e(\lambda_{\text{laser}}) \int_{\lambda_1}^{\lambda_2} \frac{F(\lambda)\left(1 - e^{-(\mu_e(\lambda_{\text{laser}})+\mu_e(\lambda))\cdot d}\right)}{\mu_e(\lambda_{\text{laser}}) + \mu_e(\lambda)} d\lambda,
\end{aligned} \tag{2.5}$$

where $d = z_2 - z_1$. There is a clear diminishing return with increasing liquid thickness where eventually change in thickness will not be detected as change of intensity as is illustrated by the blue curve in Figure 2.2(c). There is from this model a clear limitation in how thick liquid one can measure using VLIF. Note that these equations in Paper I have for some reasons been jumbled so they are mostly wrong.

The model in equation (2.5) is useful for understanding the fluorescence signal as a function of liquid thickness, however, this model is still too simplified for these experiments. Firstly, the illumination is assumed to be in the same direction as the detection direction when it actually is at a relative 45 degree angle. In addition, there will be significant refraction effects and internal reflections at the irregular liquid-air interface. These complexities not included in the model make equation (2.5) not suitable to use for liquid thickness prediction. Instead, a non-parametric calibration curve will be extracted.

2.3 Description of the experimental setup

The experimental setup used for the VLIF experiments is illustrated in Figure 2.3 and briefly described below. More details can be found in Paper I.

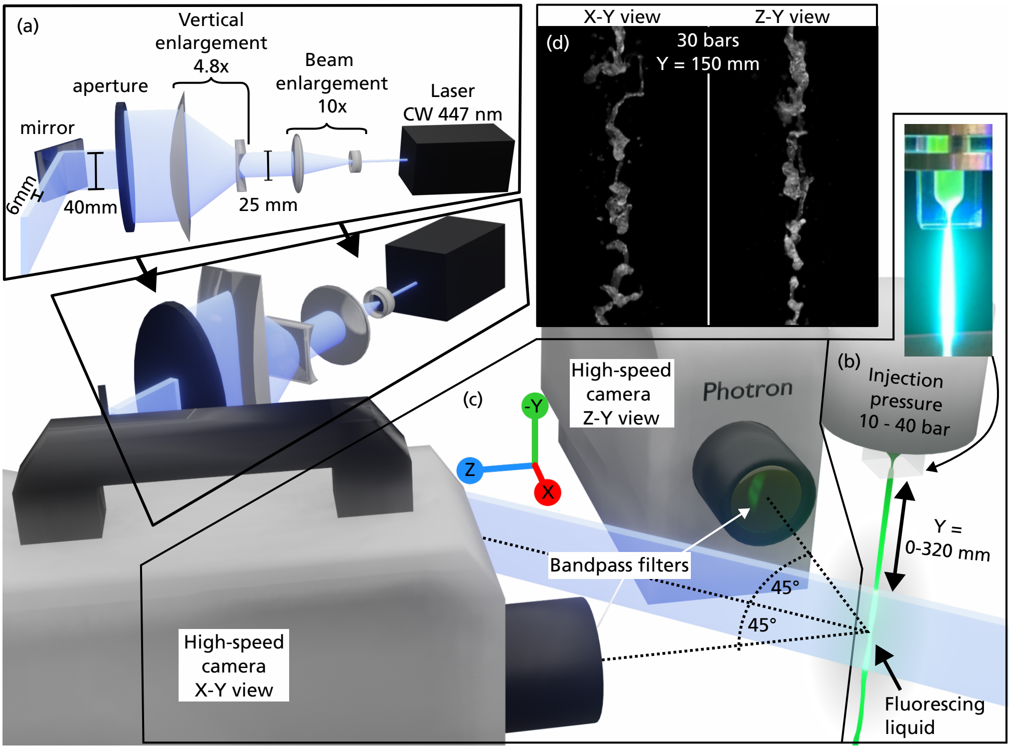


Figure 2.3: The experimental setup. The illumination (a), the injection system (b) and the high-speed cameras for imaging (c). Examples of recorded raw images are shown in (d).

(a) Illumination: In this work, a blue continuous wave laser at 447 nm wavelength and 4 Watt average power is shaped into an illumination volume that is 6 mm thick and 40 mm. The shape is chosen to verify that all liquid that is imaged. Care was taken to make a homogeneous top-hat illumination profile using the enlargement optics and aperture, but a post-processing correction is also performed, detailed in section 2.4.1.

(b) Liquid: A liquid jet is produced by a custom-made nozzle built for this experiment. This nozzle is designed to minimizes complex flow features inside and at the exit of the nozzle such as non-uniform exit velocity, turbulence inside the nozzle and cavitation. Such a nozzle will simplify modelling of this experiment. The nozzle was realized by applying methods for designing wind-tunnel contraction sections. The created nozzle has a liquid chamber with the shape of a wine glass where the foot of the glass is the exit point of the liquid. The nozzle is made from transparent acrylic that enables inspection of the liquid inside the nozzle. The injection pressures 10, 20, 30 and 40 bar are used for the experiments, and it is possible to move the nozzle vertically to image the liquid at different distances from the nozzle. The liquid is doped with the fluorescent dye Fluorescein that is a key component for extracting volumetric information. Fluorescein was chosen since it has

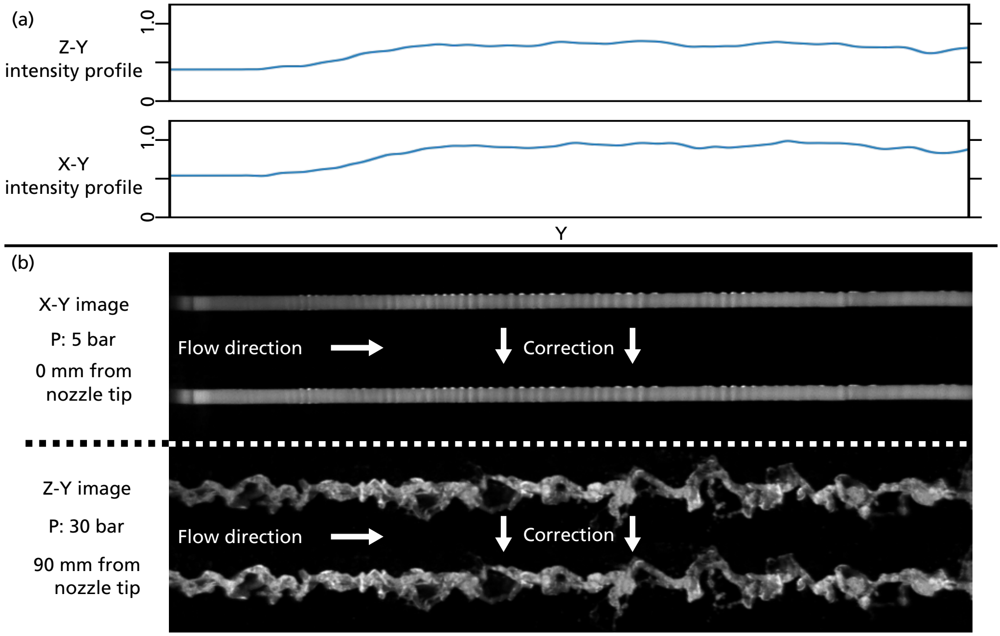


Figure 2.4: Estimated laser profiles in each camera (a) and example corrections (b) of two different cases with 5 bar and 30 bar injection pressure, respectively.

a high quantum yield and absorption cross section at the used illumination wavelength that produce enough fluorescent light for the high-speed measurements. This fluorescent dye is also water-soluble, non-toxic, and inexpensive which makes it ideal for this kind of experiment.

(c) Imaging: Two high-speed cameras are recording the liquid jet simultaneously with orthogonal views to one another. They are recording at 50 000 frames per second for a duration of around 8 ms ($\sim$ 400 frames). Both cameras are equipped with high-performance bandpass filters that will reject scattered light from the laser illumination and transmit the fluorescent light. When the nozzle is translated vertically, the cameras image the liquid at positions from 0 cm to 330 cm distance from the nozzle. An example image of the liquid at 30 bar injection pressure and a distance of 150 mm from the nozzle is shown in Figure 2.3(d).

2.4 Fluorescence to liquid thickness calibration

2.4.1 Laser sheet profile correction

The laser profile correction is found using recordings of a low-pressure water jet that is approximately cylindrical. For each camera, the following procedure is performed to find the profile. The jet is recorded for 20 single shots at seven different horizontal positions. All of these, $20 \times 7 = 140$, images are averaged in the vertical Y direction to extract a raw laser profile. A final convolutional smoothing with a gaussian ($\sigma = 5$ pixels) result in the laser profiles shown in Figure 2.4(a). Note that some artefacts induced by the nozzle has here been removed after averaging. Example corrections for a 5 bar case and 30 bar case are shown in Figure 2.4(b).

2.4.2 Corrupt LIF peaks correction

Volumetric LIF of water induce artefacts of the imaged intensity that are here called LIF peaks. The peaks are areas in the image with unproportionally large pixel intensities that cannot be explained by the liquid thickness, seen in Figure 2.5(a). The peaks are generally found close to the edges of the liquid as seen from the cameras. These artefacts can be explained by the refraction effects of both the laser and fluorescence light crossing the irregular liquid-air interface. In paper I, this effect was contributed to micro-lenses created by strong local curvatures on the surface. The micro-lenses focus laser light into a small volume and, in conjunction with the fluorescence dye being unsaturated, emit strong local fluorescence signal. The work by Gomez et al. [66] performed a similar experiment of high-speed VLIF but with eight views of the liquid enabled by two cameras equipped with quadscopes. The LIF peak artefacts were found here as well, and ray-tracing simulations were performed to find their origin. The simulations found that fluorescence light emitted close to the liquid surface has a higher probability of being trapped by internal total reflections. All trapped fluorescence then escapes at a position where the surface shape allows refraction instead of total reflection which result in the observed large intensity. Since the trapped signal travels along the surface inside the liquid, the escaped signal with a direction toward the camera will be imaged close to the liquid edge. The signal trapping theory can explain why the peaks are found to cover larger areas of pixels than what would be expected from the micro-lenses which makes it the more common effect even though micro-lens peaks can still be present.

The first step in correcting the corrupt peaks is to detect which pixels intensities I_{LIF} are corrupt. The detection is done through an outlier detection algorithm similar to a Bayesian classifier with the main assumption that a majority of the data does not include peaks. In addition, the assumption is made that the liquid is generally thicker when it is wider in

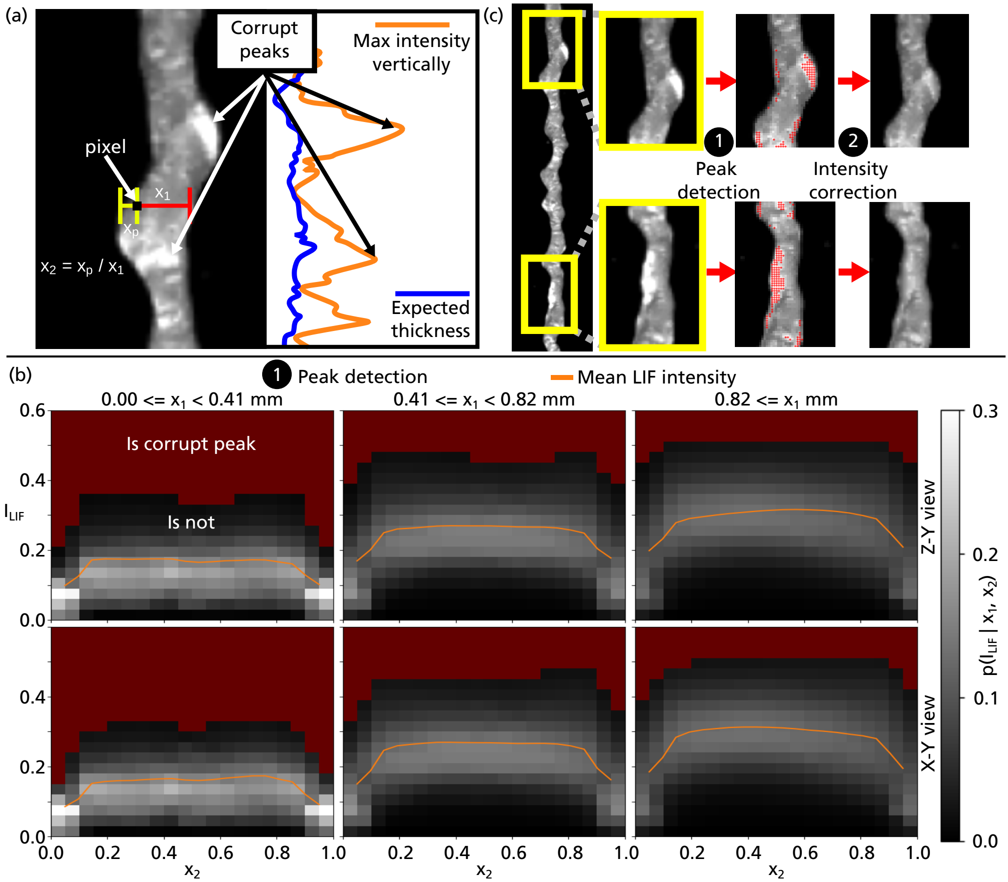


Figure 2.5: Detection and correction of corrupt LIF peaks. In (a), instances of corrupt LIF peaks are shown and the key values x_1 and x_2 used to detect them. In (b), the estimated conditional probability distribution $p(I_{LIF}|x_1, x_2)$ for the data is shown. The unproportionally large LIF peaks are defined as intensities larger than the 95th percentile for each combination of x_1 and x_2 shown as the red areas in the plots. In (c), the full process from profile corrected image to detected LIF peak pixels to corrected results is shown.

the current camera view which is captured through two key values. The key values are correlated with liquid thickness since there is a general rotational symmetry of the liquid around the Y axis. The key values are,

- x_1, liquid width for the row of the current pixel.
- x_2, normalized horizontal distance of the current pixel from the left edge of the liquid, value between 0 and 1.

The key values are illustrated for a single pixel in the left image of Figure 2.5(a). To calculate x_1 and x_2, a segmentation of the images is required that describes where the liquid is in the image. The segmentation is estimated from a threshold at 5% of the maximum imaged intensity per camera of all recorded 10 bar images. Now all data for the 10 bar injection case with vertical positions between 120 and 270 mm are analysed to extract around 80 million correspondences per camera between imaged intensity I_{LIF} and the key values x_1 and x_2. The found correspondences is used in a 3D histogram algorithm to estimate the joint probability distribution $p(I_{LIF}, x_1, x_2)$. The marginal distribution $p(x_1, x_2)$, the frequency of liquid widths and normalized pixel distance, does not here contribute information to whether a pixel is a corrupt peak, and when removed the conditional probability $p(I_{LIF}|x_1, x_2)$ remains as,

$$p(I_{LIF}|x_1, x_2) = \frac{p(I_{LIF}, x_1, x_2)}{p(x_1, x_2)}. \tag{2.6}$$

This conditional probability describes how frequent LIF intensities are given a liquid width x_1 and a normalized horizontal distance from edge x_2. Relatively lower LIF intensities are expected given both small x_1 values and x_2 values close to 0 and 1. The found distribution inhibit these properties as is seen in the histograms in Figure 2.5(b). For the largest widths (rightmost column of Figure 2.5(b)), a skew of the intensity distribution is visible. The skew is on the right side in the $Z - Y$ view and on the left in the $X - Y$ view that aligns with the side that the laser enters the liquid for each camera. The skew verifies that laser attenuation and fluorescence reabsorption affect the thickest liquid structures and will cause minor bias in the results.

The corrupt LIF peaks are defined as the LIF intensities larger that the 95th percentile of the estimated cumulative distribution function $F(I_{LIF})$.

$$\begin{aligned} F(I_{LIF}) &\geq 0.95 \rightarrow I_{LIF}|x_1, x_2 \in \text{corrupt LIF peak} \\ F(I_{LIF}) &= \sum_{f=0}^{I_{LIF}} p(f|x_1, x_2) \end{aligned} \tag{2.7}$$

The dark red areas in Figure 2.5(b) show where equation (2.7) is true for the estimated conditional distribution. The probability of the high LIF intensities in the red areas is very low and those intensities are therefore likely to be outliers. Example of pixels identified as corrupt is shown in Figure 2.5(c). The ratio of found corrupt pixels to the total number of segmented pixels in the images are commonly between 5% and 20%.

The detected corrupt LIF peaks should now be corrected. Here, an interpolation using a Gaussian Markov Random Field (GMRF) is performed. An introduction of what the GMRF is and how the interpolation works is found in appendix A. The GMRF interpolation is here used with a larger correlation horizontally compared to vertically since the liquid structures generally are elongated vertically. This weighting is done through the local

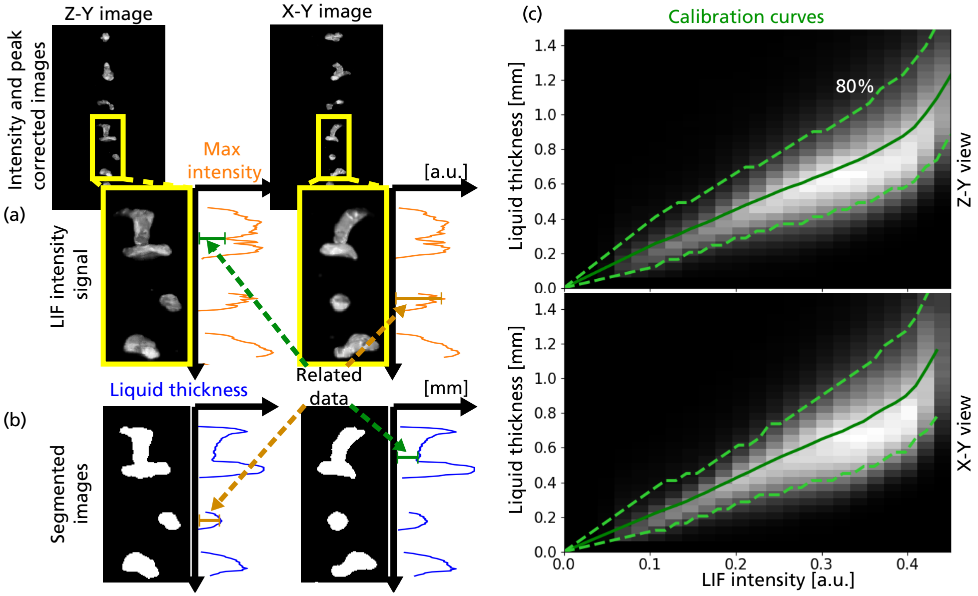


Figure 2.6: Extraction of calibration points and calibration curve. The maximum LIF intensity row wise in (a) is combined with the liquid width in (b) of the opposite camera since this width is the maximum liquid thickness for the first camera. This combination forms a calibration point where more than a million points were found for each camera. The distribution of these points is found in (c) where the non-parametric calibration curve in green is found as the expectation of liquid thickness given the LIF intensity. The dotted lines form a 80% confidence interval of the calibration data.

neighbourhood matrix g introduced in section A seen below,

$$\mathbf{g} = \begin{bmatrix} 0 & -1.3 & 0 \\ -0.7 & 4 & -0.7 \\ 0 & -1.3 & 0 \end{bmatrix}. \tag{2.8}$$

The used parameters of the GMRF are $\kappa^2 = 1e - 4$ and $\tau = 1$ and μ is calculated as the mean value of each image. Note that a side effect when using GMRF here is that values outside the segmentation is assumed to have the mean value μ as intensity in the interpolation. This is not wanted for the pixels at the edges of the liquid and therefore the segmentation is dilated with one pixel in all direction to add neighbouring pixels that are not LIF peaks and push edge intensities towards zero.

2.4.3 Extraction of calibration curve

The maximum fluorescence value horizontally in one camera view is connected to the thickness of the liquid in the other view since the views are orthogonal to each other as is illustrated in Figure 2.6(a) and (b). The thickness is here calculated in the same way the width x_1 was calculated for the corrupt LIF peak detection. Care is here taken to avoid rows

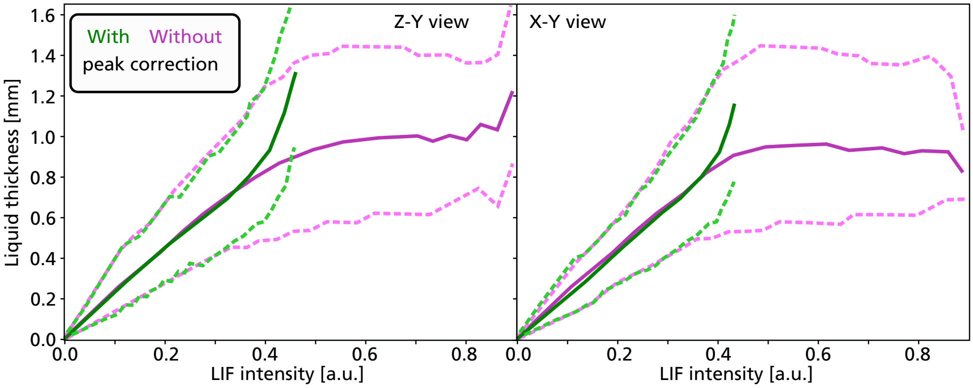


Figure 2.7: Comparison of the calibration curves with and without peak correction.

where multiple blobs of liquid are found. The combination of a maximum fluorescence value and a liquid thickness is called a calibration point. More than 1 million calibration points for each view are extracted from the 10 bar case and vertical positions from $120 - 270$ mm. A 2D histogram of the distribution of the calibration points are found in Figure 2.6(c). From the histogram of the calibration points, a calibration curve is estimated as the expected value of liquid thickness for each fluorescence bin found in the histogram $E[\text{thickness}|I_{LIF}]$. The calibration curve for each camera together with an 80% confidence interval is shown in green in Figure 2.6(c). The cameras are not positioned symmetrically around the laser which is the reason for why a calibration curve was estimated for each camera. Now all fluorescence intensities can be inserted into the calibration curve to estimate a liquid thickness.

2.5 Artefact peak correction performance

Calibration curves when no peak correction was applied has been estimated for comparison and the results are found in Figure 2.7. The main difference between the curves is visible for the larger LIF intensities as expected. The expected liquid thickness plateaus for intensities above 0.4. This means that the LIF intensities between 0.4 and 0.9 have been corrupted by the LIF peaks to give almost no information of the liquid thickness. With the peak correction, the expected diminishing return of liquid thickness is clearly visible for the largest intensities. The correction also enables prediction of larger liquid thickness.

The peak corrected calibration has varying performance when compared to the calibration without correction as can be seen in Figure 2.8. One example is here shown for the 10 bar and 20 bar injection pressure, respectively. On the left for each pressure curves show the expected liquid thickness from the other camera, and the predicted thickness both with and without peak correction. For the 10 bar pressure, three areas are indicated. The top and the bottom one show examples of how the peak corrected calibration can more accurately predict larger thickness. However, the indicated centre area show how the peak corrected calibration is more sensitive to minor changes in intensity for intensities above 0.4. This is the result of the mentioned the diminishing return of the calibration where a slight change in intensity give a substantial change in predicted thickness. On the images of liquid thickness in the same figure, the same observation can be made. The calibration without peak correction predicts a more spatially smooth thickness that is the result of the plateau in the calibration curve. This experiment is clearly challenging given the refraction effects that are at play. Even though the peak detection algorithm from observation is found to be accurate in finding areas with unproportionally large intensities, there is a loss of information for every pixel that is interpolated. This loss of information seems however to be an improvement since the peak correction adjust the calibration curve to better follow the expected theory.

2.6 Database of high-speed 3D jet volumes for different injection pressures

`https://spray-imaging.com/water-jet.html`

This experiment was created with modellers in mind. The nozzle is designed to avoid complex liquid dynamics, the volumetric data is a quantity readily available from simulations and care has been taken to estimate the limitations of the predicted liquid thickness. The last step is to give these results to modellers who can use it as validation of simulation results with the same boundary conditions as the experiment. The results are openly available for download at the link above. On this page, a detailed video description of the experimental setup and volumetric imaging approach is shown. One can then navigate to a tab called "Results overview" where short videos shown the whole volumetric dataset or the tabs with specific measurement results. The data can be downloaded either as a whole or only a single injection pressure and position distance from nozzle. If this page for any reason is no longer available, the data can also be found in a long-term available repository at `https://doi.org/10.17605/OSF.IO/CG3DF`.

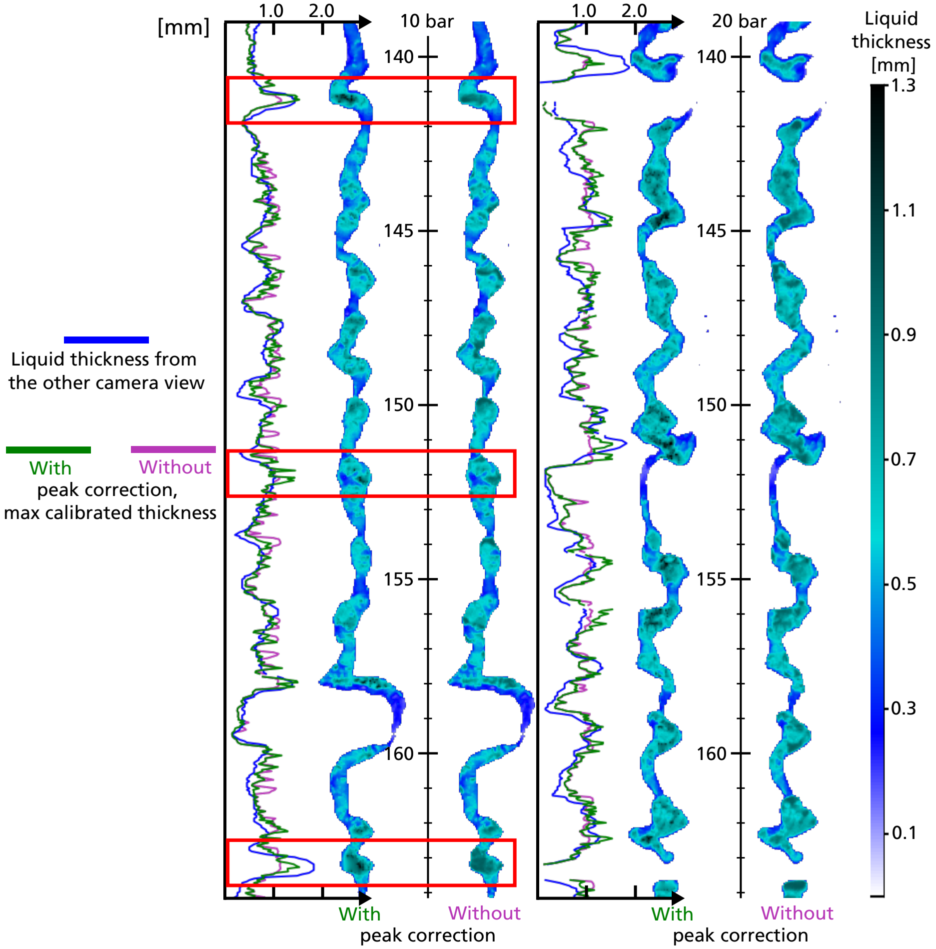


Figure 2.8: Comparison of calibrated liquid thickness with and without peak correction for results of 10 and 20 bar injection pressure. Three curves are shown for both cases. The liquid thickness from the other camera view (blue) should be compared to the corresponding predicted thickness using calibration either with (green) or without (purple) LIF peak correction. Two liquid thickness images for each pressure are shown where the left is with, and the right is without peak correction. The color in the images correspond to the predicted thickness as indicated by the colorbar on the right. The red boxes indicate areas where the two calibrations deviate from another that is further discussed in the text.

Chapter 3

High-speed 3D reconstruction of liquid surfaces

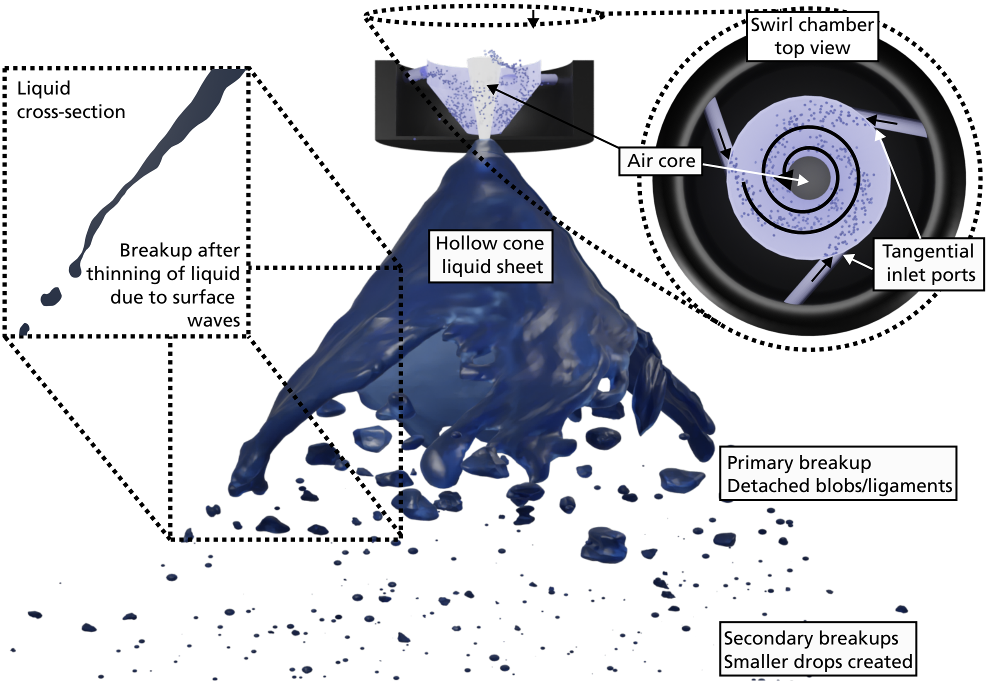


Figure 3.1: Hollow cone liquid sheet produced by a pressure swirl atomizer. Inside the nozzle the liquid rotates around the air core while moving towards the exit. The rotational and vertical momentum creates the hollow cone after exiting the nozzle. The liquid along the cone gets increasingly thinner when the liquid is dispersed over a larger area downstream from the nozzle. Eventually the liquid is so thin that it becomes unstable and breaks creating drops, blobs, and ligaments.

3.1 Characteristics of liquid sheets

Liquid sheets, just like liquid jets, are unstable and will eventually break to form droplets and the applications of liquid sheets are also similar to the liquid jet [60]. A liquid sheet can be produced in multiple ways where the pressure swirl nozzle, illustrated in Figure 3.1, is one common nozzle design that is used in this thesis. The liquid is here injected into a swirl chamber through one or more tangential ports. This injection creates a vortex motion of the liquid and if the injection pressure is high enough, an air core is created that blocks part of the nozzle exit. The liquid vortex with air core is a hollow liquid structure that have both rotational velocity and velocity downstream from the nozzle. These two velocities combined create the hollow cone liquid sheet. The radial expansion of the conical sheet downstream from the nozzle result in decreased sheet thickness that will rupture and breaks into drops, ligaments and blobs. These structures will then further break with high enough injection pressure. The pressure swirl atomizer is often used because of the simplicity in its design and operation. For example, with this nozzle one can avoid extra control devices that add cost, weight, and reduce reliability, crucial aspects in for example aerospace applications. The nozzle also efficiently creates small droplets relative to the pressure used for injection [67] which is preferred in combustion applications.

3.2 Liquid surface 3D reconstruction approach

The 3D reconstruction approach in this chapter is based on the Fringe Projection technique [68]. This is a structured illumination technique where fringes (parallel lines) are projected onto an object. When the fringes are observed from an angle, they curve as a

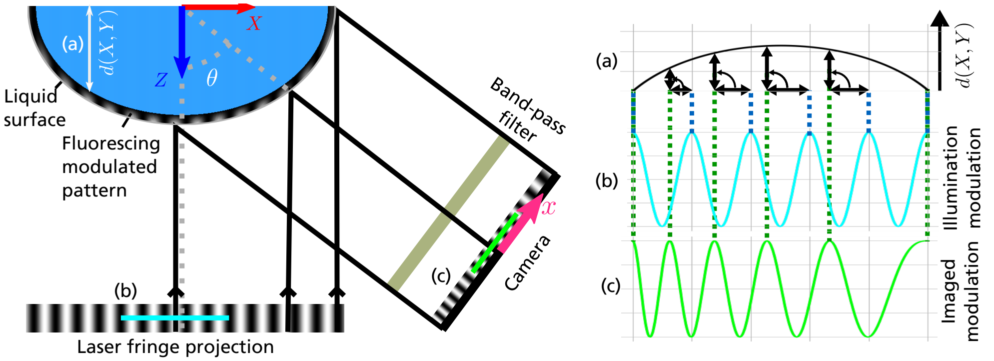


Figure 3.2: Illustration of the 3D recontruction approach. A cross-section of the liquid shape is here viewed from the top and the laser fringe projection is indicated. The laser induces fluorescence mainly on the surface of the liquid and specular light is rejected by the band-pass filter when imaged. It is seen how the 3D structure (a), shifts the cosine illumination modulation (b) when imaged (c) from the angle θ. A comparison of the magnitude of these shifts, as is illustrated in the plots on the right, can give quantitative 3D reconstruction values.

function of the surface structure. The deformation of these fringes can be quantified using a post-processing procedure to extract the topological structure of the object. A conceptual illustration of the 3D reconstruction procedure is shown in Figure 3.2. Fringe projection 3D reconstruction has previously been applied to both static [69], moving objects [68] and high-speed measurements [70] and the concept of Fringe Projection enable 3D reconstruction from a single snapshot image [71, 72]. Applying fringe projection 3D reconstruction on liquids is not straight forward since liquids typically are both transparent and induce significant amount of specular reflections which are problematic since the fringes should in an optimal case only reflect of the liquid surface and be proportional to the illumination intensity. The approach in this work solves the problem by adding fluorescent dye to the liquid at a fairly high concentration. With this solution, a LIF signal will mainly be emitted close to the liquid surface. In addition, a band-pass filter attached to the camera will sufficiently reduce specular scattered light and mainly transmit the fluorescent light from the surface. This technique we call Fringe Projection - Laser Induced Fluorescence (FP-LIF). FP-LIF is suitable for measurements on any liquid surfaces where also waves or other deformations found in phenomena outside of liquid sheets can be reconstructed.

3.3 Description of the experimental setup

FP-LIF is here applied with the experimental setup illustrated in Figure 3.3 and briefly described below. More details can be found in Paper II.

(a) **Illumination**: An illumination is here, similar to the Volumetric LIF experiment in the previous chapter, used to induce fluorescence in the liquid where a blue continuous wave laser at 445 nm and 3W is used. The fringe structure is here created using a Ronchi grating and spatial filtering to get a sinusoid modulation. Note that it is nowadays often preferred to create the structured illumination using a Diffractive Optical Element (DOE) instead of a Ronchi grating since the DOE has a lower loss in the optical transform.

(b) **Liquid**: A hollow cone liquid sheet is here 3D reconstructed that is created by a pressure swirl injector. The injector is a $10x$ enlarged model of an injector used in aircraft engines. The enlargement of the injector simplifies imaging of smaller details in the created hollow cone structure. More details on the nozzle can be found in Paper III . The water used in the experiment is doped with a fairly high concentration of fluorescent dye (Fluorescein, $1 : 100$ ratio). The high concentration limits the fluorescence to the surface of the liquid. This is important since only the surface should be imaged, and 3D reconstructed.

(c) **Imaging**: A single high-speed camera is used recording at 20 000 frames per second. The fact that only one camera is required makes this high-speed 3D reconstruction technique more accessible both in simplification of experimental setup and with regards to cost. To

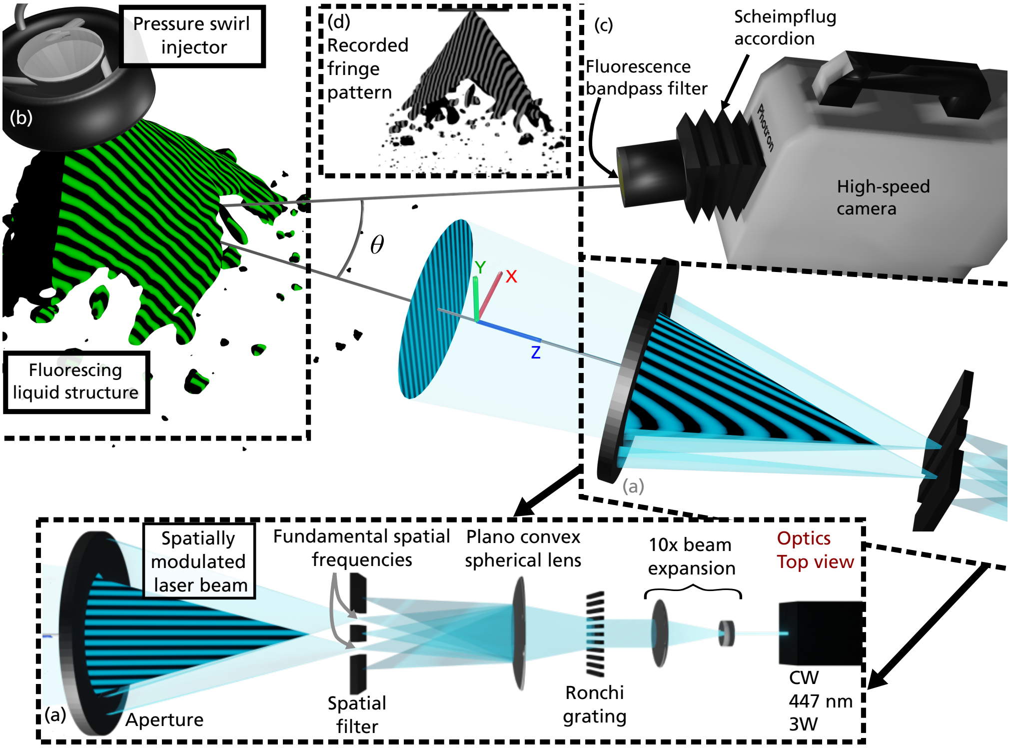


Figure 3.3: The experimental setup. The illumination (a), the liquid to reconstruct and injection system (b) and the high-speed camera (c). A simulated example recorded image is shown in (d) called a fringe pattern.

extract 3D, the camera must have a non-zero angle θ relative to the illumination direction. The camera is mounted with a Scheimpflug adaptor that rotates the focus plane of the camera to follow the conical surface of the liquid. The camera is also equipped with a band-pass filter that transmits the fluorescent green light and rejects the scattered blue laser light.

Image post-processing is required to extract 3D coordinates from a fringe pattern image.

3.4 FP-LIF post-processing

In Figure 3.3(d) a simulated fringe pattern is seen that is called a fringe pattern. This pattern can be modelled by equation (3.1).

$$I(x,y) = \begin{cases} A(x,y) + B(x,y)\cos(\varphi(x,y)) + \epsilon(x,y), & (x,y) \in \mathrm{fg} \\ \epsilon(x,y), & (x,y) \in \mathrm{bg} \end{cases} \tag{3.1}$$

Software box 3.1

The whole described FP-LIF post-processing routine is implemented in the open-source python package called `fp23dpy` (short for fringe pattern to 3D python). For information on installation and usage with example code, see the gitlab repository `https://gitlab.com/roth.adrian/fp23dpy`.

This model contains two parts where the first is the foreground (fg) where A is the background illumination, B is the modulation amplitude and φ is the phase of the fringe pattern. The second part is the background (bg) that is modelled as noise ϵ. The phase φ of a fringe pattern contains information of the imaged objects third coordinate. The first step towards a 3D reconstruction is to extract the phase using phase demodulation.

3.4.1 Phase demodulation with the Continuous Wavelet Transform method

The kind of fringe pattern that is recorded with the FP-LIF technique is called an open fringe pattern. The opposite, a closed fringe pattern, have areas where the fringes connect to themselves and create circular patterns. The simplest approach for demodulating an open fringe pattern is to record three or more images where the fringe projection is shifted with a known phase for each image. The imaged phase for a pixel can then be estimated by only using information of the same pixel in the phase-shifted imaged. Since multiple images are required, this is not a suitable approach for high-speed imaging where only one image is preferred. With one image, a spatial area around each pixel must be considered to estimate its phase and different phase demodulation techniques have been developed that are reviewed in Paper II . The phase demodulation using the Continuous Wavelet Transform (CWT) method is applied in this work.

The CWT method is based on wavelets which is a wave-like oscillation package. In this work, one of the most common wavelets called the 2D Morlet wavelet is used which is simply a real cosine and imaginary sine multiplied by a gaussian as seen in equation (3.2).

$$\Psi_{Morlet}(x, y) = \exp\left(ik_0 x - \frac{1}{2\sigma^2}\left(x^2 + \frac{y^2}{\varepsilon}\right)\right). \tag{3.2}$$

Here, $2\pi/k_0$ is the base period length of the wavelet, σ is the standard deviation of a Gaussian envelope and ε controls the degree of anisotropy of the envelope where $\varepsilon = 1$ is isotropic. A wavelet has the requirement of a close to zero amplitude at the zero frequency. To fulfil this requirement the inequality $k_0 \geq \pi\sqrt{2/\ln(2)} \approx 5.34$ must be satisfied for the Morlet wavelet [73]. A value of ε greater than 1 is used to enable demodulation of a wider range of wave directions in the fringe pattern. The used parameters for the Morlet wavelet are $k_0 = 6$, $\varepsilon = 3$ and $\sigma = 0.6$.

The Morlet wavelet is chosen as the so-called mother wavelet Ψ in the CWT jargon. This wavelet can now be given three degrees of freedom through scaling and translation.

$$\Psi_{a,b}(x) = \frac{1}{a}\Psi\left(\frac{\mathbf{x}-\mathbf{b}}{a}\right). \tag{3.3}$$

Here, a is the scaling parameter and b is the translation vector. a is in this application a real positive nonzero value and $\mathbf{b}$ is a 2 value vector with integer values corresponding to a pixel in the image to demodulate. The daughter Morlet wavelet will have a period length of $2\pi a/k_0$. Note that one can also add another degree of freedom in rotation. The rotation add accuracy in the demodulation for large phase gradients parallel to the fringe direction, but it also adds a significant computational cost. The open fringe patterns do as mention generally not have significant changes in wave direction that could not be handled by the choice of ε parameter. One exception is for some areas in simulations that is discussed in section 3.6.2.

Now to the definition of the 2D-CWT of a function f.

$$W_f(a, \mathbf{b}) = \int\int f(\mathbf{x})\frac{1}{a}\Psi^*\left(\frac{\mathbf{x}-\mathbf{b}}{a}\right) d^2\mathbf{x}, \tag{3.4}$$

where f is here the recorded image. The three degrees of freedom of the wavelets, scale a and translation in two dimensions $\mathbf{b}$, are the input parameters to the CWT. The value of the CWT for each combination of a and $\mathbf{b}$ is denoted wavelet coefficient.

With the CWT, this work demodulates the phase of a fringe pattern by first finding the so-called ridge of the CWT. It is calculated as follows,

$$R(\mathbf{b}) = \text{argmax}_a(|W_f(a, \mathbf{b})|). \tag{3.5}$$

The ridge is the wavelet scale a, and in continuation period length, that best match the fringe pattern frequency at pixel $\mathbf{b}$. The ridge is then used to estimate the wrapped phase as,

$$\hat{\varphi}_{\text{wrapped}}(\mathbf{x}) = \arg(W_f(R(\mathbf{x}), \mathbf{x})). \tag{3.6}$$

Wrapped in this context means that the phase only has values in the interval $(-\pi, \pi]$. The next step is then to unwrap this phase where the unknown multiple of 2π is found and added for each pixel to extract a continuous phase. The phase unwrapping is an important step of the demodulation since an unwrapping error in one part of the image can propagate to other areas. Phase unwrapping can be performed by recording multiple images with different fringe period lengths, called temporal phase unwrapping. The unwrapping can then be performed on a pixel level in the image [68]. This is however, once again, not preferred in high-speed imaging and therefore an algorithm using spatial information is applied instead. In this work, the phase unwrapping algorithm implemented in the scikit-image python package [74] is used. This unwrapping is a reliability guided unwrapping

Software box 3.2

The CWT phase unwrapping is implemented in the python function:

```
def demodulate(
    signal: np.ndarray,
    calibration: dict,
    return_amplitude: bool = False,
    return_ridge: bool = False,
):
```

in the module `fp23dpy.demodulation`. Here, the input arguments are the fringe pattern image (`signal`), the calibration with information of fringe period length and orientation. Optionally this function can return an estimation of fringe amplitude (B in equation (3.1)) and/or the ridge that correspond to an estimate of the local frequencies of the pattern. All implementation of the continuous wavelet transform is found in module `fp23dpy.wavelets`. The other common phase demodulation technique called the Fourier method is also implemented in the function `fp23dpy.demodulation.ft2`.

method that first finds a reliability map of where the wrapped phase is most reliable. Then, the algorithm starts to unwrap the phase in areas with high reliability to avoid creating problems that propagate through the unwrapping [75]. This phase unwrapping technique work sufficiently for this data. The unwrapped phase is relative which means that it has an added unknown constant. To extract the absolute phase, required in the next section, one can correct the phase by knowing the absolute phase in at least one pixel. In this work, it is assumed that one pixel close to the injection source is always at the same 3D position with a known absolute phase. Note that if the phase is not continuous throughout the image, the absolute phase must be known in one pixel for each continuous area.

3.4.2 Triangulating 3D coordinates from image phase

The fringe projection has a so-called global phase[1] that is a function of the world coordinate system X, Y, Z. The connection between the global phase $\varphi(X, Y, Z)$ and the imaged phase $\varphi(x, y)$ enables the 3D reconstruction of a surface. The global phase is here modelled as,

$$\varphi(X, Y, Z) = \frac{2\pi}{T_g} X, \tag{3.7}$$

---

[1] Note that all phases in this section are absolute.

where T_g is the period length of the fringe projection in a world length unit. Meters is used in this theory, but any length unit can be used as long as it is consistent. Note that this is a simplification. Generally, fringes produced by either a laser or a projector will diverge with distance from the projection source. In the case of fringes produced with lasers, this change will be negligible and is therefore not included in the model. However, a simple calibration together with an added parameter for the divergence can be added to the model if required.

The fringes are projected onto an object with a topography $d(X, Y)$ which can be described by a set of homogeneous coordinates,

$$\{\, D = [X, Y, d(X, Y), 1]^T, (X, Y) \text{ is on object} \,\}. \tag{3.8}$$

The homogeneous coordinates simplify projective geometry since affine and projective transforms can be represented as a single matrix. The final number with value 1 is here called the homogeneous coordinate. After scaling this coordinate with a non-zero value it will still represent the same Cartesian coordinate. The extraction of the Cartesian coordinate from homogeneous coordinates is performed by first scaling all four coordinates so that the homogeneous coordinate equals to 1. The Cartesian coordinates are then the first three values in the vector.

In this reconstruction model, it is assumed that the camera is orthographic. In an orthographic camera, an object will have the same imaged size no matter the distance between the camera and the object. This can be contrasted to the more commonly used perspective camera where objects further away appear to be smaller. To get an orthographic camera one can use a telecentric lens. An orthographic camera can also be approximated if the distance from the camera to the object divided by the size of the object is large enough. This approximation is analysed further in section 3.6.5. The orthographic camera matrix is here modelled as using a modified pinhole camera matrix[2] P,

$$P = KC = \begin{bmatrix} s & 0 & x_0 \\ 0 & s & y_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{3.9}$$

P consists of two parts, the intrinsic matrix K and the extrinsic matrix C. The intrinsic matrix converts the world coordinate lengths to the camera coordinates in pixels. Here, s is the scale in pixels/meter and x_0 and y_0 are the principal image point. The extrinsic matrix rotates and translates the world coordinate system to the cameras orientation. In this case, a rotation of θ degrees is performed around the Y-axis and then no translation is performed. The orthographic part of this camera can be seen in the final row of C. If it were to be perspective, the first three numbers would be a vector with Euclidean norm 1.

---

[2] The pinhole model and camera matrix is a common way of modelling imaging systems in computer vision [76]

Software box 3.3

Calibrating the fringe pattern base period length can be performed using the script `fp23dpy.__calibrate_main__`. If the `fp23dpy` package is installed one can in the terminal run:

```
python -m fp23dpy calibrate <path-to-calibration-image>
```

where a graphical interface helps you and create a calibration file typically called `calibration_<calibration-image-path>.txt`. This calibration file needs more information of scale s and camera angle θ to be fulfilled.

Figure 3.2 illustrates how the topography coordinates of the object D is projected onto the orthographic camera plane. This process can be described with a matrix multiplication of the camera matrix P and the homogeneous coordinates D. The result is the following,

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = PD = \begin{bmatrix} s(\cos(\theta)X - \sin(\theta)d(x,y)) + x_0 \\ s(-Y) + y_0 \\ 1 \end{bmatrix}. \tag{3.10}$$

Here one can see for the first equation row that the projection of the fringes from an angle θ that is non-zero results in encoded information of both the world X and Z coordinate in the image x direction. Solve for X in the first row of equation (3.10) gives,

$$X = \frac{x - x_0 + s\sin(\theta)d(x,y)}{s\cos(\theta)}, \tag{3.11}$$

Then the expression for X is inserted into the global phase equation (3.7) which gives the image phase,

$$\varphi(x,y) = \frac{2\pi}{T}(x - x_0 + s\sin(\theta)d(x,y)). \tag{3.12}$$

Here, T_g has been exchanged to the more practical $T = T_g s\cos(\theta)$ which is the base period length of the fringe pattern where T is measured in pixels. This is the phase that was demodulated in the previous section. Now, equation (3.12) is solved for d,

$$d(x,y) = \frac{T}{2\pi s\sin(\theta)}\left(\varphi(x,y) - \frac{2\pi}{T}(x - x_0)\right). \tag{3.13}$$

This is the third coordinate of the object that would in ordinary imaging not be accessible. All 3D topology coordinates of the object are calculated as,

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \frac{x - x_0 + s\sin(\theta)d(x,y)}{s\cos(\theta)} \\ -\frac{y - y_0}{s} \\ d(x,y) \end{bmatrix}, [x,y] \in \text{fg}. \tag{3.14}$$

Software box 3.4

The full post-processing routine is run through the function:
`def fp23d(signal: np.ndarray, calibration: dict):`
Required input is the fringe pattern and calibration and the output is the estimated 3D coordinates for each pixel. To estimate correctly scaled output coordinates, the calibration must at least contain values for: T, gamma (fringe direction in the image), scale, theta, and absolute_phase (or absolute_threed). Note here that a masked numpy array can be used as input fringe pattern to avoid demodulation of the masked pixels and therefore also unnecessary errors in the phase unwrapping.

To use this model for 3D reconstruction the parameters θ, s, x_0, y_0 and T must be calibrated. θ is measured by appropriate means from the setup. s can be found by measuring the pixel size using for example a ruler and inverting it. x_0 and y_0 are typically set to the centre of the image, half of the image shape. Finally, a calibration image of a flat 3D object orthogonal to the illumination direction, $d(x, y) = 0$ can be used to estimate T. Either the frequency peak in the Fourier transform or the pixel length between fringes can then be used to estimate T. A script has been written to assist in this process described in Software box 3.3

Note that it is important to have vertical fringes in this setup, the phase change in the X-direction. A horizontal or diagonal phase will give none or less information respectively of the hidden dimension. However, if horizontal fringes are preferred, one can put the camera above or below the object instead.

3.5 The hollow cone liquid sheet

The experimental results of the hollow cone liquid sheet created with the pressure swirl atomizer is discussed extensively in Paper III and only a brief discussion is given here. The calibration parameters for this experiment were a camera angle $\theta = 16°$, period length $T \approx 6.9$ pixels and a scale $s = 18$ pixels/mm. The liquid sheet produced by the pressure swirl nozzle is in the form of a hollow conical structure when stabilized. However, the structure has temporal variations that can be detected and analysed with this 3D measurement at 20 000 frames per second. Figure 3.4 show, in addition to example fringe pattern and segmentation, five 3D reconstructed timepoints with 50 µs time increment. Here, the initial phase of a sheet rupture is seen where the hole is expanding. The five reconstructions are part of a 1000 frames long sequence that can be interactively viewed at `https://3d.spray-imaging.com/cone4d/`. In this sequence, surface waves, ruptures and breakups are found. In addition, there is a sequence of 3D reconstructions from the start of injection to stabilised sheet that can be viewed at `https://3d.spray-imaging.com/cone4d_` `evolution/`. To access all data used in Paper III see the repository `https://doi.org/10.17605/OSF.IO/ZT97W`. In this repository there is also a backup solution to view the 3D results interactively if the links above are dead.

Spray-angles are an essential spray characteristic since it defines the dispersion of the liquid for different situations. Typically spray-angles are measured using shadowgraphy or planar laser imaging. Then only a single spray-angle is extracted from each image. With the 3D reconstruction, it is possible to extract multiple angles along different azimuth-angles along the sheet as is shown in Figure 3.4(e) and (f). These angles are important since they define the dispersion of the liquid in spray applications and temporal variations give information of the flow stability. To estimate the angle, the radial variation of the 3D reconstructed surface along the 3D Y-coordinate is extracted for different azimuth-angles as is shown in Figure 3.4(f). Then a line is fitted to the radius curve for Y values between -1 and -12 mm where $Y = 0$ mm is the approximate position of the nozzle. The angle of the fitted line is defined as the spray-angle. The 2D spray-angle is also estimated from the same images to have as comparison. Note that this angle corresponds to the azimuth-angle $-16°$ because of the camera angle. Time resolved spray-angles are shown for two azimuth-angles together with the 2D spray-angle in Figure 3.4(g). Generally, the spray-angle varies between $40°$ and $50°$ where the fluctuations are found to be connected to the previously observed surface waves. An analysis of the accuracy of the estimated spray-angles are found in the next section.

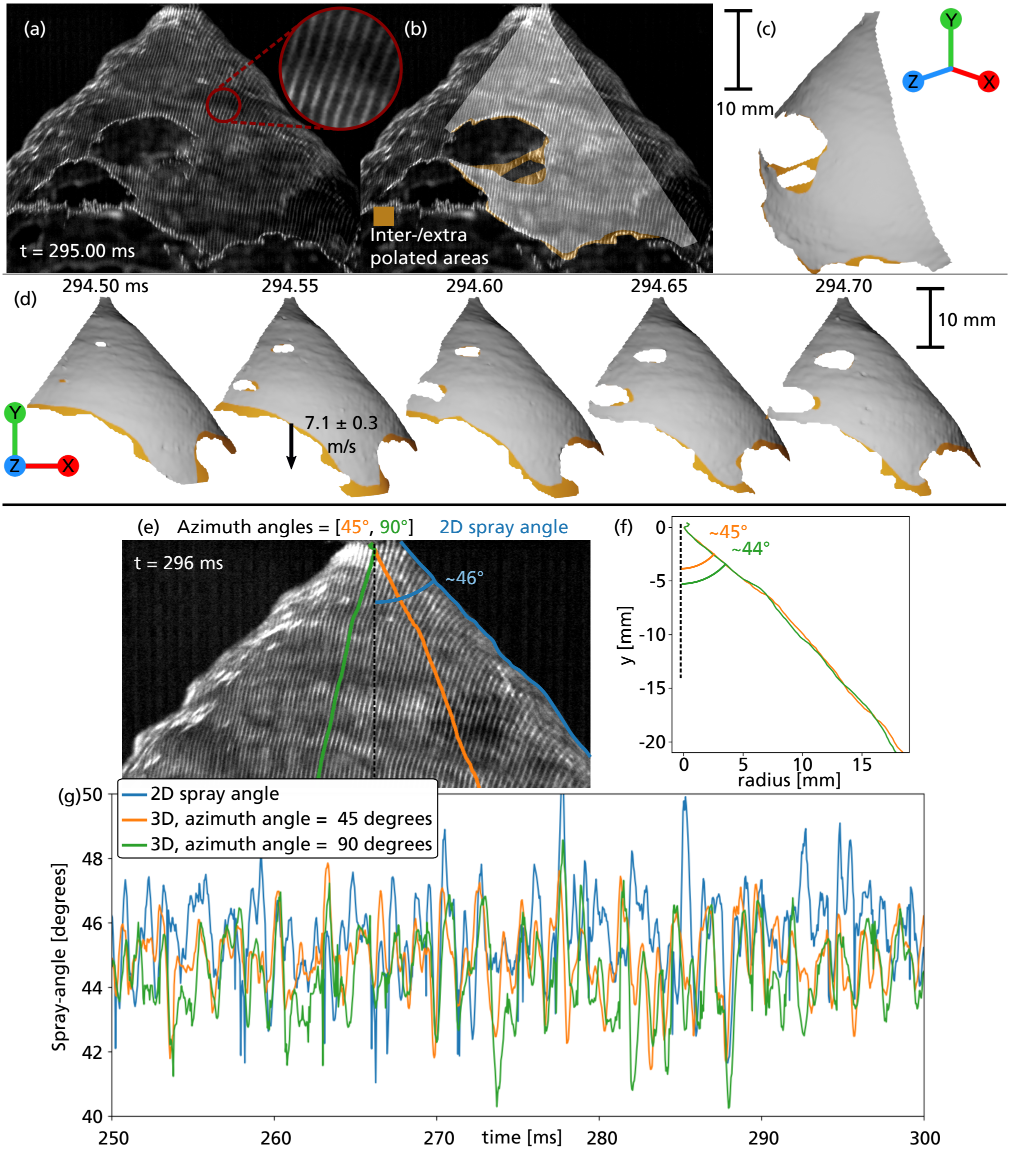


Figure 3.4: High-speed 3D reconstructions of the hollow cone liquid sheet together with spray-angle results. In (a), the raw fringe pattern is shown and the same image with areas of overlayed brighter pixels are seen in (b). These pixels have been manually segmented by the criteria of having clear enough fringes to be reconstructed. The dark orange colour represents areas that could not be reconstructed but are interpolated in 3D to clearly show where liquid is. This is for example important to show the actual edge of the liquid sheet holes. A single 3D reconstruction is shown in (c) from an angle indicated by the coordinate axes. Then 5 consecutive 3D reconstructions are shown in (d) with a time increment of 50 µs. The absolute time in the figure is here relative to the start of the pump for injection where the high-speed recording started. The reconstructions are used to estimate spray-angles along different azimuth-angles where two are shown in (e) together with the definition of the 2D spray-angle. The estimation of the spray-angle for each azimuth-angle is shown in (f) for a single time point. Finally, (g) show the time evolution of the spray-angles along the two azimuth-angles together with a spray-angle estimated directly from the 2D image.

3.6 Validation using simulations of conical structures

3.6.1 Cone structure and fringe pattern simulation

A simulated cone has been produced to validate the accuracy of the 3D reconstructed liquid sheet from the pressure swirl nozzle. The simulation setup parameters are chosen to match the experiment on the pressure swirl atomizer which means that conclusions from these simulations should be mainly drawn for this specific experiment. Only limited conclusions can be drawn for other experimental setups and liquid structures. The simulated structure can have a varying cone-angle that is connected to the spray-angle. The cone-angle α is modelled by the following equation of how the radius r change with the vertical world coordinate Y as,

$$r(Y) = -\tan(\alpha) \cdot Y + r_0, \tag{3.15}$$

where r_0 is the initial radius. Note that the world Y vector is up which is the reason for negating the first term. To represent an observed surface wave, a bump with different widths and heights on the conical surface is added with the following gaussian profile,

$$\hat{r}(\hat{Y}) = h \cdot \exp\left(\frac{-(\hat{Y} - \hat{Y}_0)^2}{\frac{\mathrm{w}^2}{2\log(2)}}\right), \tag{3.16}$$

where h is the height of the wave, w is the width (full width half maximum) and $\hat{Y}_0$ is here set to place the wave at the vertical centre of the structure. The wave is then rotated to the chosen cone-angle by,

$$\begin{aligned} r(\hat{Y}) &= \cos(\alpha)\hat{r}(\hat{Y}) - \sin(\alpha)\hat{Y} \\ Y(\hat{Y}) &= \sin(\alpha)\hat{r}(\hat{Y}) + \cos(\alpha)\hat{Y}. \end{aligned} \tag{3.17}$$

Since the second equation is hard to solve for $\hat{Y}$, instead the solution for $r(Y)$ is found by calculating r and Y for many $\hat{Y}$ and estimating $r(Y)$ through interpolation.

Software box 3.5

The `fp23dpy` package has several example structures that can easily be simulated. A cone with wave can for example be simulated using the function:
`fp23dpy.examples.example_cone_bump.render()`
See the examples folder for code on how to generate and 3D reconstruct the examples. Note that the GMRF noise used here is not part of the package but if you can produce the GMRF sample (there is example code on this in the appendix A), it can be used as input to the function `fp23dpy.simulation.render_from_map` to simulate a fringe pattern.

The radius function is used to extract a rotationally symmetric structure in the XZ plane for each Y coordinate as,

$$d(X, Y) = r(Y) \sin \left(\arccos \left(\frac{X}{r(Y)} \right) \right), \ \text{abs}(X) < r(Y). \tag{3.18}$$

Values of absolute X larger than the radius are outside of the structure. Examples of cones without and with wave are shown in Figure 3.5(a) and (b) respectively.

The simulated 3D cone structure is then virtually imaged in a fringe projection setup. The setup assumes a camera angle of $\theta = 16°$, a fringe period length $T = 6.9$ and scale $s = 18$ pixels/mm, the same as was used in the experiments. The 3D structure is first projected from the world coordinates (X, Y, Z) into the coordinate system of an orthographic camera (x, y).

$$\begin{aligned} x &= sX \\ y &= -sY \\ d(x, y) &= -X \sin(\theta) + d(X, Y) \cos(\theta) \end{aligned} \tag{3.19}$$

A grid of $1024x1024$ x and y pixel coordinates are used to calculate the phase φ in each pixel using equations (3.19) and (3.12). The imaged fringe pattern is then finally estimated through equation (3.1).

A simulated fringe pattern with constant values of A and B in equation (3.1) is shown in Figure 3.5(c). This simulated image is not realistic when comparing to the experimental image in Figure 3.5(e). To make the simulation more realistic, two types of noise are added,

- Background noise ε is sampled as Gaussian noise with a $\sigma_\epsilon \approx 10\%$ of the modulation amplitude.
- Amplitude (B) noise sampled from a Gaussian Markov Random Field (GMRF) that here adds a spatial correlation to the sampled values where two neighbouring pixels has a higher probability to have similar values. The used parameters of the GMRF are $\sqrt{1/\tau} = \sigma_{\text{GMRF}} \approx 30\%$ of the modulation amplitude and $\kappa^2 = 0.05$. The parameters are chosen to give similar amplitude variation as is found in the experimental data. More information of the GMRF sampling is found in section A.

An example simulated fringe pattern with sampled background noise and amplitude variation is shown in Figure 3.5(d).

3.6.2 General reconstruction performance

The simulated fringe pattern is 3D reconstructed using the described post-processing where example reconstructions are shown in Figure 3.5(f) and (g). To get an overall understand-

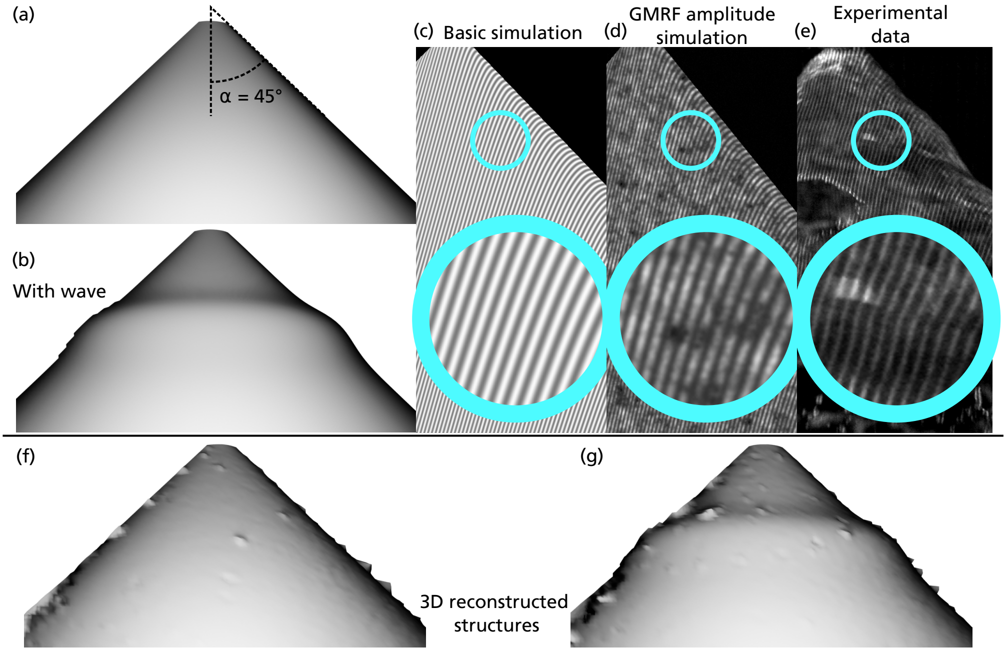


Figure 3.5: Simulation and reconstruction of fringe patterns with realistic noise. (a) and (b) are 3D structures of a cone without and with a surface wave (width 7 mm, height 1.2 mm). From (a), a fringe pattern is simulated without noise, shown in (c), that is less realistic than the simulated fringe pattern with background and amplitude noise in (d) when compared to the experimental image in (e). (f) and (g) are 3D reconstructions of the structures in (a) and (b) respectively with simulated fringe pattern including noise.

ing of the reconstruction error depending on the pixel position in the image, 300 simulated fringe patterns without any wave and with $\alpha = 45°$ are reconstructed. Each reconstruction is subtracted by the correct 3D structure to extract the error in each pixel for all three dimensions. Then the Root Mean Square of the Errors (RMSE) for each pixel is calculated. The distribution of these errors both spatially and in histogram format are shown in Figure 3.6 categorized by reconstruction dimension. The Y dimension is here omitted since it does not depend on the quality of the phase demodulation and will always be perfectly reconstructed.

The spatial distribution of the RMSE shows larger errors on the horizontal edges of the structure. On the left edge the problem is connected to the shorter local fringe period length in these areas since the surface structure is moving away from the camera. These short period lengths are more problematic for the phase demodulation when there is noise and will therefore induce larger errors. At some point the local fringe period will exceed the Nyquist frequency limit where the fringes no longer can be resolved nor demodulated. The problem on the right edge is instead connected to the almost horizontal direction of the fringes. Mainly vertical fringes are searched for in the phase demodulation and variation can be handled only up to some extent by the chosen ε for the Morlet wavelet. As

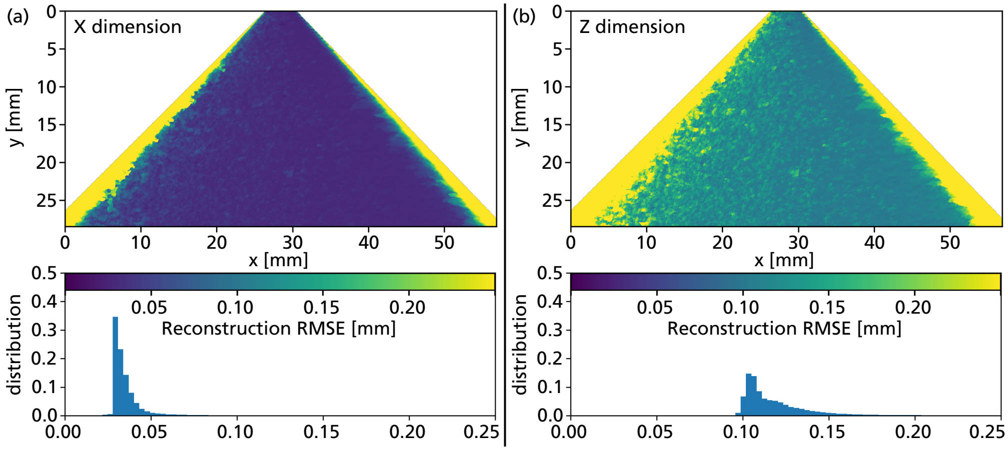


Figure 3.6: Error in 3D reconstruction of simulated fringe patterns. 300 fringe patterns are simulated with the same procedure as the one found in Figure 3.5(d) and each image is then reconstructed. The RMS of the error between the correct structure and the reconstruction is calculated for each pixel is calculated for each dimension and the resulting values for the X-dimension are shown in (a) and Z in (b). The Y-dimension is not included since its reconstruction does not include the phase estimation is always correct. On the top is the distribution of RMSE over the image and on the bottom are the histogram of all pixel RMSE. The speckles in the Z-dimension results is most probably because the RMSE did not converge after only 300 samples.

was mentioned in the post-processing, one can add a rotation dimension in the CWT to handle this but at a computational cost. The errors in the 3D reconstruction are also caused by the sampled amplitude variation. Amplitude variation is problematic for fringe projection 3D reconstruction since it introduces frequencies in the imaged intensity that are not connected to the 3D structure [68]. If the amplitude variation is reduced, either experimentally or in post-processing, the convolutional envelope of the CWT can be shrunk which will reduce the smoothing and improve reconstruction resolution. Another side effect of the convolutional envelope is that a consistent bias is induced for this conical structure. The bias is caused by how the phase of each pixel is estimated through an average of the local phases found spatially around the pixel. On the left and rightmost areas of the cone structure, this weighting is biased toward the outer areas of the structure since the surface gradients are increasing away from the camera. This bias will have a minor effect of the spray-angle estimations as is discussed in the next subsection.

The RMSE of X and Z dimension is generally around 0.04 and 0.12 mm, respectively. These uncertainties are propagated from the uncertainty of the phase demodulation procedure. The X uncertainty is 30% of the Z which can be explained by that the X calculation multiply the phase with a sine and Z a cosine and $\sin(\theta)/cos(\theta) = 0.29$. To receive the lowest uncertainty for all dimensions θ should be set to 45°. The problem with this approach is that a larger θ has a penalty in that a larger part of the object is shadowed in the recorded images. For example, with $\theta = 45°$, 25% of the illuminated area of a rotationally symmetric object will be hidden behind itself while only 9% is hidden with $\theta = 16°$. The used θ was

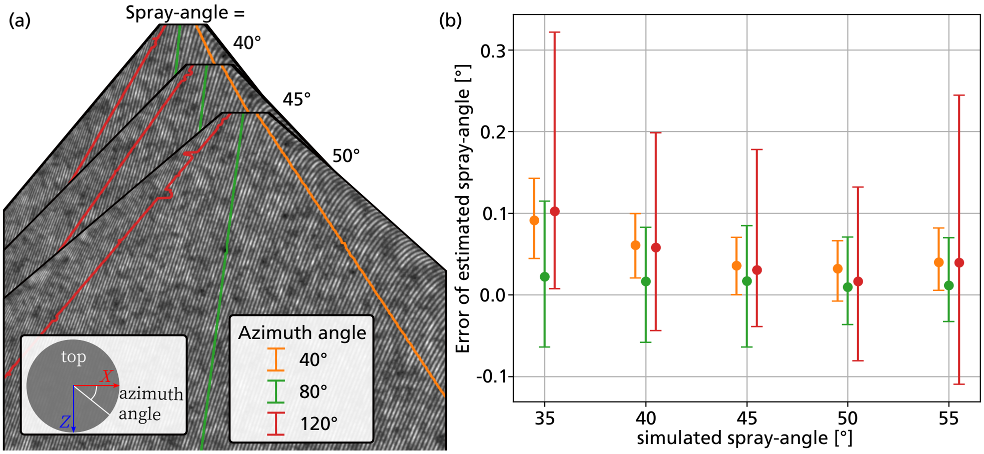


Figure 3.7: Spray/cone-angle simulation and estimation to find accuracy and uncertainty. In (a) the lines following three azimuth-angles are shown in a simulated fringe pattern. For each of these azimuth-angles, a spray-angle is estimated 300 times and the median error and variation (16th and 84th percentile) are shown in (b) for 5 different actual spray-angles.

found to be a good trade-off both in this work and from others using fringe projection techniques [68]. The base period length T should, however, be slightly larger (above 10 pixels) for improved reconstruction.

3.6.3 Spray/cone-angle analysis

Cones with different angles are simulated to examine the accuracy of spray-angle estimation used in the experiments. Cone structures with angles ranging from 35° to 55° are simulated and for each angle, 300 simulated fringe patterns were sampled and reconstructed. The cone-angle of the three different azimuth-angles 40°, 80° and 120° for each reconstruction, shown in Figure 3.7(a), was estimated with the exact same procedure as for the experimental case where median bias and standard deviation of estimated angles is shown in Figure 3.7(b). All biases are larger than zero with a maximum around 1% of the simulated angle which indicate overestimation of angles. The smallest bias is for the centre 80° azimuth-angle and the side azimuths have larger bias. The reason for this is found to be the bias caused by the convolutional envelope of the CWT, described in section 3.6.2. The spray-angle in the centre of the image (azimuth-angle 80°) is least biased since the centre pixel of the image has been calibrated to the correct 3D coordinates. A final observation is that larger azimuth-angle give a larger variation in estimated angle which was also observed for the experimental spray-angles. This is connected to how the phase demodulation is more sensitive to noise in these areas as was also discussed in section 3.6.2.

Software box 3.6

The code used to estimate spray/cone-angles can be found in the examples folder of the `fp23dpy` gitlab repository.

3.6.4 Surface wave analysis

Reconstruction of simulated conical structures with waves of different widths and heights give important information of the FP-LIF technique limitations. One such surface wave was found to have an approximate width of 7.6 mm and a height of 0.7 mm as is shown in Figure 3.8(a), (b) and (c). Conical structures are here simulated with waves of certain width and height. Example fringe patterns and reconstruction of cones with two different waves are shown in Figure 3.8(d). The ratio between the height and the width of the wave is an important heuristic connected to the maximum 3D gradient of the waves. A limiting Height/Width Ratio (HWR) is expected where the reconstruction can no longer resolve the large gradients of the wave. To explore this limitation, conical objects with different simulated wave widths from 0.4 to 20 mm are applied together with 40 HWRs between 0.25 and 1. A fringe pattern is simulated, and 3D reconstructed 100 times for each combination of width and HWR where the reconstructed wave height is estimated. The median height of all 100 simulations is used as the estimated height for this combination of width and HWR. The HWR is then extracted when the error between the estimated height and the simulated height is larger than 2%. This HWR is defined as limiting value of the reconstruction. The performance of the reconstruction is found by dividing the limiting HWR with the theoretical maximum and the result for all widths is shown in Figure 3.8(e). The overall performance lies around 65% of the theoretical maximum and is then decreasing for wave widths thinner than 4 mm. The decrease here is most likely connected to the convolutional envelope effects of the CWT. One can conclude that the observed surface wave in the experimental data can clearly be resolved in the 3D reconstruction and is no artefact.

The found limiting HWRs for waves are not limits for the FP-LIF technique itself. The limits are what can be reconstructed with the implemented post-processing and the simulated experimental parameters. In theory this limit is at 0.57 where waves above this ratio will no longer have continuous phase. Some improvement can be made with the post-processing by attempting rotations of the mother wavelet during the phase demodulation as was discussed earlier. However, if the wave on the cone has a height above 57% of the width, other approaches must be attempted to correct the non-continuous phase. With non-continuous phase, the challenge is to extract the absolute phase for the disconnected areas of continuous phase. The mentioned multiframe technique called temporal phase unwrapping can be used with a cost of reduced recording frame rate, If this is not possible,

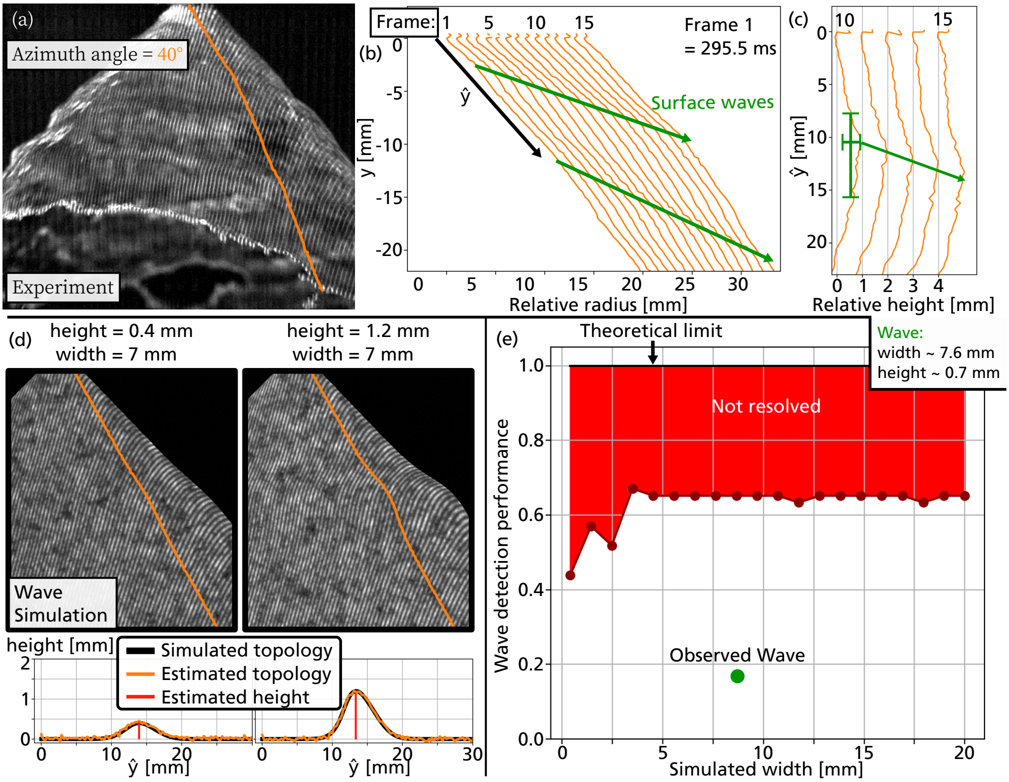


Figure 3.8: Surface wave simulation and estimation to find reconstruction limitations. (a), (b) and (c) show experimental observations of a surface wave and how it is moving over time. This wave has an approximate width of 7.6 mm and a height of 0.7 mm. In (d), example of simulated waves is shown with their corresponding reconstruction. In (e), the wave detection performance is found that is relative to the theoretical limit of when the wave creates unresolvable fringes. The performance is found for different wave widths by iteratively increase the height until the reconstruction fails to estimate the wave height within 2% of the correct value. This height is then divided by the theoretical maximum of around 57% maximal.

another approach is to approximate the absolute phase in at least one pixel of each area by use of temporal information. The 3D structure at the previous time-point in the recording can be used if the absolute phase is known there. This approach was used in a few areas of the recorded data of the pressure swirl atomizer just after starting the pump. However, for temporally deforming liquid processes, it is both hard to track where the same liquid was previously, and the liquid deformation will induce errors even for a perfect track. The final possibility to be able to resolve higher HWR's is to tune the experimental parameters. Typically, it is not possible to change the object to be 3D reconstructed but one can often position the camera and illumination so that they align with the object's surface. It is preferred to have a surface as much perpendicular relative to the illumination and camera as possible. This enables the reconstruction of larger surface gradients and in addition have the advantage of reducing out of focus effect, making the use of Scheimflug configuration redundant, and reduce perspective projection effects that are discussed in section 3.6.5. To increase possible resolved HWR's one can also increase the fringe period length T or reduce the camera angle θ. However, both these changes come at the cost of larger uncertainty of 3D coordinates.

3.6.5 Perspective camera analysis

The implemented reconstruction algorithm assumes that an orthographic camera is used which is not the case for the pressure swirl experiment. The impact of this assumption is evaluated here through simulated fringe patterns using a projective camera instead of an orthographic one. The following procedure is performed to simulate these fringe patterns.

The first step is to calculate the projective camera matrix that describes the cameras position and rotation according to the pinhole camera model [76]. The projective camera is placed at a projection distance r from the centre of the object[3] to reconstruct and is rotated with

---

[3] The centre of the object is also the origin of the coordinate system.

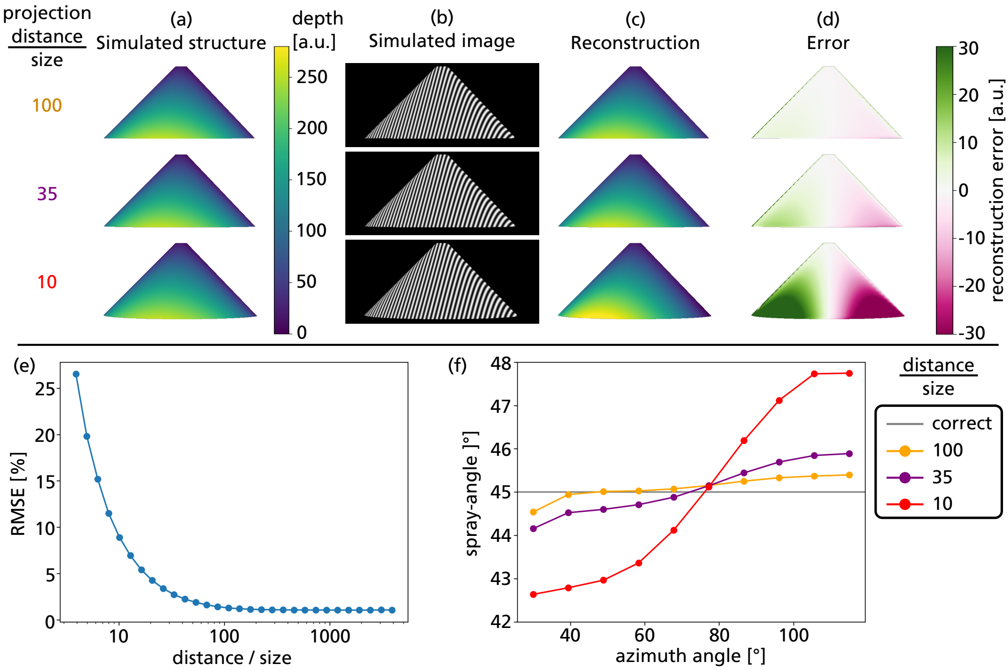


Figure 3.9: Perspective projection simulations to understand how the assumptions of orthogonality affect the reconstruction and spray-angle accuracy. Simulations are visualised for three different ratios of projection distance and size. Larger value here corresponds to a better assumption of orthogonal projection. In (a), the simulated depth coordinate of the projected 3D structure is shown, (b) show the simulated fringe pattern, (c) the reconstructed depth and (d) the error. In (e), the RMSE of all pixels has been plotted in relation to different distance/size ratios. Finally, (f) show the estimated spray-angle for different distance/size ratios and azimuth-angles.

an angle $\theta = 16°$ around the Y-axis. The position of the camera is then,

$$c_0 = r \begin{bmatrix} \sin(\theta) & y_0 & \cos(\theta) \end{bmatrix}^T, \tag{3.20}$$

where y_0 is chosen so that the object is in the centre of the image. The camera should look at object which gives the camera rotation matrix R,

$$R = \begin{bmatrix} -\cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & -\cos(\theta) \end{bmatrix}. \tag{3.21}$$

The extrinsic camera matrix C that describes both the position and rotation is,

$$C = \begin{bmatrix} R & -R \cdot c_0 \end{bmatrix}. \tag{3.22}$$

The intrinsic camera matrix K that transforms camera coordinates to pixel coordinates is,

$$K = \begin{bmatrix} f & 0 & x_0 \\ 0 & f & y_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3.23}$$

This is the same as was used for the orthographic camera with the change of notation from scale s to focal length f. The focal length is here calculated so that the object fills the whole image at the chosen r and θ. The projective camera matrix P is then calculated as $P = KC$

With the projective camera matrix, the 3D coordinates of a simulated structure, in this case a cone, are projected to estimate the pixel coordinates for each 3D coordinate on the surface. Some parts of the structure are behind other parts from the view of the camera. Since we have a rotationally symmetric structure and a camera rotated around the Y-axis, one can find these parts from a horizontal pairwise difference operation on the projected camera x-coordinates. The negative differences are areas where the structure is behind other areas and are therefore removed. A 2D interpolator is produced that enables estimation of 3D coordinates from camera pixel coordinates and the world coordinates is found for all pixels in a $1024x1024$ image. The Z-coordinate for three example projections are shown in Figure 3.9(a). The difference between the examples is the radial distance between the camera and the cone-object. Note that the important heuristic for understanding how large role perspectivity plays is the ratio between the distance r and the size of the object. The projected X world coordinate in each pixel can now be used with equation (3.7) to estimate the imaged phase. The global period length T_g is here calculated as,

$$T_g = \frac{T}{\cos(\theta)}, \tag{3.24}$$

with the same parameter values as in the experiments. The phase is then used with equation (3.1) to produce the simulated fringe pattern where examples are shown in Figure 3.9(b).

Software box 3.7

Projective camera simulation is also implemented in the `fp23dpy` python package. See the example code of the gitlab repository to get you started.

The reconstruction and error to the correct structure is shown in (c) and (d) of the same figure.

The projection plays a larger role for large objects relative to the projection distance (smaller projection distance size ratios) which correspond to larger errors as is seen in Figure 3.9(e). From this error, the reconstruction will underestimate spray-angles for azimuth-angles below around 75° and overestimate the remaining spray-angles as is seen in Figure 3.9(f). For the pressure swirl nozzle experiment, the ratio between projection distance and hollow cone liquid sheet size is approximately 35. This gives a bias of up to 0.9° that has a size comparable to the uncertainty of the measured spray-angles and can easily be compensated for in measurements using the FP-LIF technique. Another solution is to implement fringe projection 3D reconstruction with projective cameras as is described in [68] that requires a more complex calibration and post-processing to successfully perform.

Chapter 4

High-speed 3D velocimetry and sizing of respiratory droplets

4.1 Characteristics of respiratory droplets

Since the COVID-19 pandemic started, a large amount of research has been delved into understanding the virus SARS-CoV-2 and its spreading. However, there are still questions to answer regarding the respiratory droplets. Information of the droplet velocity distribution, size distribution and number concentration are here of great interest since these quantities largely affect where the droplets end up and their risk of infecting other people. An illustration of a spreading event via coughing is shown in Figure 4.1. The droplet spreading

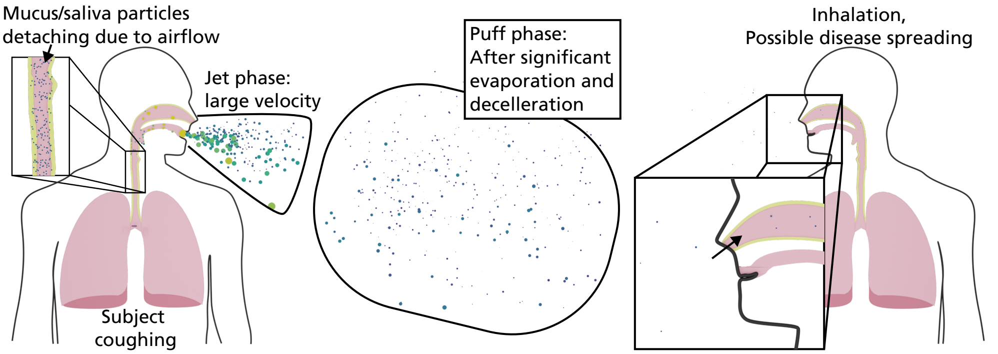


Figure 4.1: Illustration of airborne disease spreading via respiratory droplets. An infected subject cough where the large airflow detaches droplets of mucus and saliva. The droplets exit the respiratory system through the mouth and nose where there typically is an initial jet phase with high velocity of the ejected droplets. The droplets are then decelerated and expand into a cloud called the puff phase where slower velocities are found and evaporation shrinks the size of droplets. Finally, the droplets might be inhaled by another subject that is then at risk of infection.

quantities are especially important for modellers to use as boundary conditions in simulations of disease spreading where risks of different situations can be compared [77]. Several types of respiratory events are here of interest such as speaking yelling and coughing.

The research on droplet velocities have been mainly measured indirectly by estimating air flow velocities. The air flow speeds found for coughing measurements alone show a large variation from 2 m/s to 29 m/s [78]. The main reason for the large variation is how different measurement techniques and metrics[1] are presented and more work is needed to fully characterize the velocity of different types of respiratory events.

The size of respiratory droplets is often discussed in connection to a 5 µm limitation that has been used to define whether a droplet is an aerosol that contribute to airborne disease transmission. This limit has been challenged during the COVID-19 pandemic because of how larger droplets have been found to remain suspended in still air for more than 5 seconds [3]. Similar challenges to the velocity are apparent for measurements on respiratory droplet size distribution. Different measurement techniques have been applied that have different sizing ranges and counting efficiencies[2] which makes comparisons challenging. In addition, evaporation is a significant effect for most measurements where then the estimated droplet sizes are underestimated. This is important to be aware of when comparing size measurements that have compensated by using an evaporation model [53] and the ones that have not. A vast majority of number of exhaled droplets have sizes below 5 µm [54, 53]. However, the number of virions within a droplet is connected to the volume of the droplet. Since droplet volume increases by the cube of size, droplets larger than 5 µm can contain significant more numbers of virions and are therefore also of interest for the full characterization of disease spreading.

Previous research have found large variations in the concentration or number of produced droplets from both person to person for the same type of respiratory event and the same person and same type of respiratory event [50, 47, 53]. The number of produced droplets seem to be correlated to whether a subject is infected with a respiratory tract infection [54] and with age [53]. Here, infected, and older subjects produce more droplets. The number of droplets does also seem to be correlated to the volume of a vocal activity [53].

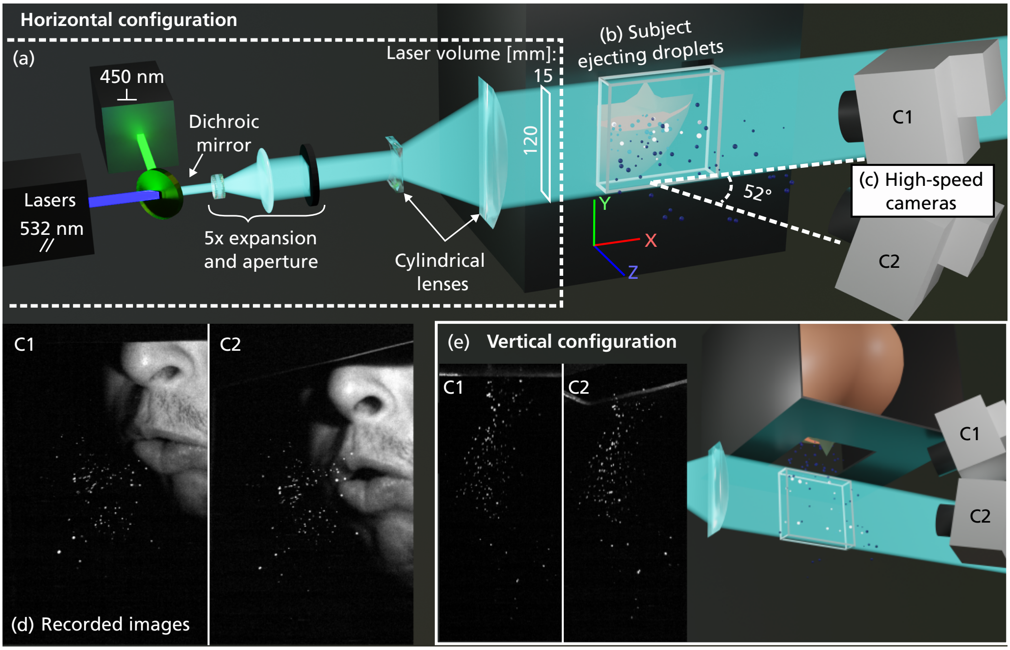


Figure 4.2: The experimental setup. The illumination (a), the person ejecting respiratory droplets (b) and the high-speed cameras (c). Example recorded images are shown in (d). For these experiments, two alternative configurations are used, the horizontal shown at the top and the vertical shown in (e).

4.2 Respiratory droplet 3D velocimetry approach and experimental setup

In this work stereo 3D Particle Tracking Velocimetry (3D-PTV) is applied to track respiratory droplets and extract their velocity. 3D-PTV is a technique of tracking individual particles where multiple different approaches have been previously developed [12]. Even though the tracking approach described in this work is applied on respiratory droplets it can also be used on other droplets or particle dynamics of interest. The experimental setup illustrated in Figure 4.2 is used in this work described below. For more details on the experimental setup, see Paper v.

(a) **Illumination**: Lasers are used to illuminate the droplets that will scatter the laser light and the scattered light is used for detection. In this work, two continuous wave lasers with a total of 9 W power are combined using a dichroic mirror and the beam is then shaped to create a homogeneous laser beam that illuminates a volume with cross section height

---

[1] For example, some use the maximum speed, others the average and measurements are performed at different distances from the mouth.

[2] The counting efficiency is the percentage of number of actual particles that are found for different particle sizes.

120 mm and width 15 mm. This area can be varied depending on the requirements of the measurement. However, the fluence of the illumination will decrease with a larger illumination volume which means that the minimum detectable droplet size will increase.

(b) **Liquid**: Respiratory droplets ejected by a subject performing one of the three respiratory events: speaking, yelling, and coughing. In the speaking and yelling events, the subject will say "Protect better against COVID" to further clarify the goal of this work and to illustrate the use of specific vowels discussed in the results. There are two different configurations in these experiments where the subject is either ejecting droplets from the side relative the illumination volume, horizontal configuration, or from the top, vertical configuration. The advantage of the horizontal configuration is that in principle all droplets will cross the illumination and can be tracked. The vertical configuration instead enables detection of some droplets for a longer time where velocity changes can be detected. For the horizontal configuration seen in Figure 4.2, measurements of all three respiratory events were performed while in the vertical configuration only measurements of coughs were recorded. The reason for this was the challenge of consistent aiming into the 15 mm thick sheet when speaking and yelling.

(c) **Imaging**: Two high-speed cameras are used in a stereo configuration with an approximate 23° relative angle between them. Both cameras image the scattered laser light by droplets inside the illumination volume. The stereo configuration enables 3D triangulation of droplet positions. The cameras are placed in a semi-forward 52° angle relative to the il-

Software box 4.1

The algorithms described in detail below have been implemented in python and are available as open source software in the python package called `ptv3py` (particle tracking velocimetry in 3D python). For more information on installation and usage see `https://gitlab.com/roth.adrian/ptv3py`. This package is based on `pandas.DataFrame` to store information of particles. This datatype stores tabular data like what you find in spreadsheets where for example a single 2D track is represented by the following:

```
id_track  id  frame  coordinates  coordinates
       0   0      0     1.590874     4.587343
       0   1      1     2.207410     4.887104
       0   2      2     2.826070     5.172979
       0   3      3     3.433815     5.484175
       0   4      4     4.053660     5.797738
```

More columns and rows are added when required. There is a folder with example code in the repository to get you started.

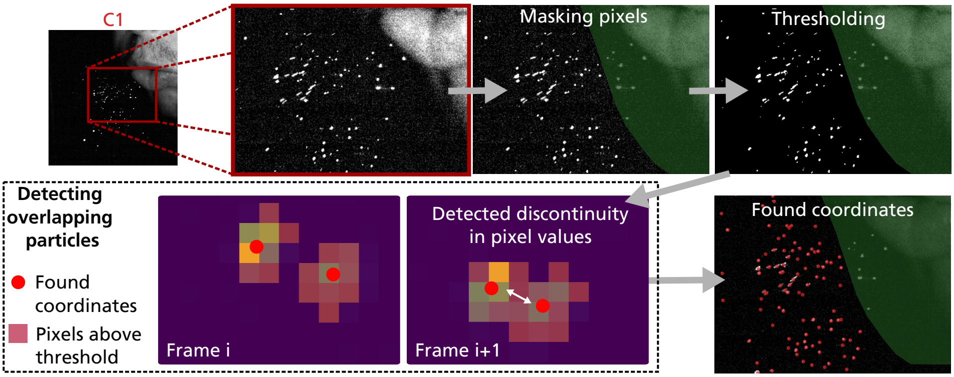


Figure 4.3: Illustration of post-processing in step 1, finding particle pixel coordinates. Here, an overlap detection algorithm is also applied that find discontinuities in the pixel values and in continuation particles that are too close to one another in the images.

lumination direction that gives a stronger scattering signal and enables detection of smaller droplets. The cameras record at 15 000 frames per second with an exposure time of 50 µs per frame.

4.3 Post-processing, from raw images to 3D particle tracks

This process of finding velocity is developed for droplets but can be generalized to other particles which is why the term particle is used throughout this section. The post-processing 3D-PTV used here is divided into three consecutive steps.

1. Finding the pixel coordinates of particles in both cameras, Figure 4.3.
2. Using a camera calibration to triangulate the particle positions in 3D, Figure 4.4.
3. Use tracking in 3D to extract particle tracks and particle velocities, Figures 4.5 and 4.6.

4.3.1 Finding particle pixel coordinates

The finding process is illustrated in Figure 4.3. First some pixels are masked because of the lamp used to illuminate the face of the subject. The lamp is important to find when in time the cough or speaking events start and end, however, this also adds light that should not be considered as particles. For the remaining pixels, a threshold is used to binarize the image. The used threshold was 25 counts for the horizontal and 40 counts for the vertical configuration of the 12-bit pixel depths. The binary image is then processed by a connected components algorithm, with 8 connectivity, which finds areas of neighbouring pixels with

Software box 4.2

The finding algorithm is implemented in the module `ptv3py.particle_finding`. The function that does the most work in detecting particle positions is,

```
def find_particles_single_frame(
    images,
    threshold: int,
    connectivity: int = 8,
    gradient_threshold: float = None,
    relative_gradient_threshold: bool = False,
    ...
):
```

where a sub function `divide_label_subpeaks_gradient` is used to detect and split overlapping particles. Note that one can supply a function as parameter to this function to extract features from each particle island in addition to the coordinates. In this work the island integrated intensity is calculated and added as a column in the dataframe to be used for both triangulation matching and particle sizing described in section 4.7. Another note is that you typically use the function `ptv3py.particle_finding.find_particles` that iterates through all frames in your recording and returns all found particles as a single dataframe.

values above the threshold, here called particle islands. Some particles are here overlapping and then two or more particles are found in the same particle island which will create errors in 3D triangulation and tracking. In this work, the overlap detection algorithm described by Maas et al. [79] is used. The overlap can be detected by assuming that pixel values should generally be decreasing with radial distance from the maximum pixel value. Here, a Breadth First Search (BFS) algorithm with 4 connectivity in the image is employed that starts from the maximum pixel value of a particle island. The search continues as long as the pixel value gradients are negative or below a slightly positive threshold to account for noise. If any pixels within an island is not reached in this BFS, a new BFS round is initiated from the maximum value of the remaining pixel values. This is repeated until all pixels in the island have been reached. Finally, the particle island is split into new particle islands according to the number of performed BFS rounds. Pixels that can be connected to multiple different new islands are classified to the closest particle island. Each one of the newfound islands are now defined as a particle where the intensity weighted centroid defines the pixel coordinates for the found particle. The process of finding particle pixel coordinates is repeated for both cameras and all frames in the recording.

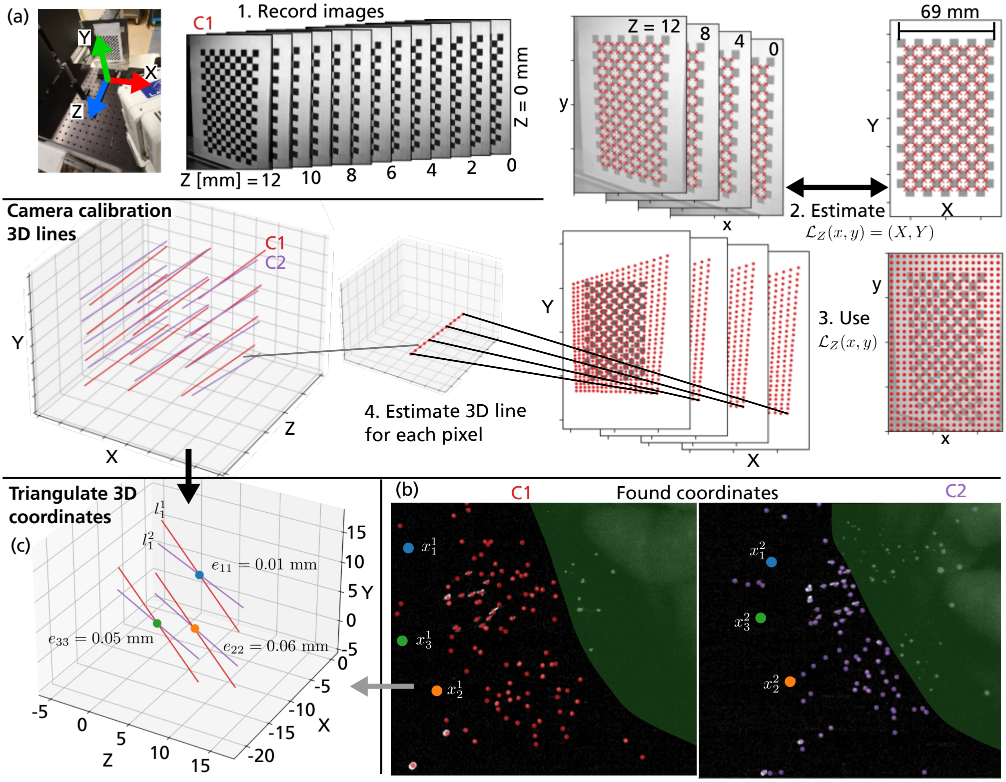


Figure 4.4: Illustration of post-processing step 2, camera calibration and 3D triangulation. In (a), a description of the calibration procedure if found where a checkerboard pattern is imaged at 13 various positions along a rail. Then a polynomial transform $\mathcal{L}$ is estimated for each position to go from pixel coordinates (x, y) to world coordinates (X, Y). The transform for each position is applied to estimate the world (X, Y) for all pixel coordinates in the camera. Finally, the 13 3D coordinates for each pixel is used to estimate a 3D line where all 3D lines are the camera calibration. The previously found pixel positions seen in (b) are then used to triangulate the 3D coordinates in (c) of particles using a matching algorithm described in the text.

4.3.2 Camera calibration and 3D triangulation

The second step is to triangulate 3D positions using the found image positions and a camera calibration. In this work, the camera calibration scheme explained in [80] is used. Here, in contrast to the common pinhole approach [76] that estimate a camera matrix, a 3D line is found for every single pixel in both cameras. A particle or particle found anywhere along this 3D line will then be imaged in the corresponding pixel. The 3D lines are estimated from calibration images of a known pattern (here a checkerboard pattern with 19x10 intersections is used). The pattern is fastened on a rail and calibration images are recorded with decided spacings on the rail. The pattern should be fastened so that the plane of the pattern defines two of the world coordinates axes, here X and Y. The direction of the rail is then the third world base direction Z. In this work 16 images with 1 mm spacing between $Z = 0$ and $Z = 15$ was used. In each calibration image, the 190 corners of the pattern are located where each corner also represents a world (X, Y) coordinate. All 190 correspondences are used to estimate a 2D polynomial transform $\mathcal{L}_Z$ to go from pixel coordinates to world coordinates. The order of the polynomial is chosen to 3 as suggested by [80] where 10 coefficients are estimated for each dimension. At least a first-degree polynomial is required to take the perspective aspect of the cameras into account and with a third degree, distortion coefficients are also implicitly considered.

$$\mathcal{L}_Z(x, y) = \begin{pmatrix} a_0^X + a_1^X x + a_2^X y + a_3^X x^2 + a_4^X y^2 + a_5^X xy + a_6^X x^3 + a_7^X y^3 + a_8^X x^2 y + a_9^X xy^2, \\ a_0^Y + a_1^Y x + a_2^Y y + a_3^Y x^2 + a_4^Y y^2 + a_5^Y xy + a_6^Y x^3 + a_7^Y y^3 + a_8^Y x^2 y + a_9^Y xy^2 \end{pmatrix}. \tag{4.1}$$

One set of coefficients for the transform $\mathcal{L}_Z$ is estimated for each of the 16 calibration images. Each transform is now used to estimate world X and Y coordinates for all pixels in the corresponding calibration image. Now, each pixel in each camera has 16 3D coordinates, X and Y from the transform, and Z from the known rail position. These 3D points are finally used to fit a 3D line in a total least squares algorithm for each pixel in both cameras. With a recorded particle pixel coordinate, it is now known that the particle must be positioned somewhere close to the 3D line for that position. Note that non-integer pixel positions are linearly interpolated from the surrounding pixel 3D-lines.

The calibration can be used to triangulate 3D positions. If one position in each of the two cameras are known to be the same 3D particle the following calculations estimate its 3D position. The pixel positions found in the previous sub section are here denoted x_i^c where c is the camera 1 or 2 and i is the index of the position in a single frame. Each position is translated to a 3D line l_i^c with the calibration where each line is a combination of a 3D point $\mathbf{q}_i^c$ and a direction $\mathbf{v}_i^c$. All points on the line can be found in the following set:

$$l_i^c = \{\mathbf{p}_i^c \mid \mathbf{p}_i^c = \mathbf{q}_i^c + \lambda_i^c \mathbf{v}_i^c, \lambda_i^c \in \mathbb{R}\}. \tag{4.2}$$

With one position and then also 3D line from each camera l_i^1, l_j^2, there exists a 3D point that minimizes the sum distance to both lines. This point can be found by equivalently finding the point on each line where the lines are closest to one another. These points correspond to $\lambda_i^{1,\text{closest}}$ and $\lambda_j^{1,\text{closest}}$ which are calculated as,

$$\begin{bmatrix} \lambda_i^{1,\text{closest}} \\ \lambda_j^{2,\text{closest}} \end{bmatrix} = \begin{bmatrix} ||\mathbf{v}_i^1||^2 & -\mathbf{v}_i^1 \cdot \mathbf{v}_j^2 \\ -\mathbf{v}_j^2 \cdot \mathbf{v}_i^1 & ||\mathbf{v}_j^2||^2 \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{v}_i^1 \cdot (\mathbf{q}_j^2 - \mathbf{q}_i^1) \\ \mathbf{v}_j^2 \cdot (\mathbf{q}_i^1 - \mathbf{q}_j^2) \end{bmatrix}. \tag{4.3}$$

Here $\mathbf{u} \cdot \mathbf{v}$ is the scalar product between vectors. The points on the lines are found to be,

$$\begin{aligned} \mathbf{p}_i^{1,\text{closest}} &= \mathbf{q}_i^1 + \lambda_i^{1,\text{closest}} \mathbf{v}_i^1 \\ \mathbf{p}_j^{2,\text{closest}} &= \mathbf{q}_j^2 + \lambda_j^{2,\text{closest}} \mathbf{v}_j^2 . \end{aligned} \tag{4.4}$$

The distance between the points is $||\mathbf{p}_j^{2,\text{closest}} - \mathbf{p}_i^{1,\text{closest}}||$ which is defined as the triangulation error.

Software box 4.3

The calibration is implemented in the module `ptv3py.camera_calibration_module` where either the described calibration process or a classic pinhole camera matrix calibration can be performed using OpenCV functions.
The matching of particles is implemented in the module `ptv3py.particle_3d_matching` where the main function is:

```
def threedmatch_particles(
    particles: pd.DataFrame,
    camera_calibration: camera_calibration_module.
                                            CameraCalibration,
    max_error: float,
    camera_to_keep: int = 0,
    coordinate_limits: list = None,
    feature_column: tuple = None,
):
```

The `particles` parameter is a `pandas.DataFrame` with the output from the particle finding function and `camera_calibration` an object of a class described in the calibration module. The mentioned intensity matching is defined by the `feature_column` argument, see Table 4.1 for more information. Note that the implementation can only handle two cameras at the time of writing this thesis.

The challenge in the particle triangulation is that it is *not* known which two particle indexs i, j in each camera that corresponds to the same particle in 3D space. Here a matching algorithm is required. The matching algorithm used in this work attempts every single combination of positions in the two cameras. For each combination, a triangulation error is calculated, and all combinations create a triangulation error matrix $\mathbf{E}$. The row of the matrix corresponds to the particle positions found in camera 1 and the column camera 2. The index corresponding to the minimum value of this matrix (i', j') is found. If the triangulation error is smaller than a max triangulation error ε_{max} threshold the closest 3D point $(\mathbf{p}_{i'}^{1,\text{closest}} + \mathbf{p}_{j'}^{2,\text{closest}})/2$ is calculated. If this is inside the known coordinates of the illumination volume, the match is valid and added as a triangulated 3D point. The max triangulation error threshold is set from a calibration sensitivity analysis to $\varepsilon_{max} = 0.2$ mm as is detailed in the next section. After the match, all values in row i' and column j' of the triangulation error matrix are set as occupied to not be reused in a different 3D position. The procedure is repeated in finding minimum index and extracting the 3D particle position until the triangulation error exceeds the max triangulation error. This was the process for a single frame, and it is repeated for all recorded frames of the recorded event.

A final improvement to this matching algorithm can be implemented for this data since the particles have a large variance in size and therefore also imaged intensity. Since one particle should be imaged with roughly the same intensity in both cameras, the intensity can be used to guide the matching similar to what has been done in [81]. The intensity here refers to the integrated intensity of all pixels in the found particle island. The intensity guiding is implemented by first calculating a relative intensity error matrix ΔI where each element ΔI_{ij} is calculated by,

$$\Delta I_{ij} = \mathrm{abs}(I_j - I_i)/\max(I_j, I_i), \tag{4.5}$$

where I is the imaged intensity. The resulting matrix contains values between 0 and 1 of how similar the intensities of the possible match are. The triangulation error matrix is now redefined as,

$$\Delta M = \mathbf{E}/\varepsilon_{max} + \Delta I \tag{4.6}$$

This puts equal weight on the matching error and the relative intensity error that has been found to be suitable in the simulation in section 4.5. The new matrix is used in the algorithm above to improve the priority of the matching.

4.3.3 Particle tracking

The final step is to find tracks from the particles 3D positions[3]. The tracking algorithm used in this work is largely inspired by the Four Frame Best Estimate (FFBE) method [82] and an extension of how to initialize a track [83]. The tracking algorithm presented below is called the *Recursive FFBE* since the main change to the previous versions is the addition of a deeper recursion. First some notation is introduced. FFBE extensively use estimates of the velocity v_n and the acceleration a_n of a track which are calculated as follows,

$$v_n = \frac{x_n - x_{n-1}}{\Delta t}, \tag{4.7}$$

$$a_n = \frac{x_n - 2x_{n-1} + x_{n-2}}{\Delta t^2}. \tag{4.8}$$

Here, n is the relative frame of the track, x_n is the position of the particle at frame n and Δt is the time between two frames. The tracking use extrapolation to guess where the particle will end up in the next frame where either only the velocity is used in frame three of the track $\hat{x}_2$,

$$\hat{x}_2 = x_1 + v_1 \Delta t, \tag{4.9}$$

or both velocity and acceleration to perform this extrapolation $\hat{x}_{n+1}$,

$$\hat{x}_{n+1} = x_n + v_n \Delta t + a_n \Delta t^2. \tag{4.10}$$

---

[3] Note that this tracking algorithm works just as well for 2D tracks.

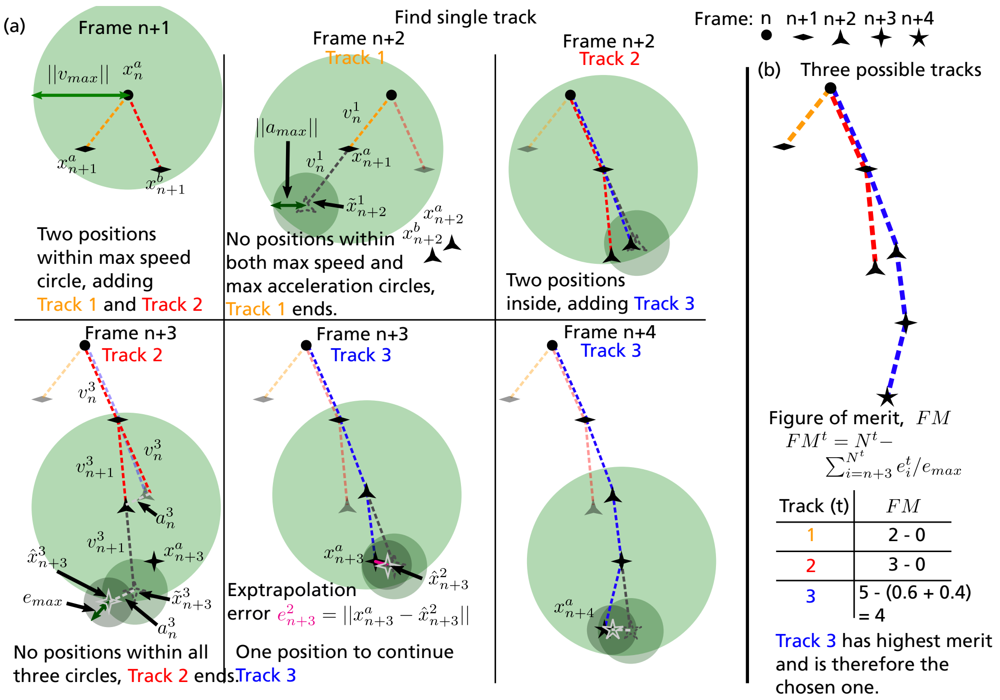


Figure 4.5: Illustration of the post-processing tracking from a single starting position. A recursion is initiated in the top left of (a) that in the first step locates all particle positions within a maximum speed $||v_{max}||$. For each of these a track is continued by extrapolating the track's next position using the estimated velocity. All positions inside a maximum absolute acceleration $||a_{max}||$ and maximum speed are here chosen recursively. In the next recursion step, both previous velocity and acceleration is used in extrapolation. Here, droplet positions closer than the maximum extrapolation error e_{max} and within the max acceleration and max speed limits is added to the recursion. This process is then continued until no more positions are found. With the full recursion tree in (b), the Figure of Merit (*FM*) showed on the bottom right in the figure is used to find the optimal track from a single starting position. An extension to the tracking and *FM* that can be used is described in the text.

After an extrapolation, a new position in the track is chosen and what is called extrapolation error e_{n+1} is the euclidean distance between the guess and the actual point,

$$e_{n+1} = ||x_{n+1} - \hat{x}_{n+1}|| \tag{4.11}$$

The FFBE method starts with a single position x_0 and generally no knowledge of where the particle is heading and how fast it is. Initially it was suggested that the second position in this track should be chosen as the nearest neighbouring position in the next frame. This simplification was found to be the most error prone process in the tracking [83]. Clark et al. then suggested that a recursive approach should be employed to attempt all particle positions in the first four frames with certain limitation of distance from the initial position. This approach is also employed in this work, however, instead of just performing the recursion in the starting four frames, the recursion is used all the way to the end of the track as is illustrated in Figure 4.5(a). From a position x_0, all positions in the next frame within a distance of *max speed* are tried in turn. The max speed is a value set to be appropriate for the tracking situation. For each of the tried positions x_1 of the track, a velocity v_1 is estimated and a position is extrapolated $\hat{x}_2$ using this velocity. All particle positions in the third frame of the track within the max speed limitation from before and now also a *max acceleration* limitation from the extrapolated position is tried in turn. For each of the tried positions, a new velocity v_2 and an acceleration a_2 is used to extrapolate a position $\hat{x}_3$. All particle positions in the fourth frame within the max speed limitation and max acceleration from before and finally also the *max extrapolation error* are tried in turn. From here on the recursive process of calculating new velocities and accelerations to find new positions within the three limitations is repeated. Here, a tree is built with all the possible tracks as branches from the starting position (the root) to the final position (the leaves) as seen in Figure 4.5(b).

From this tree of tracks, the most probable track (the tree trunk) should be found. To find this, each of the possible tracks are reduced to a single number called the track's Figure of Merit (*FM*), The *FM* of a track from the root to one of the leaves is calculated as the length of the track subtracted by the sum of the extrapolation errors divided by the max extrapolation error. A larger *FM* is better which means that it is defined that longer tracks are better, as long as the extrapolated position of the track is close to the found positions. The track with the largest *FM* is chosen as the most probable track from this starting position. An extension has been added to the tracking procedure since it was found that some tracks have frames with a missing triangulated value. The value could be missing for various reasons such as particles that are leaving and re-entering the illumination volume or through an error in the matching. Then the tracking would wrongly produce two tracks instead of one. To solve this, the possibility to jump frames in each recursion step is added where the track still must follow the speed, acceleration, and extrapolation limits. An input variable called *max jumped frames* decides how many frames in total a track is allowed to take. Jumping should only be used when necessary and will therefore reduce the Figure of merit by 1 for each jump. The figure of merit with the extension is found below.

$$FM^t = N^t - \left(\sum_{i=n+3}^{N^t} e_i^t / e_{max} \right) - N^t_{\text{jumps}}. \tag{4.12}$$

The tracking procedure introduced in the previous paragraphs gives an algorithm for finding a track from an already decided starting point. However, how should the algorithm know which starting points are the best ones to use. A simple solution is to start with the earliest positions in the recording and then continue. This, however, implies that tracks starting earlier are more important in the case of a conflict where two tracks want one particle position which is not always a good heuristic. It is also not clear which track to favor in a conflict when two tracks start in the same frame, see Figure 4.6. The suggested solution is here to let the mentioned Figure of Merit decide. The next step organizes the chosen starting

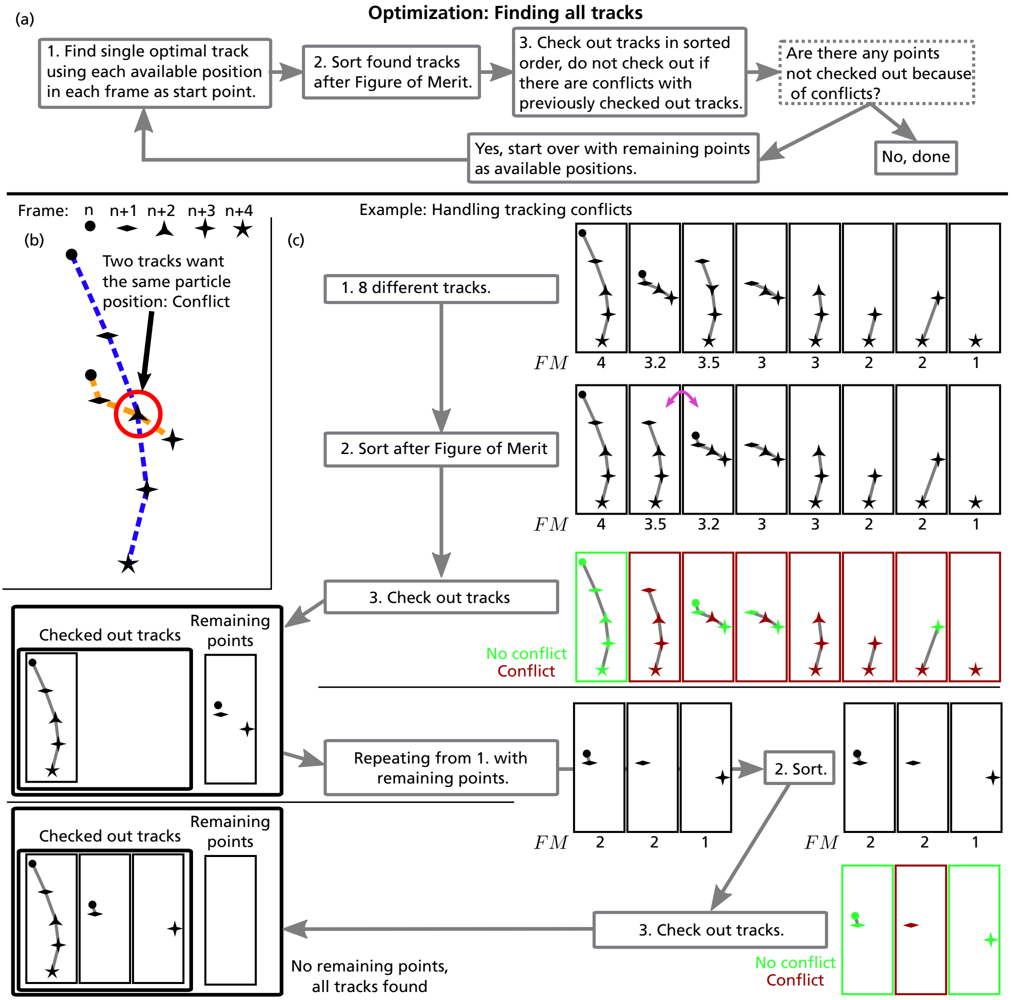


Figure 4.6: Illustration of the optimization algorithm to decide which starting positions are the most probable according to the previously defined Figure of Merit. A flowchart is shown in (a) of how tracks are found and prioritized. In (b), an example is shown where there is a tracking conflict and (c) how the optimization handles the situation. The process of finding an optimal track from a single starting position is described in Figure 4.5.

points as shown in the flowchart of Figure 4.6(a). First all positions are in turn selected as starting positions and then a track is found from that position as previously explained. Then, the tracks are sorted according to their *FM* value. In descending order, the tracks are extracted as final tracks if they do not include any conflicts with the previously finalized tracks. The tracks with conflicts are reprocessed until all positions have a final track.

The algorithm explained here to find tracks can be quite computationally intensive. The limitations of max speed, max acceleration and max extrapolation error are important for reducing the number of branches in the tree extensively. These parameters should be chosen

Table 4.1: The used input parameters for the 3D-PTV algorithm. The parameter names correspond to the names of input parameters found in the python package `ptv3py`.

Finding particle coordinates		
threshold	25 counts	
relative_gradient_threshold	True	gradient threshold is multiplied with the previous intensity in the breadth first search
gradient_threshold	0.2	if the intensity increases more than 20% compared to the previous intensity, the breadth first search stops.
Triangulation		
max_error	0.2 mm	
coordinate_limits	[[−50, 50], [−80, 80], [−15, 0]] mm	the limits of the illumination volume
feature_column	("intensity", 1, "relative")	here is the instruction to use particle intensity in the matching with weight 1 and relative comparison as described in equation 4.5
Tracking		
max_speed	8 mm/frame	
max_acceleration	1.4 mm/frame	
max_extrapolation_error	1 mm	
max_jumped_frames	1	
min_track_length	6	

with care by the person using this tracking algorithm. Too large numbers will give high computational cost and possibly more erroneous tracks compared to stricter limitations in the form or narrower limits. To further optimize the processing, the implementation of this tracking in the python package ptv3py, use dynamic programming. Here, all found tracks from a single starting point are stored and for new tracks it is checked whether the current track has the same last three positions as a previously found track. Then, the ending of the branch will be the same and a recalculation is not required. This optimization saves a significant amount of computation time since there are many overlapping tracks with the proposed algorithm.

4.4 Velocity results

The two experimental configurations have been applied to track respiratory droplets. Results from the horizontal configuration can be seen in Figure 4.7. The advantage of this configuration is that one can assume that all exhaled droplets cross the illumination and will be detected. One measurement each of the respiratory event types speaking, yelling, and coughing is here shown. The speaking and yelling were as mentioned of the phrase "Protect better against COVID". For each event there are multiple ways to present the result that show the diversity of how this measurement technique can be used for under-

Software box 4.4

The tracking of particles is implemented in the module `ptv3py.particle_tracking`. The following function is used for both 2D and 3D tracking:

```
  def recursive_4frame_best_estimate(
    particles,
    min_speed: float = 0,
    max_speed: float = -1,
    max_acceleration: float = -1,
    max_extrapolation_error: float = -1,
    max_jumped_frames: int = 0,
    min_track_length: int = 4,
    ...
):
```

There is an alias name for this function called `track_particles` and there are more input parameters to this function found in the source code.
One of the main drivers in the tracking algorithm is the `class Parameter_derivative_tracking:` that keeps track of the position, acceleration, extrapolation errors and feature of merit (called `score` in the code) of each branch in the recursion. It can also be used to track other parameters than coordinates of a track such as intensity that was used in the triangulation matching. However, it is challenging to choose weights in the feature of merit calculation between the columns such as intensity and the coordinates.

standing droplets. The purest presentation form of the results are the 3D tracks that are shown in (a), (b) and (c) for each event type in Figure 4.7. The colour of the tracks represents the estimated speed of the droplet according to the colorbar at each time point which is calculated by the droplet 3D displacement divided by the time between frames. A general downward angle can be seen for all three events and the fastest droplets are most commonly found in the centre of the found tracks. The speaking and yelling cases have a relatively larger count of slower droplets with larger variation of direction compared to the coughing case. This variation can be explained by how the speaking and yelling have continuous variation of exhalation speed during speech that give rise to complex air patterns together with the fact that the speech events are around six times longer than coughs. For all three events, there are no found droplets in the top right area of the 3D volume as is indicated in red for the cough. This area is masked when finding droplets because of the illuminated face in the background. The illumination is here used as assistance to see when the respiratory event starts and for speaking and yelling it is extra important to locate when the specific words are pronounced. In the future one can improve the measurements by remove the background light and replace it with an audio recorder that is synchronized with the recording.

From a found track the average of all inter-frame velocities represent how fast the droplet was travelling through the volume. This average speed will in the worst case have an uncertainty of around 1.5 m/s given the estimated position uncertainty found in the next section to be 0.2 mm and the median track length of 10 frames. However, this worst does not seem to be representative since a variation found from the actual tracks give an estimated standard deviation of 0.3 m/s (corresponding to a position uncertainty of around 0.04 mm. The difference in found uncertainty is probably caused by a combination of that the worst-case scenario in the simulation is not representative and that the tracking does not allow tracks that have too high uncertainty in position.

The average droplet speeds can be analysed in the temporal domain that is shown in Figure 4.7(d). Here, both the temporal evolution of average speed and number of found droplets are shown. A general correlation of number of droplets and speed is visible. This can be explained by how more droplets are produced by a larger air flow that will also accelerate droplets to a higher speed. The larger air flow is also found for yelling that have a substantially higher count of produced droplets. The increased production of droplets with speech loudness have been previously reported [50]. The speaking case show a relatively large speed and number of droplets for the letters P in ”protect” and B in ”better” and the yelling mainly for P. These are letters that together with ”r” and ”t” have been found to produce more droplets [47]. The cough shows two temporal peaks of produced number of droplets during the event. A cough is expected to have a consistent high exhalation air speed and the change in number of produces droplets then suggests that there are other drivers for droplet production than air speed. One explanation can be the availability of saliva in connection to the air stream. A study on singing noted that the sense of accumulated saliva coincided with increased droplet generation [47]. Finally, the average speed for all found tracks is visualised as a histogram in used in Figure 4.7(e). The average speeds are 4.2, 5.1 and 8.3 m/s for the three different events. It is expected that the speaking has the slowest droplets and coughing the fastest and using this technique the actual difference in speed can be quantified. The results shown here is a subset of what was used in Paper v where all results can be found and downloaded at `https://doi.org/10.17605/OSF.IO/R7YBD`.

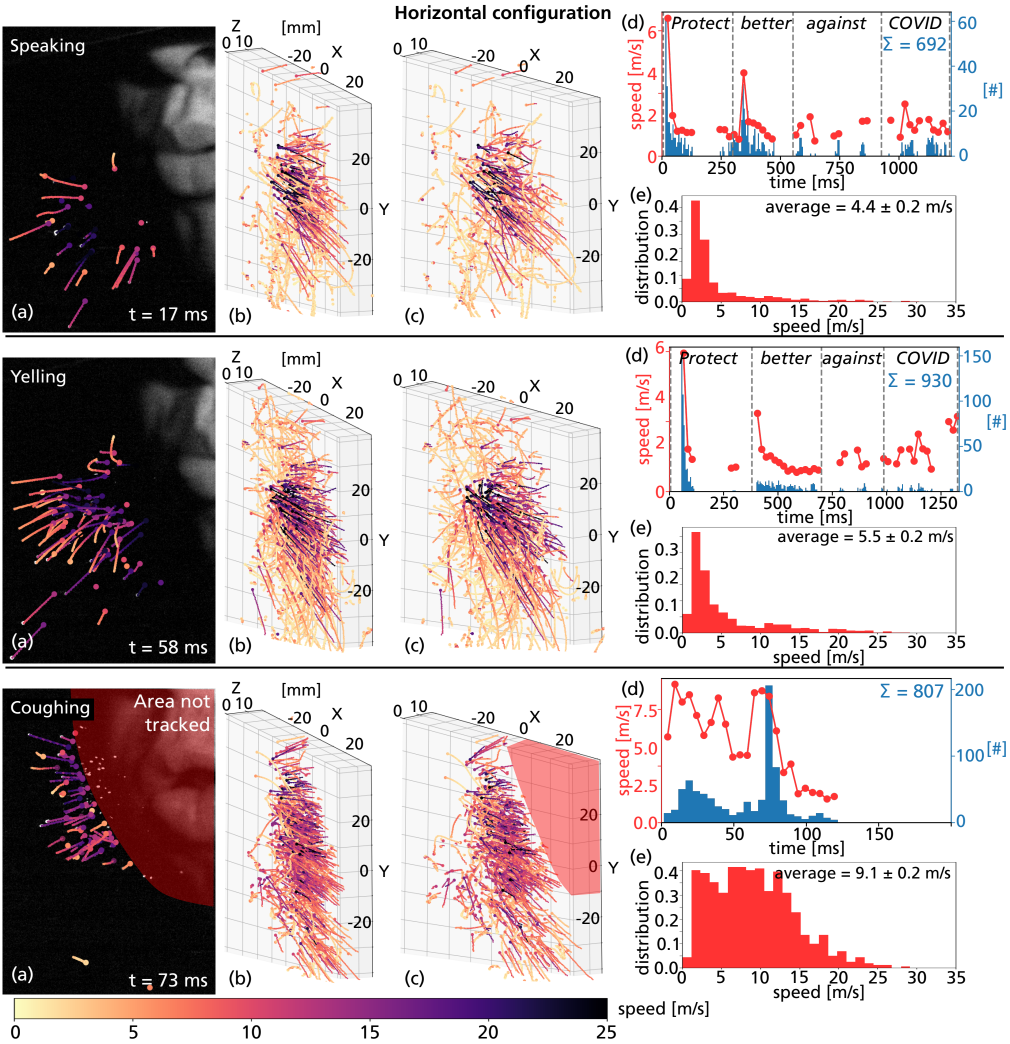


Figure 4.7: Velocity results using the horizontal configuration. Three different kinds of respiratory events are shown including speaking, yelling, and coughing from top to bottom. For each case multiple ways of visualizing the same data are presented to show what information 3D-PTV can give from respiratory events. (a) show the tracked droplets, projected into camera C2, found at a single timepoint. (b) and (c) show all tracks found for the whole event duration in 3D from two angles. For the cough, the red area indicates where no droplets are found due to masking of pixels. (e) show the time evolution of the average speed and number of found droplets. Note here that the actual time resolution is at the camera frame rate with 67 μs between frames and the temporal data is here accumulated into bins of 10 ms width. Finally, (d) show the distribution of estimated average speeds from these droplets.

The horizontal configuration has the disadvantage of only capturing the droplets over a short distance and time that makes it harder to extract information of droplet trajectories. The vertical configuration is here more advantageous where example results of a cough can be found in Figure 4.8. The vertical arrangement posed challenges in capturing the respiratory droplets since the illumination volume is quite thin together with how the subject was left some freedom to perform naturally. For speech events, the movements of head and mouth resulted in variation of exhalation direction where only parts or none of the droplets hit the illumination. This means that the vertical configuration is not suitable for analysis of such events. Instead, coughing events were found to be more suitable even though the aim of the cough was still important. The long but thin illumination volume enables tracking of droplets for up to 12 cm as can be seen in panels (a), (b) and (c) of Figure 4.8 which are illustrated similarly as results from the horizontal configuration. Some longer tracks show clear deceleration over the droplet trajectory. This is also seen in panel (d), where the spatial average of droplet speed and direction have been estimated for different 3D positions. Along *Y* direction, the largest speeds are found close to the mouth and a reduction of speed is seen further downstream that indicate the transition from jet to puff phase. The average

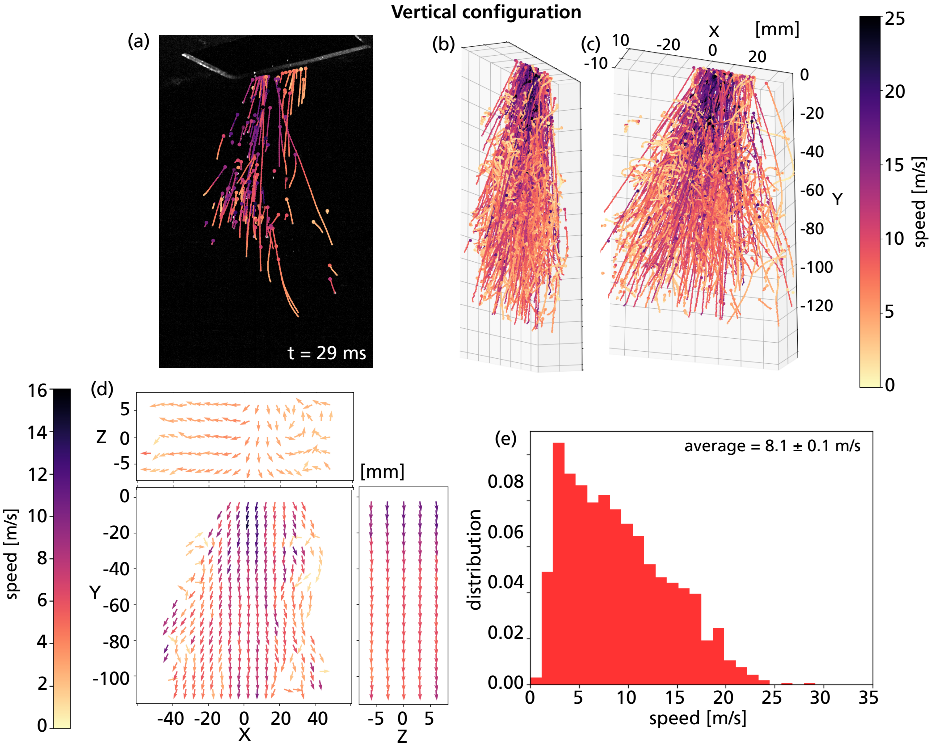
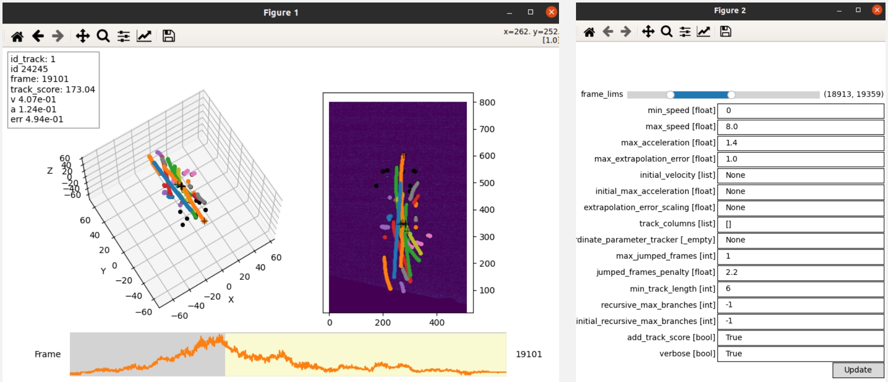


Figure 4.8: Results of a single cough using the vertical configuration. (a), (b) and (c) are similarly visualised as Figure 4.7. (d) show a spatial averaging of all tracks and at all time-points for different cross sections through the data. (e) show the distribution of the average speed of all tracks found in this cough.

Software box 4.5

The visualisation of the resulting tracks is important to validate the results and the input parameter to the PTV algorithm. Multiple functionalities for this is implemented in the `ptv3py` package. Below is an example view of the visualisation created by the `ptv3py.particle_tracking_visualization.particle_tracking_parameters_helper` function. Here, one can see the tracks in 3D and projected 2D, interactively move through frames in the recording and try different input parameters to optimize the tracking for the current situation.



id_track: 1
id 24245
frame: 19101
track_score: 173.04
v 4.07e-01
a 1.24e-01
err 4.94e-01

x=262. y=252.
[1.0]

frame_lims	(18913, 19359)
min_speed [float]	0
max_speed [float]	8.0
max_acceleration [float]	1.4
max_extrapolation_error [float]	1.0
initial_velocity [list]	None
initial_max_acceleration [float]	None
extrapolation_error_scaling [float]	None
track_columns [list]	[]
rdinate_parameter_tracker [_empty]	None
max_jumped_frames [int]	1
jumped_frames_penalty [float]	2.2
min_track_length [int]	6
recursive_max_branches [int]	-1
initial_recursive_max_branches [int]	-1
add_track_score [bool]	True
verbose [bool]	True

Similar visualisation helpers are also implemented for finding pixel coordinates and 3D matching particles where either the input parameters can be tuned or just the results are shown.

speed is calculated for each track and the distribution of these speeds is found in (e).

These are examples of what can be extracted and there are more possibilities such as finding spreading angles, droplet cloud evolution that this experimental data enables. Due to the small number of recorded events and few subjects it is hard to draw statistically significant conclusions currently. However, with a rigorous study of multiple subject and events for each subject, tracking results using this technique can add valuable information for a better understanding of respiratory droplets. Before those measurements, it is of great importance to perform a validation of the tracking technique.

4.5 Evaluation of the 3D-PTV algorithm

The evaluation is performed for each of the three steps of the 3D-PTV algorithm in turn.

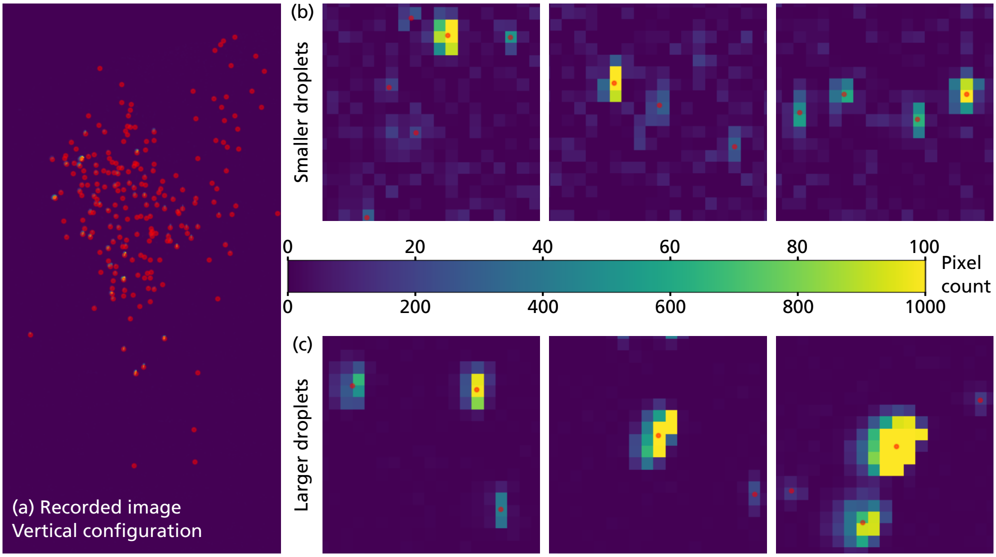


Figure 4.9: Example droplet scattering footprints together with the estimated weighted centroid positions as red dots in the recorded image shown in (a). The long camera exposure relative to the speed of the fastest droplets result in elongated shapes. In addition, a large variation of droplet size transfers to large variation of scattering intensity and footprint sizes. Note that the colorbar in (b) is 10x shorter than the one in (c).

4.5.1 Particle finding evaluation

The finding particles approach use a combination of a threshold binarization, connected components and weighted centroid to estimate pixel coordinates. As alternative to the weighted centroid, gaussian fitting using both iterative least squares and the three-point algorithm were attempted [84]. Both methods were found to be insufficient for the particles imaged in this experiment. The main problem for the gaussian is the large variation of particle footprint sizes and shapes caused by the long exposure time of the cameras relative to the particle speeds and variation in particle scattering intensity as seen in Figure 4.9. Anisotropic gaussians with and without rotation were also attempted but the extra parameters did even more damage. The weighted centroid algorithm with its high reliability and simplicity was chosen as the best option for this data. The problems connected to long exposure can be reduced with a pulsed laser that freezes the motion of the droplets during an exposure. However, this requires an advanced laser source that can produce pulses at the recording rate of the camera and will increase security challenges to protect the subject ejecting droplets.

The pixel value discontinuity overlap detection algorithm described by Maas et al. [79] was implemented to handle overlap cases mainly in the areas close to the mouth of the person performing the respiratory event. Note that there are high degrees of overlap that the algorithm does not detect. This missing position will propagate as errors in both the triangulation and tracking and further validation of its impact should be made. Then comparison to other overlap detection schemes such as neural networks [85] or particle mask correlation [86] can be made for this data.

The following validation scheme by use of simulation of the particle finding and overlap detection algorithm is suggested. First extract a large number of particle footprints from the data such as the ones found in Figure 4.9 that will represent the different situations that the finding algorithm is facing in the actual data. The particle positions from each footprint is then assumed to be the weighted centroid value. Note that this is an assumption since the particle position of a footprint is not actually known. An alternative is to use a predefined optical transfer function, such as a gaussian [87], as footprint for all particles. However, then the large variation of particle footprints that are found in this data is not as accurately taken into account in the simulation. The database of footprints should be manually verified to avoid usage of footprints that are actually overlapping particles. Now random footprints are given to sampled particle pixel positions and interpolations of the footprints is used to simulate subpixel particle positions in the images. The finding algorithm's sensitivity to overlap can then be evaluated by using different distances between two or more simulated particles. In addition, images with different number density of particles can be simulated to find at what density the particle finding algorithm deteriorates.

4.5.2 3D triangulation evaluation

The 3D triangulation is challenging because of the limiting experimental situation. The first limitation is that there is a requirement of high-speed cameras to temporally resolve the movement of droplets and only two were available at the time of the measurement. Typically, three or more cameras are employed to reduce problems with 3D position uncertainty and number of found ghost particles[4] [12].

A second limitation is connected to the possible positions that were available for the cameras. The cameras must be positioned on the same side of the illumination volume. On the other side of the illumination is the protection plate for the subject ejecting droplets. The cameras should also preferably be positioned to have the same approximate scattering angles. Then, similar scattering intensities will be recorded in the two cameras that give the same limit on minimum detectable droplet size. In addition, this simplifies the use of scattering intensity in the matching for triangulation. The resulting positions of the cameras was at a 52° semi-forward detection angle with 23° between the cameras. The angle between the cameras should in a stereo detection setup be 90° to optimize the triangulation precision and an increase of this angle is suggested for future experiments. Advanced techniques for estimating particle 3D coordinates such as Iterative Particle Reconstruction

---

[4] Ghost particles are particles found in the triangulation even though they do not exist in the actual 3D space.

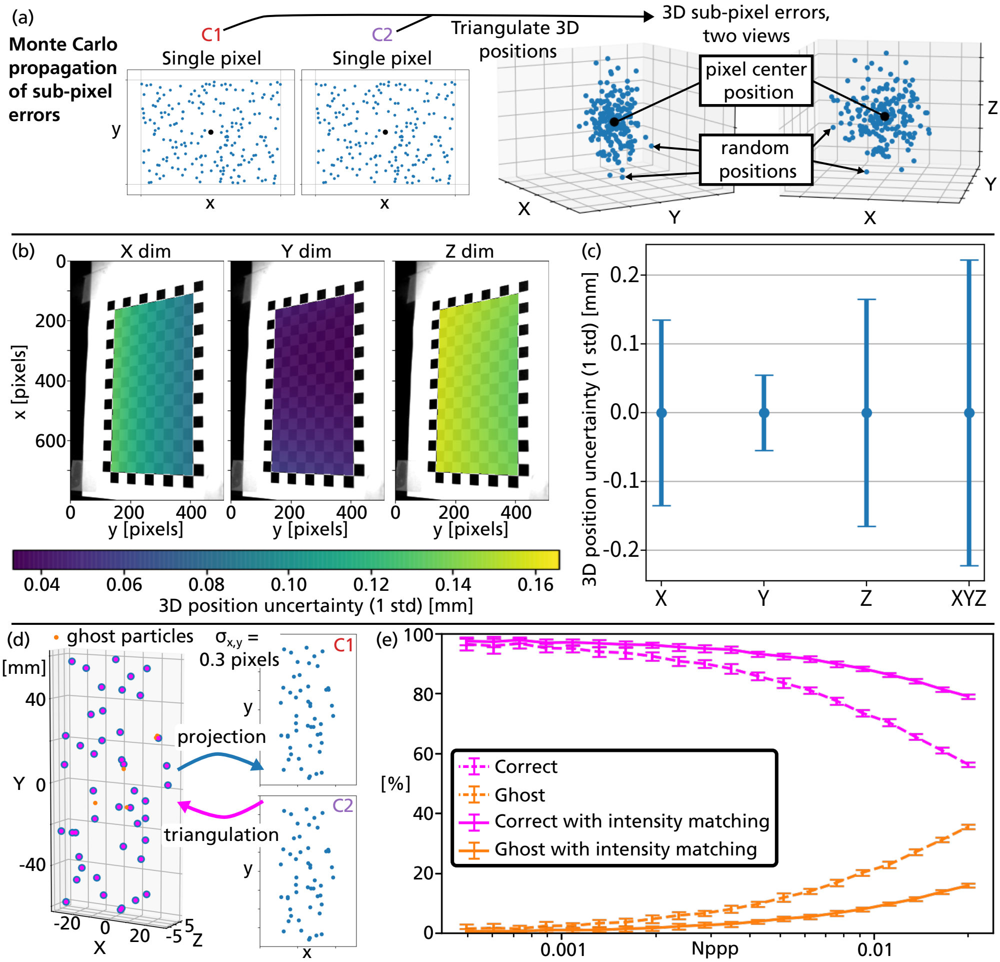


Figure 4.10: Simulation analysis of the 3D triangulation sensitivity and ghost particle detection. In (a), the Monte-Carlo approach used to estimate the 3D triangulation sensitivity is illustrated. Panels (b) show the sensitivity results for $Z = 8$ mm in the three different dimensions. The maximum sensitivity in each dimension and the euclidean norm of all dimensions is found in (c). In (d), the process of estimating how particle number density affect the number of found ghost and missing 3D positions with the results shown in (e).

[87] have not been attempted. It would be interesting to see how these techniques cope with the large variation of footprints in combination with only using two cameras.

Two simulations have been performed to understand the limitations of 3D triangulation using this experimental setup. The first simulation estimate the 3D position sensitivity using a Monte-Carlo approach, illustrated in Figure 4.10(a). The main assumption here is that the particle pixel position error is uniformly distributed ± 0.5 pixels which corresponds to the worst case when the particle is only found in a single pixel and also assumed that there is no particle overlap. A known 3D point is now projected into each camera to receive two pixel positions. The pixel positions are randomly displaced 10 000 times by sampling pairs of displacement vectors from the mentioned uniform distribution. All the pairs of random positions are triangulated and the standard deviation error to the correct 3D point describes the sensitivity of the setup to errors in found particle pixel coordinates. The process of estimating sensitivity is repeated for a range of 3D coordinates inside the illumination volume and resulting sensitivities are shown in Figure 4.10(b) and (c). Generally, a larger sensitivity is found for 3D positions further away from the cameras as is seen in (b). In (c), the lowest sensitivity is found for the triangulated Y coordinate which is explained by how estimation of the Y coordinate have a small connection to the distance between the cameras and the 3D position. This is in contrast to the X and Z dimension where the distance error is propagated through a sine and cosine of the semi-forward detection angle respectively. The euclidean norm of the maximum sensitivities is found to be around 0.2 mm which has been used as guidance for the max matching error parameter to the 3D triangulation algorithm. Note that an increase of angle between the camera views (β) is expected to reduce the triangulation sensitivity proportionally to $1/\sin(\beta/2)$ which is calculated from a simplified geometrical model. β equal to 45° and 90° will results in halved and one third sensitivity, respectively. To decrease the sensitivity further, more cameras must be used.

The second simulation, illustrated in Figure 4.10(d), is performed to estimate how well the matching during triangulation performs under different particle seeding densities $\mathrm{N_{ppp}}$. This will give information of how many particles are correctly triangulated and how many ghost particle are found. A specific number of 3D positions, corresponding to a known seeding density, are sampled inside a 3D volume with the same size as the laser illumination volume ($60x120x15$ mm^3). The positions are projected into the cameras, and they are randomly displaced by a uniform distribution ± 0.5 pixels. The pixel positions are used with the exact same matching and 3D triangulation algorithm as has been detailed before firstly without and then with intensity matching. The process is repeated 20 times to extract the average performance and a standard deviation of the result as is shown by the errorbars in the plots. As expected, there is a typical decrease in correctly found tracks and increase in number of found ghost particles with increasing particles density $\mathrm{N_{ppp}}$. Note here that the matching largely profits on that the illumination volume is known, all found possible matches that triangulate to 3D positions outside this volume can simply be discarded which improves the results significantly. One post-processing solution that is implemented to improve matching is the use of imaged particle intensity as guidance. To include this in the simulation, one intensity was sampled for each 3D particle by inverse transform sampling from the estimated cumulative distribution function of found particle intensities. Then, independent normal noise with standard deviation at 12% of the intensity[5] was applied to each camera intensity of the same particle. The use of intensity as guidance in the matching is clearly beneficial in both reducing the number of found ghost particles and increasing the number of correctly matched 3D positions.

---

[5] This intensity uncertainty was found in Paper IV when estimating particle size from intensity.

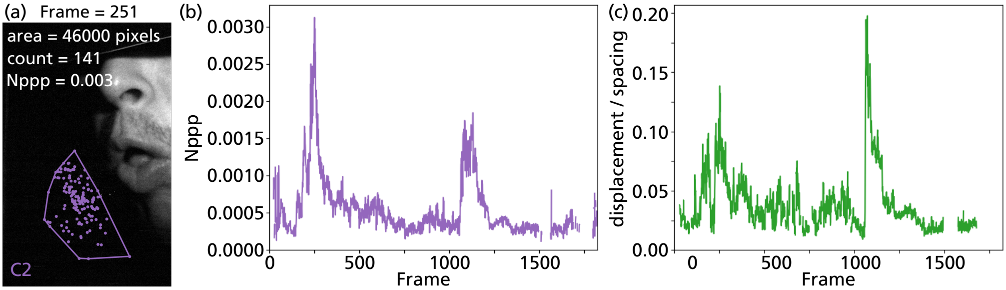


Figure 4.11: Particle density N_{ppp} and displacement spacing ratio for coughs, the most violent events in this work. In (a), an example frame for estimation of particle density is shown and (b) show the evolution of the particle density for the cough results shown in Figure 4.7. (c) show the displacement spacing ratio that has been previously used in tracking validation since it can be used to predict tracking error [83].

The number of particles per pixel N_{ppp} have been estimated for a single cough in Figure 4.11(a) and (b). A cough was chosen here since it is the most challenging of the three respiratory events that have been recorded. Since this is not a homogeneously seeded volume of particles, the calculation of N_{ppp} here also estimates an area of the particles for each frame as the convex hull of all found particles. As is visible in the example calculation in panel (a), there is a significant variation of density inside this area which means that the found N_{ppp} is underestimated. In the example image, one can approximate that the highest density is around 50% higher than the found value. In panel (b) of Figure 4.11, the estimated N_{ppp} is found for each frame of the cough. The peak value is at around 0.003 which is found to be the maximum N_{ppp} for most cough events where the exceptions have a lower maximum N_{ppp}. With the 50% addition to a N_{ppp} at 0.0045 as a worst case, the rate of correctly matches is around 93% and the number of found ghosts reach around 5%. One can reduce the number of ghosts by reducing the maximum triangulation error, however, that will also reduce the number of found correct 3D matches. The ghosts and lost 3D positions will challenge the tracking procedure in producing correct tracks.

4.5.3 Tracking evaluation

The Four Frame Best Estimate algorithm was chosen for its simplicity and its intuitive approach for tracking. After implementation, it was found that there were problems with ghost particles that required an improved tracking. The natural step to improve the algorithm was to continue along the path that Clark et al. [83] had initiated with recursion. The extended recursion attempts more possible paths for each track. This extension was possible with respect to computational time because of the relatively few particles together with the dynamic programming approach. The challenge was to create the feature of merit function that can correctly differentiate between good and bad tracks. It was found that the length of the track was an important feature. A longer track that followed the extrapolated guesses within the boundaries of max acceleration and max extrapolation error was more

Software box 4.6

The estimation of the temporal change of particle density in the images can be found in the function:
`ptv3py.particle_finding.estimate_particle_density`

A similar function for the displacement/spacing ratio is found in:
`ptv3py.particle_tracking.estimate_displacement_spacing_ratio`

often correct. Then, as mentioned, to decide which of two tracks with equal lengths is best, it is natural to choose the one that best follows the extrapolated coordinates. Another extra complexity implemented by Clark et al. is to give a starting velocity to all tracks. This should be suitable for this data since most droplets travel in the same direction. It has been implemented in the `ptv3py` package but no major improvement could be seen with it and therefore this feature was not used. No simulation has yet been performed to evaluate the tracking algorithm developed here. However, Clark et al. performed an evaluation of their extension to the FFBE tracking. They evaluated their tracking performance based on the so-called displacement spacing ratio of the 3D tracks. Generally, larger displacement of particles between two frames divided by the spacing between particles correspond to a harder tracking problem and vice versa. This ratio has been estimated for the tracked coughed droplets and the results are found in Figure 4.11(c). The maximum value of 0.20 is representative for most coughs and it corresponds to less than 2% of wrong tracks. Note that no ghost particles nor missing 3D triangulated values was used in this simulation. In addition, our algorithm is expected to have better performance than the one presented by Clark et al. since it as mentioned attempts more possible paths to avoid infiltration of bad positions. A suggested simulation scheme for the tracking is to use 3D particle tracks simulated by the John Hopkins Turbulence database [88]. The tracking performance can then be evaluated by randomly adding ghosts and removing 3D position to see how it affects the tracking. A more advanced simulation can also be performed by use of the full 3D-PTV algorithm. The 3D positions are then projected into the cameras through the camera calibration and then recorded PTV images are simulated, suggestively using the particle footprint database previously suggested. The errors of each step are then propagated to extract the performance of the whole process.

4.6 Quantifying and comparing efficacy of facial masks

In the early part of the COVID-19 pandemic there was a lack of protective medical equipment in Sweden which included facial masks. This led to imports of brands from previously

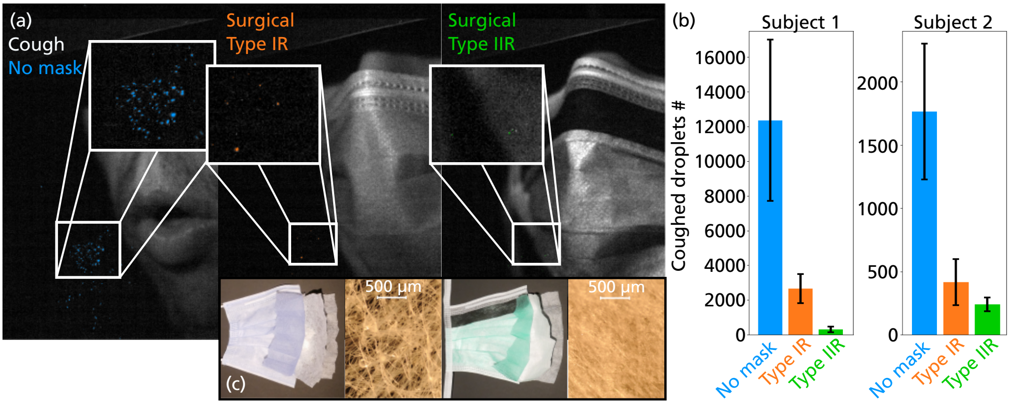


Figure 4.12: Comparison between number of produced droplets for coughing in the three cases: without mask, with a low-cost mask and with a hospital graded mask. Example images of the three cases are shown in (a). The average and standard deviation of counted droplets are shown in (b) and microscope images of the middle layer of the two different masks are shown in (c).

unused sources to fulfil the increased demands. To better understand the quality of these masks an investigation was performed with coughing subjects to understand how well the cheapest masks found in the supermarket in autumn 2020 (surgical mask type I) compares to the masks used in the hospital (surgical mask typ IIR). The same setup was used here as is detailed in Figure 4.2 with one difference that the laser was focused to a 1.5 mm thickness along the Z dimension instead of the 15 mm previously used. With this setup there is a 10 times higher laser fluence which means that smaller particles can be detected. From knowledge of the sizing calibration found in the next section, the calculation $10^{0.5} \approx 3$ give the approximation that droplets three times smaller can be detected using this thickness compared to the 15 mm thickness. Given a slightly higher threshold at 40 counts for finding droplets, this corresponds to a minimum detectable droplet size of approximately 1 µm. Two subjects were asked to cough in three different cases, when wearing no mask, the type I mask, or the type IIR mask. Example images of the high-speed recordings are shown in Figure 4.12(a).

The droplets are counted by finding areas in the images in the same way as explained in section 4.3.1. Here, no correction is performed for overlapping particles since the thin illumination volume significantly limits their occurrence. The recursive FFBE tracking algorithm is also used to find slower droplets that stay in the illumination over multiple frames. The detected droplets are counted where the tracks are counted as only one droplet. This approach to identify face mask efficacy has also been used in [46]. Average result for each case is found in Figure 4.12(b) that includes four coughs for subject 1 and six cough for subject 2 per case. The error bars show the standard deviation calculated from the four and six events, respectively.

Initial observations of the results show that subject 1 on average produce around 7 times more droplets than subject 2 and that there is a substantial variation between cough events. For both subjects, the average number of droplets is reduced when using the type IIR mask compared to the type I mask shown in Figure 4.12(c). An improved filtration can be explained by the denser middle layer of the type IIR mask compared to the type I mask. The estimated expectation of filtration efficacy for both masks and subjects are found in Table 4.2. A single sided independent Welch T-test is applied to test if the type IIR mask filters droplets better than the type I mask with p-value below 0.05. The test shows a significant decrease for subject 1 (p=0.005) however no significance for subject 2 (p=0.054). More measurements are required to further quantify the mask efficacy.

The facial masks reduce the number of droplets that is ejected when coughing that should reduce spreading of disease as has been clear for a long time when used in hospital environments. During the pandemic, recommendations, and obligations to wear facial masks were introduced in large scale in hope of reducing spreading of disease. Individual case studies have shown correlation between regulation on facial masks and reduced spreading. However, the evidence for causality between large scale obligation of facial masks use and reduced spreading is questioned by a review of studies concerning whether face masks during COVID-19 published by the Cochrane Library [89].

Table 4.2: Estimated filtering efficacy for the two facial masks tested. Note that the expected value of efficacy is estimated as $E[\#\text{mask}/\#\text{no mask}] = E[\#\text{mask}] \cdot E[1/\#\text{no mask}]$ assuming independence between cases for the number of estimated droplets. Here, # defines the number of detected droplets. The uncertainty of each expected value is estimated from the data through for the two factors and propagated through the multiplication to get the final uncertainty.

	Type I	Type IIR
Subject 1	0.74 ± 0.04	0.98 ± 0.01
Subject 2	0.82 ± 0.02	0.92 ± 0.01

4.7 Sizing respiratory droplets

The scattering intensity from respiratory droplets can be used to estimate the droplet diameter, here used as equivalent notation to droplet size. The sizing technique is presented in great detail in Paper IV and will be explained briefly here.

4.7.1 Sizing calibration procedure

The experimental setup used for the sizing calibration apply the horizontal configuration with the same laser illumination as the velocimetry measurements. The main difference as seen in Figure 4.13(a) is the addition of a red diffuse light source together with a high-resolution telecentric objective with an attached red band pass on the C1 camera. Together, the red light and top camera form a shadowgraphy imaging configuration (C1S) where

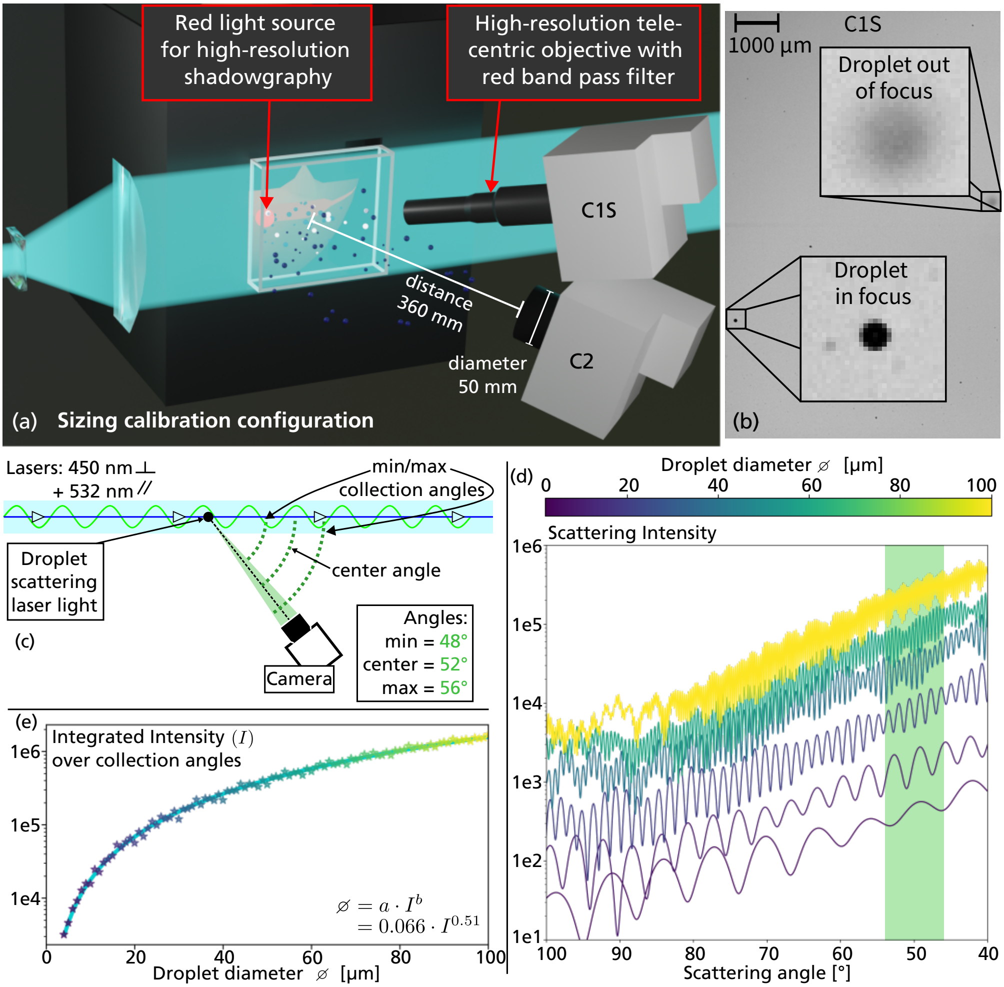


Figure 4.13: The experimental setup for the sizing calibration and the droplet sizing approach. The setup in (a) is almost identical to the one used to find velocity with the major difference of adding a high-resolution telecentric lens and a red background light that together form a shadowgraphy setup to size droplets. An example shadowgraphy image is shown in (b). An illustration of the sizing setup and theoretical scattering intensities are shown in (c), (d) and (e).

droplets imaged in focus can be sized as seen in Figure 4.13(b). Camera C2 image the scattered light as previously and the sizing approach is based on the fact that larger droplets scatter more light as is illustrated in Figure 4.13(c), (d) and (e). Here, simulated scattering intensities are used based on Lorenz-Mie scattering theory [90, 91]. A theoretical calibration curve fit well to the following parametric function,

$$\varnothing = a \cdot I^b, \tag{4.13}$$

where $\varnothing$ is the size (diameter) of the droplet, I is the scattering intensity and a, b are the parameters to fit.

Calibration droplets are now required to scale the theoretical curve correctly for this experimental setup. The experimental configuration is then used to record an event of a person saying the letter ”V” repeatedly to produce a large number of droplets. Calibration droplets are found that are detected on both the cameras. Only droplets that are found to be sufficiently in focus are used where the sizing employs an automatic algorithm that take a rough position of the droplet in the shadowgraphy image and estimate its size. This automatic algorithm applies fitting of a circle to points that have been interpolated to extract the edge of the droplet. The smallest size that the shadowgraphy can measure is around 30 µm (3 pixels) diameter. Below this, the imaging conditions cause underestimation of the droplet size[6]. The recorded intensity and the size of an example calibration droplet is found in Figure 4.14(a). The scattering intensity of droplets are often spread over multiple pixels in C2 even though all droplets are smaller than the pixel size. The reason for this is droplet movement during the exposure time and small out of focus effects. The sum of intensities in all these pixels is calculated to extract the imaged scattering intensity of the droplet. This intensity is varying when the droplet travels through the illumination volume. There are here often peak intensity values when the droplet enters and exits the illumination. The peaks were first thought to be caused by the variation of intensity at different scattering angles. This theory was tested using the same size and trajectory scattering angles as was found for the calibration droplet to simulate its theoretical scattering variation. This intensity is shown as the green curve in Figure 4.14(a). There are no clear variations of this curve that can explain the peaks. The peaks are instead dealt with by calculating a median of all intensity values above half the maximum value. The resulting intensity is found to be robust in removing the effect of these peak intensities.

In total 65 calibration droplets were found where a valid correspondence between intensity and size is here called a calibration point. The calibration points are plotted as black dots in Figure 4.14(b). Here, red dots are calibration points that saturated the sensor and was therefore invalid. The black points are used to estimate a scaling coefficient $\hat{a}$ to connect the simulated Lorenz-Mie intensity to the imaged intensity for this experimental setup. This is achieved by the least squares estimation of the following function,

$$\varnothing = \hat{a} \cdot (a \cdot I_c^b). \tag{4.14}$$

The fit gives $\hat{a} \approx 9.9$ and the blue calibration curve shown in Figure 4.14. The minimum possible intensity is 25 counts which corresponds to the minimum diameter of 3.4 µm that this calibration curve can size. Note that in Paper IV one calibration curve is calculated for each pixel because of changing scattering angles over the image. Since the 3.4 µm is found for a curve in the centre of the image, the minimum diameter that can be sized is generally smaller than 3.4 µm in the left and larger than 3.4 µm on the right part of the image. In addition, limiting size is larger for moving relative stationary droplets where

---

[6] The sizing limit was found by testing the sizing algorithm on a 1951 USAF chart.

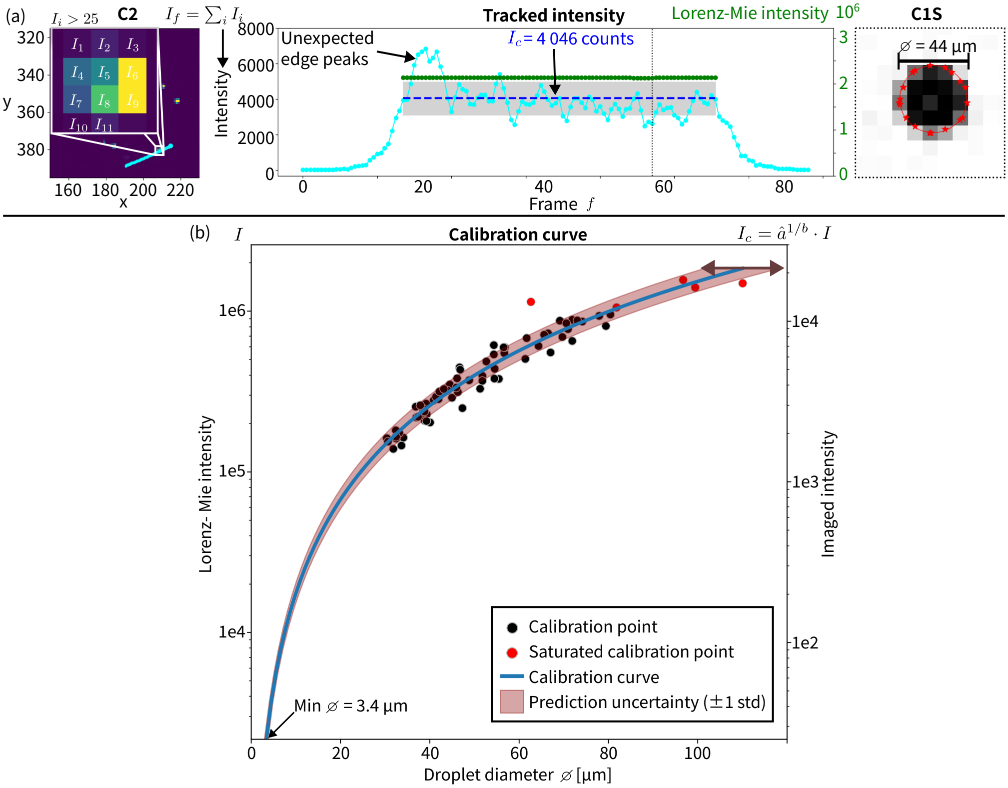


Figure 4.14: Experimental calibration results. An example calibration droplet is shown in (a) where the imaged intensity and size from the shadowgraphy setup is shown. The calibration curve is shown in (b) together with all calibration droplets that enables sizing of respiratory droplets. The red area in (b) corresponds to an estimated prediction uncertainty of the sizing.

a more reasonable limit is estimated to 4 µm. The corresponding maximum sizing limit before saturation is around 80 µm for moving droplets.

4.7.2 Correlation of respiratory droplet speed and size

This sizing technique can be combined with the 3D-PTV as is shown in Figure 4.15. Here, the speed and size are estimated simultaneously for the tracked droplets which are visualised in scatter plots for the same speaking, yelling and coughing respiratory event as was shown in Figure 4.7. The marginal distributions are also plotted for both the speed and size. The speaking and yelling show a tendency of an "L" shape where the larger droplets seem to generally have a slower speed and the smaller droplets have a larger variation of speeds. The coughed droplets do not show this same pattern where also larger droplets have a wide speed distribution. The marginal speed distributions are here the similar to what has been previously shown. There is a slight difference since only tracked particles in the centre ($5 < Z < 10$ mm) of the illumination volume are sized to avoid erroneous tracks at the illumination edges. The marginal size distribution show relatively larger sizes for yelling compared to speaking and the coughing case is in between the two. In the literature, the speech respiratory events with different loudness has in one study been found to have approximately the same size distribution measured using the APS [50]. However, another study using OPS show a higher concentration of droplets sizes below around 40 µm with higher loudness of speech [53]. To make this sizing technique useful for understanding how the size distributions actually behave for different respiratory event types, it is required to quantify performance of this sizing technique mainly by estimating its counting efficiency. The counting efficiency is largely connected to the tracking performance where the suggested validation for the PTV approach will be helpful. Otherwise, experimental validation by use of particles with known sizes such as those created by droplet generators or manufactured calibration particles are suggested. In addition, the large variation in number of droplets and sizes between respiratory events as seen in Paper v and in the literature require a large number of recorded events of multiple subjects to gain significant results.

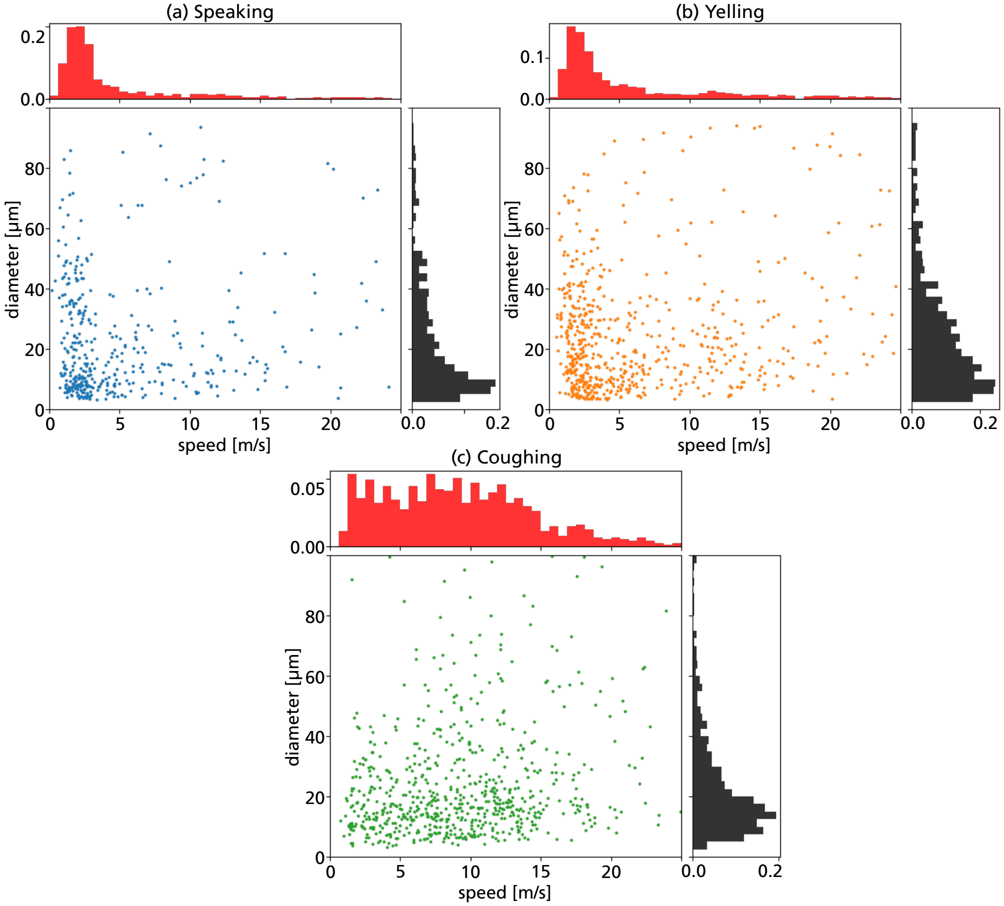


Figure 4.15: Results of how droplet speed and size correlate for one event of speaking (a), yelling (b) and coughing (c). The chosen events are the same as the ones found in Figure 4.7.

Chapter 5

Summary and outlook

This thesis has presented three measurement techniques for instantaneous high-speed 3D imaging of three respective liquid phenomena. For each technique, analysis has been performed to evaluate the accuracy of the 3D results. Each technique is summarized below.

High-speed volumetric imaging of liquid jets

- LIF intensity is used to estimate the liquid thickness of a water jet injected at four different injection pressures and at different displacements from the nozzle.
- Refraction effects through the deformed liquid-air interface create imaged artefacts of unproportionally large LIF intensities. The artefacts are caused by refraction effects which include local ejection of fluorescence light trapped by internal total reflection.
- An algorithm for finding and correcting the artefacts called *corrupt LIF peaks* have been implemented. The finding algorithm is an outlier detection scheme similar to a Bayesian classifier. Here, a pixel with high enough LIF intensity (conditioned on two key features) to be sufficiently unlikely is classified as being corrupt. The correction is an interpolation where only pixel values not found to be corrupt is used. The correction is found to improve the calibration curve for estimating liquid thickness from LIF intensity.
- The data produced from the experiments on the water jet can be viewed and downloaded at `https://spray-imaging.com/water-jet.html` or `https://doi.org/10.17605/OSF.IO/CG3DF`.

High-speed 3D reconstruction of liquid surfaces

- A technique called Fringe Projection - Laser Induced Fluorescence (FP-LIF) is developed as a combination of fringe projection 3D reconstruction and LIF. The LIF is used to limit detection of light to the liquid surface. FP-LIF is here experimentally applied to a hollow cone liquid sheet recorded at 20 000 frames per second.
- The 3D reconstructed liquid surface enables the extraction of multiple spray-angles for each recorded frame. They can reveal asymmetries in for example propagation of surface waves that is of relevance for the characterization of the injection nozzle.
- Reconstruction of simulated FP-LIF images with realistic noise show a general RMSE around 0.001%, 0.000% and 0.004% for the estimated X, Y and Z coordinates, respectively.
- Estimated spray-angles from simulated data have an uncertainty below 0.2°.
- Simulated gaussian shaped waves on the surface of a conical structure are found to be resolved up to 65% of the theoretical maximum.
- Errors due to the assumptions of orthographic imaging is found to create a bias in spray-angles by less than 0.9° for the experimental setup used.
- The python package `fp23dpy` that performs all post-processing for the FP-LIF technique is published as open-source software and is found at `https://gitlab.com/roth.adrian/fp23dpy`.
- The 3D data reconstructed for the liquid sheet hollow cone experiments can be interactively viewed at `https://3d.spray-imaging.com/cone4d/` and `https://3d.spray-imaging.com/cone4d_evolution/` (takes a minute to load) and downloaded at `https://doi.org/10.17605/OSF.IO/ZT97W`.

High-speed 3D velocimetry and sizing of respiratory droplets

- A 3D-PTV algorithm is developed and applied to respiratory droplets that enable estimation of both spatial and temporal velocity distributions.
- The sensitivity of triangulated 3D coordinates is estimated to 0.2 mm from a worst-case simulation and the position uncertainty from the tracking results is found to be 0.04 mm.
- The matching for 3D triangulation is found to be correct for 93% of the particles and create around 5% ghost particles. The droplet density used in this estimation corresponds to the toughest experimental case in this work that is the cough.

- A comparison between two surgical facial masks is performed using this detection technique. It suggests that the type IIR removes more droplets relative to the type I mask.

- The scattering intensity from the respiratory droplets can be used to estimate the droplet size. This is combined with the 3D-PTV algorithm to extract both the speed and size for the same droplets.

- The python package `ptv3py` implements the 3D-PTV algorithm described in this thesis. It is published as open-source software and can be found at `https://gitlab.com/roth.adrian/ptv3py`. The package also includes functions for interactive visualization of PTV results and tuning of post-processing input parameters.

Future considerations

- Attempt Volumetric LIF with more views together with a post-processing model that take refraction processes into account.

- Implement projective 3D reconstruction using the FP-LIF technique that can both correct for depth in projective cameras and the projective properties of the illuminated fringes.

- Perform simulations and experiments to quantify the performance of the full 3D-PTV algorithm.

- Quantify the counting efficiency of the technique for sizing respiratory droplets.

- Apply 3D reconstruction using the developed 3D imaging techniques on experiments where relevant parameters are varied and preferably in collaboration with modellers to enable comparison, validation, and development of theoretical models.

Appendix A

Gaussian Markov Random Field (GMRF) introduction

The GMRF is a useful tool for both interpolation of unknown/corrupt data and simulation of realistic images. This chapter will focus on the practical aspects of the GMRF where python code is provided below for the examples used in this chapter. For a detailed theoretical understanding, see [92]. GMRFs is in this thesis used for operations where spatial correlations in images are of importance. Images generally contain spatially correlated information which means that two pixels that are close to one another in the image have a higher probability of having similar pixel intensity and the probability drops with larger distance between the pixels. There are cases where pixels further apart have a higher correlation as well but it is less common. To perform this kind of processing is heavy when it comes to large images as will be shown below. The Markov property is used to simplify the problem and increase efficiency of the computations. Practically the GMRF in this work is used on images of size M x N that modelled as a matrix of pixel stochastical variables X with the same shape,

$$\mathbf{X} = \begin{bmatrix} x_{0,0} & x_{0,1} & \cdots & x_{0,N-1} & x_{0,N} \\ x_{1,0} & x_{1,1} & \cdots & x_{1,N-1} & x_{1,N} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ x_{M-1,0} & x_{M-1,1} & \cdots & x_{M-1,N-1} & x_{M-1,N} \\ x_{M,0} & x_{M,1} & \cdots & x_{M,N-1} & x_{M,N} \end{bmatrix}. \tag{A.1}$$

This image is then column stacked into $\mathbf{x}$ for simplification when using matrix algebra,

$$\mathbf{x} = \begin{bmatrix} x_{0,0} \\ x_{1,0} \\ \vdots \\ x_{M-1,0} \\ x_{M,0} \\ x_{0,1} \\ x_{1,1} \\ \vdots \\ x_{M-1,N} \\ x_{M,N} \end{bmatrix}. \tag{A.2}$$

With this notation we define $\mathbf{x}$ to be a Gaussian Random Field where,

$$\mathbf{x} \in \mathrm{N}(\mu, \boldsymbol{\Sigma}). \tag{A.3}$$

Here, μ is an average value of the field that can be varying for different positions in the image.

$$\mu = \begin{bmatrix} \mu_{0,0} \\ \mu_{1,0} \\ \vdots \\ \mu_{M-1,0} \\ \mu_{M,0} \\ \mu_{0,1} \\ \mu_{1,1} \\ \vdots \\ \mu_{M-1,N} \\ \mu_{M,N} \end{bmatrix}. \tag{A.4}$$

In this work it is either set to a single average value or 0 for all pixels. $\boldsymbol{\Sigma}$ is the covariance matrix of this field. It holds all the information connected to spatial correlations,

$$\boldsymbol{\Sigma} = \begin{bmatrix} V(x_{0,0}) & \mathrm{C}(x_{0,0}, x_{0,1}) & \cdots & \mathrm{C}(x_{0,0}, x_{M-1,N}) & \mathrm{C}(x_{0,0}, x_{M,N}) \\ \mathrm{C}(x_{1,0}, x_{0,0}) & V(x_{1,0}) & \cdots & \mathrm{C}(x_{1,0}, x_{M-1,N}) & \mathrm{C}(x_{1,0}, x_{M,N}) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{C}(x_{M-1,N}, x_{0,0}) & \mathrm{C}(x_{N-1,M}, x_{1,0}) & \cdots & V(x_{N-1,M}) & \mathrm{C}(x_{M-1,N}, x_{M,N}) \\ \mathrm{C}(x_{M,N}, x_{0,0}) & \mathrm{C}(x_{N,M}, x_{1,0}) & \cdots & \mathrm{C}(x_{M,N}, x_{N-1,M}) & V(x_{M,N}) \end{bmatrix}. \tag{A.5}$$

Here, V is the variance and C is the covariance of the corresponding two pixels. This covariance function can be defined differently depending on the images that are modelled, but it is limited to forms that create a symmetric and positive semi-definite covariance matrix. One challenge with the covariance matrix is its size ($(M \cdot N)^2$). For a typical 1 MegaPixel image one would require 10^{12} numbers or for double floats more than 7 TerraByte of memory which is typically not available and if it would be, calculating anything would take tooo long. To be able to work with this, the inverse of the covariance matrix, called the precision matrix, $\mathbf{Q} = \mathbf{\Sigma}^{-1}$ is considered. For specific choices of the covariance function C this matrix becomes sparse non-zero. Then it is only required to store the non-zero values and their positions that will reduce both memory usage and computation time. The non-zero elements of $\mathbf{Q}$ will be the values on the diagonal and the rows and columns connecting pixels that are neighbours in the original image. The neighbours are defined through the so-called local neighbourhood matrix g that is here a $3x3$ matrix,

$$\mathbf{g} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \tag{A.6}$$

The local neighbourhood matrix is used to create the large neighbourhood matrix $\mathbf{G}$ by placing g over a pixel with index i, j in the field $\mathbf{x}$. The values for index i', j' of surrounding pixels at this location with non-zero value in g is inserted at the corresponding position in the G matrix. This can also be defined as,

$$\mathbf{G}[i,j] = \begin{cases} 4 & \text{if } i = i' \text{and } j = j' \\ -1 & \text{if } i', j' \in \text{neighbourhood of } i, j \\ 0 & \text{otherwise} \end{cases}. \tag{A.7}$$

Below is an example $\mathbf{G}$ for a field $\mathbf{x}$ with shape $3x3$,

$$\mathbf{G} = \begin{bmatrix} 4 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ -1 & 4 & -1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 4 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 4 & -1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 & 4 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & -1 & 4 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 4 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & -1 & 4 & -1 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 4 \end{bmatrix}. \tag{A.8}$$

Finally,

$$\mathbf{Q} = \tau(\kappa^2 \mathbf{I} + \mathbf{G}). \tag{A.9}$$

τ is the inverse of squared correlation σ^2. Generally, a larger value will give smaller values of the sampled field and larger variance. κ^2 must be strictly larger than zero and a large value will result in less spatial correlation between pixels. I is the identity matrix with the same shape as G. Note that there are different ways of defining the neighbourhood matrix G to get different correlation patterns.

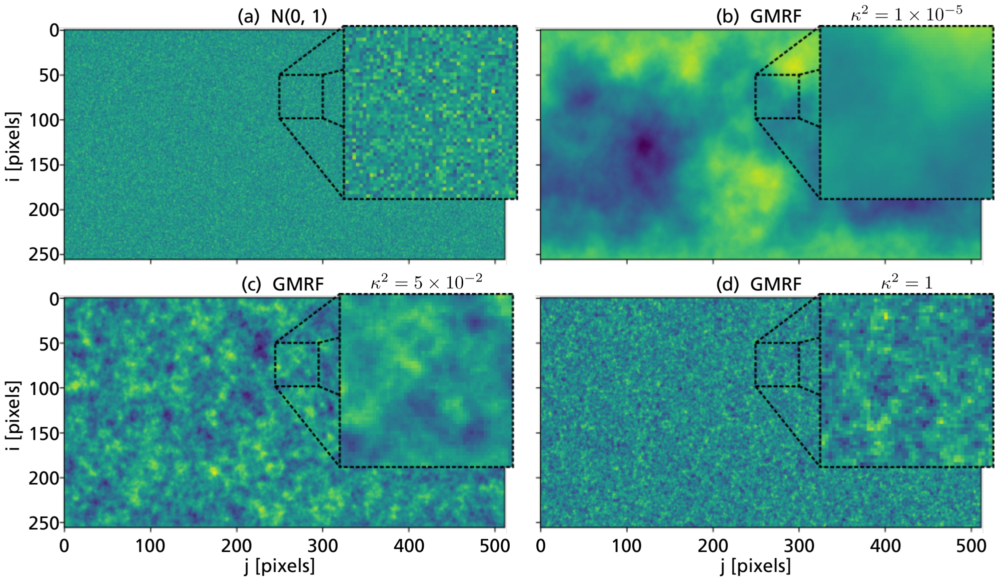


Figure A.1: Showing the difference between a normally distributed random sample (a) and GMRF samples using different κ^2 (b), (c) and (d). Larger κ^2 gives less spatial correlation. τ is set to 1 for all samples.

A.1 Simulate a spatially correlated sample

With a μ and a $\mathbf{Q}$ a sample $\mathbf{x}^*$ of our stochastical field x can be calculated as,

$$\mathbf{x}^* = \mathbf{Q}^{-1}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \in N(0, \mathbf{I}), \tag{A.10}$$

where $\boldsymbol{\epsilon}$ is independent normally distributed values with standard deviation 1. Examples of such samples are shown in Figure A.1.

A.2 Interpolating missing/corrupt pixel values

Noe let us say that our image $\mathbf{x}$ has some pixels that are corrupt/missing. An observation vector a is defined to say which pixels are known.

$$a = \begin{cases} True, & \text{pixel is known,} \\ False, & \text{pixel is unknown (missing/corrupt)} \end{cases} . \tag{A.11}$$

The observation vector is used to extract the pixels that are observed,

$$x_k = x[a]. \tag{A.12}$$

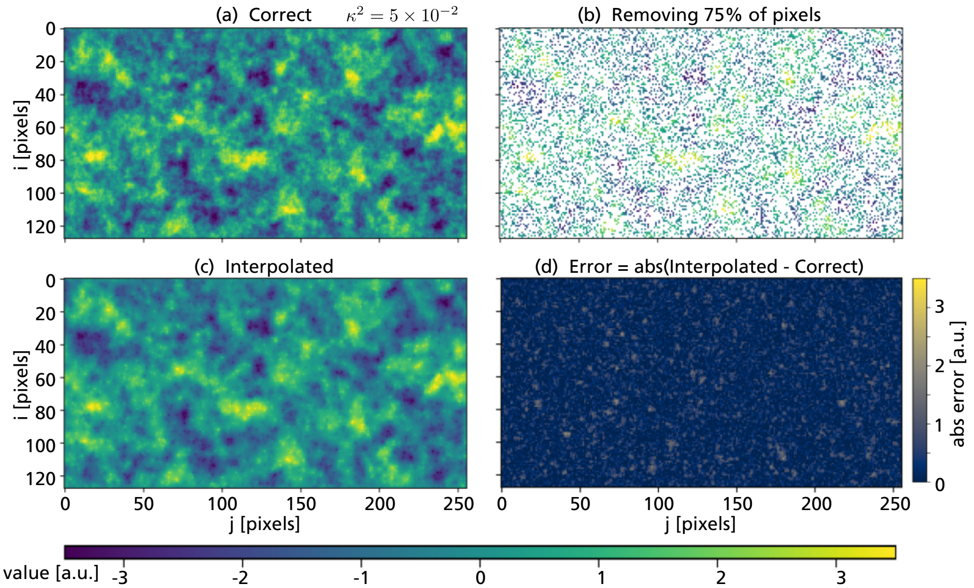


Figure A.2: Example interpolation of unknown pixels using the GMRF approach. A GMRF sample is calculated (a) and 75% of the pixels are removed randomly. With the remaining pixels and known spatial correlation, the unknown pixels are interpolated (c). One can see that some of the small-scale variations are lost in the interpolation that creates the errors shown in (d).

We then know the τ and κ^2 that best describe this data so a precision matrix $\mathbf{Q}$ is created from that. Then two different subparts of the precision matrix are extracted.

$$\begin{aligned} \mathbf{Q}_{uu} &= \mathbf{Q}[\neg a, \neg a], \\ \mathbf{Q}_{uk} &= \mathbf{Q}[\neg a, a]. \end{aligned} \tag{A.13}$$

$\neg$ is here used to negate the a vector so that it instead extracts values that are unknown. The unknown pixel values are then interpolated using the following equation,

$$\mathbf{x}[\neg a] = \mu_u - \mathbf{Q}_{uu}^{-1}\mathbf{Q}_{uk}(\mathbf{x}_k - \mu_k). \tag{A.14}$$

μ shoud be a known background of the field that can either be constant or varying. These interpolated values are a maximum likelihood estimation given the assumption that the data follows the GMRF model with the used τ and κ^2. An example interpolation of a GMRF sample is shown in Figure A.2.

A.3 Example python code

```
"""
Example script for Gaussian Markov Random Field (GMRF) sampling and
interpolation of corrupt pixels
"""
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.cm as mcm
from scipy import sparse

np.random.seed(0)


def get_neighbourhood(shape):
    # Shape of local neighbourhood g must be odd in both dimensions
    g = np.array(((0, -1, 0), (-1, 4, -1), (0, -1, 0)))

    mid_r, mid_c = (np.array(g.shape) - 1) // 2
    n_diags = np.count_nonzero(g)
    r, c = np.nonzero(g)

    size = np.prod(shape)
    data = np.ones((n_diags, size)) * np.expand_dims(g[g != 0], 1)
    diags = (r - mid_r) * shape[1] + (c - mid_c)

    G = sparse.spdiags(data, diags, size, size).tocsc()
    return G


def get_precision_matrix(shape, tau, kappa2):
    G = get_neighbourhood(shape)
    Q = tau * (kappa2 * sparse.eye(G.shape[0]).tocsc() + G)
    return Q


def sample_gmrf_field(shape, tau, kappa2):
    Q = get_precision_matrix(shape, tau, kappa2)
    invQ = sparse.linalg.splu(Q)
    x = invQ.solve(np.random.randn(Q.shape[0]))
    x.shape = shape
    return x


def gmrf_interpolate_unknown_pixels(sample_unknown, unknown_inds, mu, tau,
    kappa2):
    known_inds = ~unknown_inds.ravel()
    sample_interpolated = sample_unknown.copy()
    sample_known = sample_interpolated.ravel()[known_inds]

    Q = get_precision_matrix(sample_unknown.shape, tau, kappa2)
    Quu = Q[~known_inds, :][:, ~known_inds]
    Quk = Q[~known_inds, :][:, known_inds]
    sample_interpolated[unknown_inds] = mu - sparse.linalg.spsolve(
        Quu, Quk.dot(sample_known - mu)
    )
    return sample_interpolated
```

```
def plot_sample_fields():
    shape = (256, 512)
    tau = 1

    # normally distributed noise sample
    e = np.random.randn(*shape)
    plt.figure()
    plt.imshow(e)
    plt.savefig("gmrf_sample_e", dpi=300)
    plt.xlim([250, 300])
    plt.ylim([100, 50])
    plt.savefig("gmrf_sample_zoom_e", dpi=300)
    plt.close()

    # gmrf samples
    for i, kappa2 in enumerate([1e-5, 5e-2, 1]):
        x = sample_gmrf_field(shape, tau, kappa2)
        plt.figure()
        plt.imshow(x)
        plt.savefig(f"gmrf_sample_{i}", dpi=300)
        plt.xlim([250, 300])
        plt.ylim([100, 50])
        plt.savefig(f"gmrf_sample_zoom_{i}", dpi=300)
        plt.close()


def plot_interpolation_unknown_pixels():
    vlim = 3.5
    shape = (128, 256)
    tau = 1
    kappa2 = 5e-2

    fraction_unknown = 0.75

    mu = 0
    sample_correct = mu + sample_gmrf_field(shape, tau, kappa2)

    unknown_inds = np.random.random(shape) < fraction_unknown
    sample_unknown = sample_correct.copy()
    sample_unknown[unknown_inds] = np.nan

    sample_interpolated = gmrf_interpolate_unknown_pixels(
        sample_unknown, unknown_inds, mu, tau, kappa2
    )

    for label, sample in [
        ["correct", sample_correct],
        ["unknown", sample_unknown],
        ["interpolated", sample_interpolated],
    ]:
        plt.figure()
        plt.imshow(sample, vmin=-vlim, vmax=vlim)
        plt.savefig(f"gmrf_interpolation_{label}", dpi=300)
        plt.close()

    mappable = mcm.ScalarMappable()
```

```
    mappable.set_clim(-vlim, vlim)
    fig, ax = plt.subplots(
        figsize=np.array((5, 0.4)) * 1.5,
        gridspec_kw=dict(left=0.01, right=0.99, bottom=0.43, top=0.55),
    )
    fig.colorbar(mappable, ax, orientation="horizontal")
    plt.savefig("gmrf_interpolation_colorbar", dpi=300)
    plt.close()

    sample_interpolated_error = np.abs(sample_interpolated - sample_correct)
    plt.figure()
    plt.imshow(sample_interpolated_error, cmap="cividis", vmin=0, vmax=vlim)
    plt.colorbar()
    plt.savefig("gmrf_interpolation_error", dpi=300)
    plt.close()


if __name__ == "__main__":
    plot_sample_fields()
    plot_interpolation_unknown_pixels()
```

References

[1] T. D. Fansler and S. E. Parrish. Spray measurement technology: a review. *Measurement Science and Technology*, 26(1):012002, 2015. URL `https://doi.org/10.1088/0957-0233/26/1/012002`.

[2] E. Hilz and A. W. P. Vermeer. Spray drift review: The extent to which a formulation can contribute to spray drift reduction. *Crop Protection*, 44:75–83, 2013. URL `https://doi.org/10.1016/j.cropro.2012.10.020`.

[3] C. C. Wang, K. A. Prather, J. Sznitman, J. L. Jimenez, S. S. Lakdawala, Z. Tufekci, and L. C. Marr. Airborne transmission of respiratory viruses. *Science*, 373(6558): eabd9149, 2021. URL `https://doi.org/10.1126/science.abd9149`.

[4] E. Goldman. Exaggerated risk of transmission of COVID-19 by fomites. *The Lancet Infectious Diseases*, 20(8):892–893, 2020. URL `https://doi.org/10.1016/S1473-3099(20)30561-2`.

[5] D. Lewis. COVID-19 rarely spreads through surfaces. So why are we still deep cleaning? *Nature*, 590(7844):26–28, 2021. URL `https://doi.org/10.1038/d41586-021-00251-4`.

[6] H. Grosshans, E. Kristensson, R.-Z. Szász, and E. Berrocal. Prediction and Measurement of the local extinction coefficient in sprays for 3D simulation/experiment data comparison. *International Journal of Multiphase Flow*, 72:218–232, 2015. URL `https://doi.org/10.1016/j.ijmultiphaseflow.2015.01.009`.

[7] W. D. Bachalo. Method for measuring the size and velocity of spheres by dual-beam light-scatter interferometry. *Applied Optics*, 19(3):363–370, 1980. URL `https://doi.org/10.1364/AO.19.000363`.

[8] J. B. Ghandhi and D. M. Heim. An optimized optical system for backlit imaging. *Review of Scientific Instruments*, 80(5):056105, 2009. URL `https://doi.org/10.1063/1.3128728`.

[9] G. S. Settles. *Schlieren and Shadowgraph Techniques: Visualizing Phenomena in Transparent Media*. Experimental Fluid Mechanics. Springer Berlin Heidelberg, 2012. ISBN 978-3-642-56640-0. URL `https://doi.org/10.1007/978-3-642-56640-0`.

[10] M. Raffel, C.E. Willert, F. Scarano, C.J. Kähler, S.T. Wereley, and J. Kompenhans. *Particle Image Velocimetry: A Practical Guide*. Experimental Fluid Mechanics. Springer International Publishing, 3rd edition, 2018. ISBN 978-3-319-68852-7. URL `https://doi.org/10.1007/978-3-319-68852-7`.

[11] S. Moller, T. Käufer, A. Pandey, J. Schumacher, and C. Cierpka. Combined particle image velocimetry and thermometry of turbulent superstructures in thermal convection. *Journal of Fluid Mechanics*, 945:A22, 2022. URL `https://doi.org/10.1017/jfm.2022.538`.

[12] D. Dabiri and C. Pecora. *Particle Tracking Velocimetry*. IOP Publishing, 2019. ISBN 978-0-7503-2203-4. URL `https://doi.org/10.1088/978-0-7503-2203-4`.

[13] S. V. Sankar, K. E. Maher, D. M. Robart, and W. D. Bachalo. Rapid Characterization of Fuel Atomizers Using an Optical Patternator. *Journal of Engineering for Gas Turbines and Power*, 121(3):409–414, 1999. URL `https://doi.org/10.1115/1.2818488`.

[14] E. Berrocal, E. Kristensson, M. Richter, M. Linne, and M. Aldén. Application of structured illumination for multiple scattering suppression in planar laser imaging of dense sprays. *Optics Express*, 16(22):17870–17881, 2008. URL `https://doi.org/10.1364/OE.16.017870`.

[15] E. Kristensson, E. Berrocal, M. Richter, S.-G. Pettersson, and M. Aldén. High-speed structured planar laser illumination for contrast improvement of two-phase flow images. *Optics Letters*, 33(23):2752–2754, 2008. URL `https://doi.org/10.1364/OL.33.002752`.

[16] E. Kristensson, E. Berrocal, and M. Aldén. Quantitative 3D imaging of scattering media using structured illumination and computed tomography. *Optics Express*, 20 (13):14437, 2012. URL `https://doi.org/10.1364/OE.20.014437`.

[17] D. Guénot, K. Svendsen, B. Lehnert, H. Ulrich, A. Persson, A. Permogorov, L. Zigan, M. Wensing, O. Lundh, and E. Berrocal. Distribution of Liquid Mass in Transient Sprays Measured Using Laser-Plasma-Driven X-Ray Tomography. *Physical Review Applied*, 17(6):064056, 2022. URL `https://doi.org/10.1103/PhysRevApplied.17.064056`.

[18] R. Wellander, E. Berrocal, E. Kristensson, M. Richter, and M. Aldén. Three-dimensional measurement of the local extinction coefficient in a dense spray. *Measurement Science and Technology*, 22(12):125303, 2011. URL `https://doi.org/10.1088/0957-0233/22/12/125303`.

[19] B. R. Halls, N. Rahman, M. N. Slipchenko, J. W. James, A. McMaster, M. D. A. Ligthfoot, J. R. Gord, and T. R. Meyer. 4D spatiotemporal evolution of liquid spray using kilohertz-rate x-ray computed tomography. *Optics Letters*, 44(20):5013, 2019. URL `https://doi.org/10.1364/OL.44.005013`.

[20] E. M. Hall, D. R. Guildenbecher, and B. S. Thurow. Uncertainty characterization of particle location from refocused plenoptic images. *Optics Express*, 25(18):21801, 2017. URL `https://doi.org/10.1364/OE.25.021801`.

[21] Z. P. Tan, K. Johnson, C. Clifford, and B. S. Thurow. Development of a modular, high-speed plenoptic-camera for 3D flow-measurement. *Optics Express*, 27(9):13400, 2019. URL `https://doi.org/10.1364/OE.27.013400`.

[22] Y. Wu, L. Wang, W. Lin, G. Song, Y. He, X. Wu, Z. Wang, and K. Cen. Picosecond pulsed digital off-axis holography for near-nozzle droplet size and 3D distribution measurement of a swirl kerosene spray. *Fuel*, 283:119124, 2021. URL `https://doi.org/10.1016/j.fuel.2020.119124`.

[23] Z. Wang, F. He, H. Zhang, P. Hao, X. Zhang, and X. Li. Three-dimensional measurement of the droplets out of focus in shadowgraphy systems via deep learning-based image-processing method. *Physics of Fluids*, 34(7):073301, 2022. URL `https://doi.org/10.1063/5.0097375`.

[24] D. Kiefhaber, S. Reith, R. Rocholz, and B. Jähne. High-speed imaging of short wind waves by shape from refraction. *Journal of the European Optical Society - Rapid publications*, 9(0), 2014. URL `https://doi.org/10.2971/jeos.2014.14015`.

[25] M. Kim, D. Schanz, M. Novara, P. Godbersen, E. Yeom, and A. Schröder. Experimental study on flow and turbulence characteristics of jet impinging on cylinder using three-dimensional Lagrangian particle tracking velocimetry. *Scientific Reports*, 13(1): 10929, 2023. URL `https://doi.org/10.1038/s41598-023-37970-9`.

[26] D. Schanz, S. Gesemann, and A. Schröder. Shake-The-Box: Lagrangian particle tracking at high particle image densities. *Experiments in Fluids*, 57(5):70, 2016. URL `https://doi.org/10.1007/s00348-016-2157-1`.

[27] J. K. Gupta, C.-H. Lin, and Q. Chen. Flow dynamics and characterization of a cough: Flow dynamics and characterization of a cough. *Indoor Air*, 19(6):517–525, 2009. URL `https://doi.org/10.1111/j.1600-0668.2009.00619.x`.

[28] J. W. Tang, T. J. Liebner, B. A. Craven, and G. S. Settles. A schlieren optical study of the human cough with and without wearing masks for aerosol infection control. *Journal of The Royal Society Interface*, 6(suppl_6), 2009. URL `https://doi.org/10.1098/rsif.2009.0295.focus`.

[29] J. W. Tang, A. Nicolle, J. Pantelic, G. C. Koh, L. De Wang, M. Amin, C. A. Klettner, D. K. W. Cheong, C. Sekhar, and K. Wai Tham. Airflow Dynamics of Coughing in Healthy Human Volunteers by Shadowgraph Imaging: An Aid to Aerosol Infection Control. *PLoS ONE*, 7(4):e34818, 2012. URL `https://doi.org/10.1371/journal.pone.0034818`.

[30] H. Alsaad, G. Schälte, M. Schneeweiß, L. Becher, M. Pollack, A.W. Gena, M. Schweiker, M. Hartmann, C. Voelker, R. Rossaint, and M. Irrgang. The Spread of Exhaled Air and Aerosols during Physical Exercise. *Journal of Clinical Medicine*, 12 (4), 2023. URL `https://doi.org/10.3390/jcm12041300`.

[31] M. Han, R. Ooka, H. Kikumoto, W. Oh, Y. Bu, and S. Hu. Experimental measurements of airflow features and velocity distribution exhaled from sneeze and speech using particle image velocimetry. *Building and Environment*, 205:108293, 2021. URL `https://doi.org/10.1016/j.buildenv.2021.108293`.

[32] C.Y.H. Chao, M.P. Wan, L. Morawska, G.R. Johnson, Z.D. Ristovski, M. Hargreaves, K. Mengersen, S. Corbett, Y. Li, X. Xie, and D. Katoshevski. Characterization of expiration air jets and droplet size distributions immediately at the mouth opening. *Journal of Aerosol Science*, 40(2):122–133, 2009. URL `https://doi.org/10.1016/j.jaerosci.2008.10.003`.

[33] N. Dudalski, A. Mohamed, S. Mubareka, R. Bi, C. Zhang, and E. Savory. Experimental investigation of far-field human cough airflows from healthy and influenza-infected subjects. *Indoor Air*, 30(5):966–977, 2020. URL `https://doi.org/10.1111/ina.12680`.

[34] S.-B. Kwon, J. Park, J. Jang, Y. Cho, D.-S. Park, C. Kim, G.-N. Bae, and A. Jang. Study on the initial velocity distribution of exhaled air from coughing and speaking. *Chemosphere*, 87(11):1260–1264, 2012. URL `https://doi.org/10.1016/j.chemosphere.2012.01.032`.

[35] E. Savory, W. E Lin, K. Blackman, M. C Roberto, L. R Cuthbertson, J. A Scott, and S. Mubareka. Western Cold and Flu (WeCoF) aerosol study – preliminary results. *BMC Research Notes*, 7(1):563, 2014. URL `https://doi.org/10.1186/1756-0500-7-563`.

[36] M. VanSciver, S. Miller, and J. Hertzberg. Particle Image Velocimetry of Human Cough. *Aerosol Science and Technology*, 45(3):415–422, 2011. URL `https://doi.org/10.1080/02786826.2010.542785`.

[37] S. Zhu, S. Kato, and J.-H. Yang. Study on transport characteristics of saliva droplets produced by coughing in a calm indoor environment. *Building and Environment*, page 12, 2006. URL `https://doi.org/10.1016/j.buildenv.2005.06.024`.

[38] H. Wang, Z. Li, X. Zhang, L. Zhu, Y. Liu, and S. Wang. The motion of respiratory droplets produced by coughing. *Physics of Fluids*, 32(12):125102, 2020. URL `https://doi.org/10.1063/5.0033849`.

[39] Y. Peng and M. Yao. Quantitatively Visualizing Airborne Disease Transmission Risks of Different Exhalation Activities through CO2 Imaging. *Environmental Science and Technology*, 57(17):6865–6875, 2023. URL `https://doi.org/10.1021/acs.est.2c08503`.

[40] H. Nishimura, S. Sakata, and A. Kaga. A New Methodology for Studying Dynamics of Aerosol Particles in Sneeze and Cough Using a Digital High-Vision, High-Speed Video System and Vector Analyses. *PLOS ONE*, 8(11):e80244, 2013. URL `https://doi.org/10.1371/journal.pone.0080244`.

[41] T. Ahmed, H.E. Wendling, A.A. Mofakham, G. Ahmadi, B.T. Helenbrook, A.R. Ferro, D.M. Brown, and B.D. Erath. Variability in expiratory trajectory angles during consonant production by one human subject and from a physical mouth model: Application to respiratory droplet emission. *Indoor Air*, 31(6):1896–1912, 2021. URL `https://doi.org/10.1111/ina.12908`.

[42] S. Kniesburges, P. Schlegel, G. Peters, C. Westphalen, B. Jakubaß, R. Veltrup, A.M. Kist, M. Döllinger, S. Gantner, L. Kuranova, T. Benthaus, M. Semmler, and M. Echternach. Effects of surgical masks on aerosol dispersion in professional singing. *Journal of Exposure Science and Environmental Epidemiology*, 32(5):727–734, 2022. URL `https://doi.org/10.1038/s41370-021-00385-7`.

[43] L. Bourouiba, E. Dehandschoewercker, and J. W. M. Bush. Violent expiratory events: on coughing and sneezing. *Journal of Fluid Mechanics*, 745:537–563, 2014. URL `https://doi.org/10.1017/jfm.2014.88`.

[44] P. Anfinrud, V. Stadnytskyi, C. E. Bax, and A. Bax. Visualizing Speech-Generated Oral Fluid Droplets with Laser Light Scattering. *New England Journal of Medicine*, 382(21):2061–2063, 2020. URL `https://doi.org/10.1056/NEJMc2007800`.

[45] V. Stadnytskyi, C. E. Bax, A. Bax, and P. Anfinrud. The airborne lifetime of small speech droplets and their potential importance in SARS-CoV-2 transmission. *Proceedings of the National Academy of Sciences*, 117(22):11875–11877, 2020. URL `https://doi.org/10.1073/pnas.2006874117`.

[46] E. P. Fischer, M. C. Fischer, D. Grass, I. Henrion, W. S. Warren, and E. Westman. Low-cost measurement of face mask efficacy for filtering expelled droplets during speech. *Science Advances*, 6(36):eabd3083, 2020. URL `https://doi.org/10.1126/sciadv.abd3083`.

[47] M. Alsved, A. Matamis, R. Bohlin, M. Richter, P.-E. Bengtsson, C.-J. Fraenkel, P. Medstrand, and J. Löndahl. Exhaled respiratory particles during singing and talking. *Aerosol Science and Technology*, 54(11):1245–1248, 2020. URL `https://doi.org/10.1080/02786826.2020.1812502`.

[48] M. Alsved, D. Nygren, S. Thuresson, C.-J. Fraenkel, P. Medstrand, and J. Löndahl. Size distribution of exhaled aerosol particles containing SARS-CoV-2 RNA. *Infectious Diseases*, 55(2):158–163, 2023. URL `https://doi.org/10.1080/23744235.2022.2140822`.

[49] G. R. Johnson, L. Morawska, Z. D. Ristovski, M. Hargreaves, K. Mengersen, C. Y. H. Chao, M. P. Wan, Y. Li, X. Xie, D. Katoshevski, and S. Corbett. Modality of human expired aerosol size distributions. *Journal of Aerosol Science*, 42(12):839–851, 2011. URL `https://doi.org/10.1016/j.jaerosci.2011.07.009`.

[50] S. Asadi, A. S. Wexler, C. D. Cappa, S. Barreda, N. M. Bouvier, and W. D. Ristenpart. Aerosol emission and superemission during human speech increase with voice loudness. *Scientific Reports*, 9(1):2348, 2019. URL `https://doi.org/10.1038/s41598-019-38808-z`.

[51] L. Morawska, G.R. Johnson, Z.D. Ristovski, M. Hargreaves, K. Mengersen, S. Corbett, C.Y.H. Chao, Y. Li, and D. Katoshevski. Size distribution and sites of origin of droplets expelled from the human respiratory tract during expiratory activities. *Journal of Aerosol Science*, 40(3):256–269, 2009. URL `https://doi.org/10.1016/j.jaerosci.2008.11.002`.

[52] S. Yang, G. W.M. Lee, C.-M. Chen, C.-C. Wu, and K.-P. Yu. The Size and Concentration of Droplets Generated by Coughing in Human Subjects. *Journal of Aerosol Medicine*, 20(4):484–494, 2007. URL `https://doi.org/10.1089/jam.2007.0610`.

[53] G. Bagheri, O. Schlenczek, L. Turco, B. Thiede, K. Stieger, J. M. Kosub, S. Clauberg, M. L. Pöhlker, C. Pöhlker, J. Moláček, S. Scheithauer, and E. Bodenschatz. Size, concentration, and origin of human exhaled particles and their dependence on human factors with implications on infection transmission. *Journal of Aerosol Science*, 168:106102, 2023. URL `https://doi.org/10.1016/j.jaerosci.2022.106102`.

[54] J. Lee, D. Yoo, S. Ryu, S. Ham, K. Lee, M. Yeo, K. Min, and C. Yoon. Quantity, Size Distribution, and Characteristics of Cough-generated Aerosol Produced by Patients with an Upper Respiratory Tract Infection. *Aerosol and Air Quality Research*, 19(4):840–853, 2019. URL `https://doi.org/10.4209/aaqr.2018.01.0031`.

[55] X. Xie, Y. Li, H. Sun, and L. Liu. Exhaled droplets due to talking and coughing. *Journal of The Royal Society Interface*, 6(suppl_6):S703–S714, 2009. URL `https://doi.org/10.1098/rsif.2009.0388.focus`.

[56] R. S. Papineni and F. S. Rosenthal. The Size Distribution of Droplets in the Exhaled Breath of Healthy Human Subjects. *Journal of Aerosol Medicine*, 10(2):105–116, 1997. URL `https://doi.org/10.1089/jam.1997.10.105`.

[57] C. M. de Silva, P.. Bahl, C. Doolan, and C. Raina MacIntyre. Bespoke flow experiments to capture the dynamics of coughs and sneezes. *Measurement Science and Technology*, 32(12):125302, 2021. URL `https://doi.org/10.1088/1361-6501/ac2010`.

[58] S. Shao, D. Zhou, R. He, J. Li, S. Zou, K. Mallery, S. Kumar, S. Yang, and J. Hong. Risk assessment of airborne transmission of COVID-19 by asymptomatic individuals under different practical settings. *Journal of Aerosol Science*, 151:105661, 2021. URL `https://doi.org/10.1016/j.jaerosci.2020.105661`.

[59] A. Schröder, D. Schanz, J. Bosbach, M. Novara, R. Geisler, J. Agocs, and A. Kohl. Large-scale volumetric flow studies on transport of aerosol particles using a breathing human model with and without face protections. *Physics of Fluids*, 34(3):035133, 2022. URL `https://doi.org/10.1063/5.0086383`.

[60] S. P. Lin. *Breakup of Liquid Sheets and Jets*. Cambridge University Press, Cambridge, 2003. ISBN 978-0-521-80694-7. URL `https://doi.org/10.1017/CBO9780511547096`.

[61] A. E. S. E. T. Alajmi, N. M. Adam, A. A. Hairuddin, and L. C. Abdullah. Fuel atomization in gas turbines: A review of novel technology. *International Journal of Energy Research*, 43(8):3166–3181, 2019. URL `https://doi.org/10.1002/er.4415`.

[62] H. K. Suh and C. S. Lee. A review on atomization and exhaust emissions of a biodiesel-fueled compression ignition engine. *Renewable and Sustainable Energy Reviews*, 58: 1601–1620, 2016. URL `https://doi.org/10.1016/j.rser.2015.12.329`.

[63] S. J. Grauer, K. Mohri, T. Yu, H. Liu, and W. Cai. Volumetric emission tomography for combustion processes. *Progress in Energy and Combustion Science*, 94:101024, 2023. URL `https://doi.org/10.1016/j.pecs.2022.101024`.

[64] Y. Wu, W. Xu, and L. Ma. Kilohertz VLIF (volumetric laser induced fluorescence) measurements in a seeded free gas-phase jet in the transitionally turbulent flow regime. *Optics and Lasers in Engineering*, 102:52–58, 2018. URL `https://doi.org/10.1016/j.optlaseng.2017.10.004`.

[65] I. Ihrke, B. Goidluecke, and M. Magnor. Reconstructing the geometry of flowing water. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*, pages 1055–1060 Vol. 2, Beijing, China, 2005. IEEE. ISBN 978-0-7695-2334-7. URL `https://doi.org/10.1109/ICCV.2005.202`.

[66] M. Gomez, B. W. Yant, M. N. Slipchenko, A. M. Braun, Z. D. Rancilio, T. R. Meyer, and S. Roy. Four-dimensional laser-induced fluorescence and tomography of liquids. *International Journal of Multiphase Flow*, 166:104501, 2023. URL `https://doi.org/10.1016/j.ijmultiphaseflow.2023.104501`.

[67] N. Ashgriz. *Handbook of Atomization and Sprays. Theory and Applications.* Springer US, 2011. ISBN 978-1-4419-7263-7. URL `https://doi.org/10.1007/978-1-4419-7264-4`.

[68] S. Zhang. *High-speed 3D imaging with digital fringe projection techniques*. Optical sciences and applications of light. CRC Press, 2016. ISBN 978-1-4822-3433-6. URL `https://doi.org/10.1201/b19565`.

[69] J. Geng. Structured-light 3D surface imaging: a tutorial. *Advances in Optics and Photonics*, 3(2):128, 2011. URL `https://doi.org/10.1364/AOP.3.000128`.

[70] X. Su and Q. Zhang. Dynamic 3-D shape measurement method: A review. *Optics and Lasers in Engineering*, 48(2):191–204, 2010. URL `https://doi.org/10.1016/j.optlaseng.2009.03.012`.

[71] M. Takeda and K. Mutoh. Fourier transform profilometry for the automatic measurement of 3-D object shapes. *Applied Optics*, 22(24):3977, 1983. URL `https://doi.org/10.1364/AO.22.003977`.

[72] X. Su and W. Chen. Fourier transform profilometry: a review. *Optics and Lasers in Engineering*, page 22, 2001. URL `https://doi.org/10.1016/S0143-8166(01)00023-9`.

[73] L. R. Watkins. Continuous Wavelet Transforms. In Erwin Hack and Pramod Rastogi, editors, *Phase estimation in optical interferometry*. CRC Press, Boca Raton, 2015. ISBN 978-0-429-16524-5. URL `https://doi.org/10.1201/b17701`.

[74] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, and T. Yu. scikit-image: Image processing in python. *CoRR*, abs/1407.6245, 2014. URL `http://arxiv.org/abs/1407.6245`.

[75] M. A. Herráez, D. R. Burton, M. J. Lalor, and M. A. Gdeisat. Fast two-dimensional phase-unwrapping algorithm based on sorting by reliability following a noncontinuous path. *Applied Optics*, 41(35):7437–7444 – 7444, 2002. URL `https://doi.org/10.1364/AO.41.007437`.

[76] R. Tsai. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal on Robotics and Automation*, 3(4):323–344, 1987. URL `https://doi.org/10.1109/JRA.1987.1087109`.

[77] F. Pourfattah, L.-P. Wang, W. Deng, Y.-F. Ma, L. Hu, and B. Yang. Challenges in simulating and modeling the airborne virus transmission: A state-of-the-art review. *Physics of Fluids*, 33(10), 2021. URL `https://doi.org/10.1063/5.0061469`.

[78] K. Mahjoub Mohammed Merghani, B. Sagot, E. Gehin, G. Da, and C. Motzkus. A review on the applied techniques of exhaled airflow and droplets characterization. *Indoor Air*, 31(1):7–25, 2021. URL `https://doi.org/10.1111/ina.12770`.

[79] H. G. Maas, A. Gruen, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(2):133–146, 1993. URL `https://doi.org/10.1007/BF00190953`.

[80] N. Machicoane, A. Aliseda, R. Volk, and M. Bourgoin. A simplified and versatile calibration method for multi-camera optical systems in 3D particle imaging. *Review of Scientific Instruments*, 90(3):035112, 2019. URL `https://doi.org/10.1063/1.5080743`.

[81] K. Peterson, B. Regaard, S. Heinemann, and V. Sick. Single-camera, three-dimensional particle tracking velocimetry. *Optics Express*, 20(8):9031–9037, 2012. URL `https://doi.org/10.1364/OE.20.009031`.

[82] N. T. Ouellette, H. Xu, and E. Bodenschatz. A quantitative study of three-dimensional Lagrangian particle tracking algorithms. *Experiments in Fluids*, 40(2): 301–313, 2006. URL `https://doi.org/10.1007/s00348-005-0068-7`.

[83] A. Clark, N. Machicoane, and A. Aliseda. A quantitative study of track initialization of the four-frame best estimate algorithm for three-dimensional Lagrangian particle tracking. *Measurement Science and Technology*, 30(4):045302, 2019. URL `https://doi.org/10.1088/1361-6501/ab0786`.

[84] 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*, 29(2):145–153, 2000. URL `https://doi.org/10.1007/s003489900085`.

[85] F. Carosone, A. Cenedese, and G. Querzoli. Recognition of partially overlapped particle images using the Kohonen neural network. *Experiments in Fluids*, 19(4):225–232, 1995. URL `https://doi.org/10.1007/BF00196470`.

[86] K. Takehara and T. Etoh. A study on particle identification in PTV particle mask correlation method. *Journal of Visualization*, 1(3):313–323, 1999. URL `https://doi.org/10.1007/BF03181412`.

[87] B. Wieneke. Iterative reconstruction of volumetric particle distribution. *Measurement Science and Technology*, 24(2):024008, 2012. URL `https://dx.doi.org/10.1088/0957-0233/24/2/024008`.

[88] Y. Li, E. Perlman, M. Wan, Y. Yang, C. Meneveau, R. Burns, S. Chen, A. Szalay, and G. Eyink. A public turbulence database cluster and applications to study Lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, 9:N31, 2008. URL `https://doi.org/10.1080/14685240802376389`.

[89] T. Jefferson, L. Dooley, E. Ferroni, L. A. Al-Ansary, M. L. van Driel, G. A. Bawazeer, M. A. Jones, T. C. Hoffmann, J. Clark, E. M. Beller, P. P. Glasziou, and J. M. Conly. Physical interventions to interrupt or reduce the spread of respiratory viruses. *Cochrane Database of Systematic Reviews*, (1), 2023. URL `https://doi.org/10.1002/14651858.CD006207.pub6`.

[90] C. F. Bohren and D. R. Hoffman. Appendixes: Computer Programs. In *Absorption and Scattering of Light by Small Particles*, pages 475–476. John Wiley & Sons, Ltd, 1998. ISBN 978-3-527-61815-6. URL `https://doi.org/10.1002/9783527618156.app1`.

[91] H. C. van de Hulst. *Light scattering by small particles*. Wiley, 1957. ISBN 978-0-486-64228-4.

[92] L. Held and R. Havard. *Gaussian Markov Random Fields: Theory and Applications*. Chapman and Hall/CRC, New York, 2005. ISBN 978-0-429-20882-9. URL `https://doi.org/10.1201/9780203492024`.

Summary of papers

Paper I: High-speed imaging database of water jet disintegration Part I: Quantitative imaging using liquid laser-induced fluorescence

A. Roth, D. Frantz, W. Chaze, A. Corber, and E. Berrocal

This article focuses on creating data for the fundamental study of liquid jet disintegration. The article details two experiments on a water jet using fluorescence imaging, in contrast to the traditional shadowgraph imaging, where all experimental results are openly available in a database. The first experiment uses one high-speed camera recording at 40 000 frames per second where the transition from the Rayleigh to the atomization regime is captured by change of injection pressure. The second experiment employs two high-speed cameras simultaneously recording at 50 000 frames per second. With two cameras in a perpendicular configuration, it was possible to deduce a calibration that connects fluorescence intensity to liquid depth. To assess the quality of the predicted liquid depths, an extensive analysis of the liquid depth prediction accuracy, precision and volume preservation was performed.

E. Berrocal and W. Chaze performed the experiments and recorded all data where A. Corber designed the used injector. I performed all the post-processing with assistance from D. Frantz. This included among other calibration for intensity profile, extraction of calibration data points, estimation of calibration curve, the bias and variability analysis of the calibration curve, and the velocity analysis. I also made figures 10-25. E. Berrocal and A. Corber did most of the writing with some assistance from me and D. Frantz.

Paper II: Snapshot 3D reconstruction of liquid surfaces

A. Roth, E. Kristensson, and E. Berrocal

This paper details the technique Fringe Projection Laser Induced Fluorescence (FP-LIF) that we have created by combining two previously existing techniques. The first part of the paper details the Fringe projection technique that makes it possible to extract 3D information of a structure with a single camera and how the LIF part is required to image liquid surfaces. Then, information of the mathematical equations used in the 3D reconstruction is detailed. The FP-LIF technique is then validated on both a simulated drop structure and experimental pending liquid drops.

I implemented all the 3D reconstruction post-processing and created the open source `fp23dpy` *python package. This python package is generalized and can be used by others with different data and calibration parameters. I also performed the simulations and uncertainty analysis in connection to it. E. Berrocal recorded the data of the liquid drops, and I performed the post-processing on them. Finally, I wrote the manuscript with review and editing input from E. Berrocal and E. Kristensson.*

Paper III: Analysis of liquid surface deformation and breakups using three-dimensional high-speed data recorded with a single camera

A. Roth, M. Sapik, E. Kristensson, J. Jedelsky, and E. Berrocal

This article is a continuation of the work on the FP-LIF technique introduced in paper II . Here, it is applied in high-speed imaging of a hollow cone liquid sheet evolution. First the experimental setup is described followed by the high-speed 3D reconstruction results. The results show the evolution of the liquid surface from drop to jet to tulip to cone shape. Then, results of the stabilised conical sheet are shown in the form of 3D surfaces and 3D spray-angles where the latter has previously only has been extracted from 2D measurements. Finally, the tuning possibilities of the technique and its challenges are discussed.

M. Sapik and E. Berrocal recorded all the data. I performed the post-processing of all the data and wrote the manuscript. J. Jedelsky, E. Kristensson and E. Berrocal assisted with review and editing.

Paper IV: High-speed scattered-light imaging for sizing respiratory droplets

A. Roth, M. Stiti, D. Frantz, and E. Berrocal

In this paper an approach using Lorenz-Mie scattering intensity to size respiratory droplets is detailed. Two high-speed cameras are employed where one image the scattering light from droplets and the other is used to image the droplet size from a shadowgraph image. Now simultaneous detection of the droplet size and scattered intensity enables the estimation of a calibration curve to go from scattering intensity to size. The cameras were placed in the semi-forward scattering direction to utilize the higher scattering intensity that enable detection of smaller droplets. In addition, it is detailed how a too small illumination volume is not suitable for sizing droplets. The calibration curve and the prediction uncertainty are then explained with additional analysis of how the speed of droplets affect the range of droplet sizes that can be estimated. The sizing technique is applied to one cough of three different subjects where the droplet size averages are presented in both time and space. Finally, the experimental results are compared with a different sizing technique and tuning possibilities of this sizing is discussed.

M. Stiti, D. Frantz and E. Berrocal developed the experimental setup and performed the experiments. I was involved in the study conceptualization and recording of calibration images. I performed all post-processing of the data, created all figures, and wrote the manuscript. E. Berrocal, D. Frantz and M. Stiti assisted with review and editing.

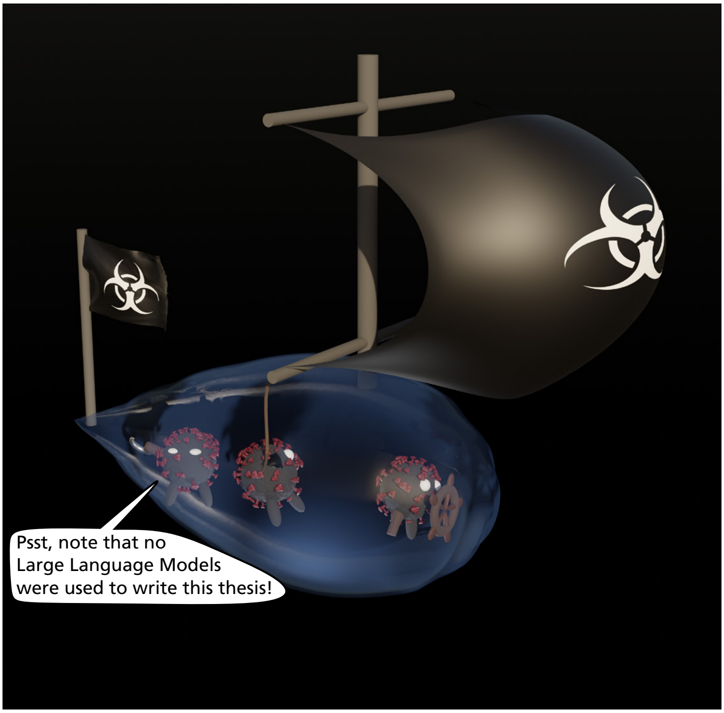
Paper v: Exhaled aerosols and saliva droplets measured in time and 3D space: Quantification of pathogens flow rate applied to SARS-CoV-2

A. Roth, M. Stiti, D. Frantz, A. Corber, and E. Berrocal

This work utilizes the sizing detailed in Paper iv and stereoscopic Particle Tracking Velocimetry to extract 3D velocities of respiratory particles. The latter use a similar configuration with the two high-speed cameras described in Paper iv where instead both cameras image the scattering intensity. In addition, a deeper analysis is performed of multiple respiratory events, where droplet deceleration and droplet virion content are included. Three different subjects are in this study speaking, yelling, or coughing. For each type of event, multiple recordings have been analysed to extract velocity and size. The velocity distributions are compared, and the size is used to extract the volume of ejected respiratory droplets. This volume was translated to the number of virions from an analysis of SARS COV-2 concentration in saliva. The virion numbers were analysed with respect to the different respiratory events to extract virion flow rate over time. For the coughs, both the temporal and spatial domain were analysed.

M. Stiti, D. Frantz and E. Berrocal developed the experimental setup and performed the experiments. I was involved in the study conceptualization and recording of calibration images. I performed all post-processing of the data, created all data analysis figures. E. Berrocal wrote the manuscript with assistance of A. Corber. I, D. Frantz and M. Stiti assisted with review and editing.

Printed by Media-Tryck, Lund 2023 NORDIC SWAN ECOLABEL 3041 0903



This thesis is intended for you with an interest in post-processing for 3D imaging and would not mind learning how some techniques can work on liquids. The front cover scene illustrates the three 3D imaging techniques used in this work. On the right is a liquid jet, on the left a liquid surface and in the centre a person coughing respiratory droplets. As a final treat, you find the so-called virusship above where these three virus pirates accompanied me on some of my popular science presentations.

Faculty of Engineering
Department of Physics

LRCP-247
ISBN 978-91-8039-803-9
ISSN 1102-8718
ISRN LUTFD2/TFCP-247-SE

LUND
UNIVERSITY