Synthetic Aperture Imaging for Three Dimensional Resolution of Fluid Flows

by

Jesse Belden

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2011

Author .....
Department of Mechanical Engineering
April 29, 2011

Certified by .....
Alexandra H. Techet
Associate Professor of Mechanical and Ocean Engineering
Thesis Supervisor

Accepted by .....
David E. Hardt
Ralph E. and Eloise F. Cross Professor of Mechanical Engineering
Chairman, Department Committee on Graduate Students

Synthetic Aperture Imaging for Three Dimensional Resolution of Fluid Flows

by

Jesse Belden

Submitted to the Department of Mechanical Engineering
on April 29, 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Mechanical Engineering

Abstract

Fluid mechanics and instrumentation have a long history together, as experimental fluids studies play an important role in describing a more complete physical picture in a variety of problems. Presently, state-of-the-art instruments for fluid flows aim to resolve various quantities in three-dimensions. This thesis describes a novel three-dimensional imaging system intended to extend laboratory measurement capabilities in complicated flows where knowledge is incomplete. In particular, the imaging system is designed to perform three-dimensional velocimetry in densely seeded flows where object geometry may partially occlude the field as well as to measure and locate bubbles, droplets and particles in three-dimensions in multiphase flows. An instrument of this kind has ramifications in a variety of engineering applications from air-sea interaction to Naval hydrodynamics to turbulence and beyond.

The imaging system is based upon synthetic aperture (SA) imaging, which has received much attention in the computer vision community recently[1]. In focus images from an array of synchronized cameras are recombined in software post-capture using a refocusing algorithm to generate a focal stack of synthetic images. Each synthetic image has a narrow depth of field, and objects residing at this depth appear sharp while off-plane objects appear blurred. The refocusing algorithm not only allows for 3D reconstruction of a scene, but also enables "see-through" effects, whereby an object occluded in some of the camera views will be seen in the synthetic images.

In this thesis, considerations for development of a three-dimensional measurement system for fluid flows based on the SA imaging field are made. A high-performance

[1] M. Levoy. Light fields and computational imaging. Computer, 39(8):46-55, Aug. 2006.

three-dimensional particle image velocimetry technique is described and validated. Also, a method for auto-calibration of mutli-camera setups for fluids experiments is derived and developed. Finally, algorithms are generated for application to multiphase flows and the technique is applied to a circular plunging jet with results showing excellent agreement to prior literature and yielding new insight into the problem.

Thesis Supervisor: Alexandra H. Techet
Title: Associate Professor of Mechanical and Ocean Engineering

Acknowledgments

ΓΝΩΘΙ ΣΕΑΥΤΟΝ

I never would have finished this journey without support and inspiration from many people along the way. I first want to thank my loving wife Mary, who has been with me even longer than I have been in college. Without your love, support, and patience this would not have been possible. Countless times I was disheartened, grumpy, nervous or anxious and every time you made it all right with your smile, perspective and laughter, for which we both share a passion. Anyhow, I really had no reason to complain while writing this thesis, as you were the one who was pregnant and still teaching seventh graders. I also want to thank my son Liam, who at the time of this writing is not yet born. You have no idea how much motivation you provided. To my parents, your love and support has guided me throughout and your example of hard work balanced by vigorous livelihood has provided me with a fruitful template for living my life. To my brother Calum, the Marine, who at the time of this writing is on a ship off the coast of Japan in the wake of a devastating earthquake - boy, talk about perspective! Cale, your sense of humor about my work humbles me and your dedication to yours inspires me.

I want to thank my advisor, Professor Alexandra Techet, for her support through the years. I especially thank you for allowing me to pursue this thesis work when it was in its nascent stages and the outcome wasn't clear, and for asking the right questions to keep me on track. For your understanding of my family life, dealing with my linear one-track way of thinking and for sending me to some great conferences in wonderful places I thank you as well.

Great thanks to my committee for their guidance on this thesis. To Professor Doug Hart for his enthusiasm about the project, expertise in the subject and for

pushing the work toward new applications. To Professor Ian Hunter for his incredible depth and breadth of knowledge and for asking questions that never occurred to me, which made me dig deeper into the work. And to Professor George Barbastathis for bringing his expertise in the area, stimulating thoughtful discussion and always keeping the meetings lively.

I want to thank two very special people who influenced me greatly during my time at MIT. Tadd Truscott and Brenden Epps, you are great friends and selfless mentors. Tadd, without your encouragement I wouldn't have pursued this project so vigorously. Your uncanny enthusiasm is infectious, and I can't wait to collaborate on future endeavors. And thanks for teaching me how to take beautiful pictures. Brenden, your patience in working with me on difficult problems is greatly appreciated and your sense of humor in the lab was greatly welcomed.

Many, many others deserve acknowledgement. Dr. Sai Ravela for sharing knowledge, giving constructive feedback on my work and collaborating with me on projects of mutual interest. Professor Ramesh Raskar for introducing me to the field of computational imaging and inspiring the beginnings of this thesis. Michael Axiak for getting version 1 of the camera array up and running. Roderick La Foy for his early work with me on 3D imaging and other imaging projects. Anna Shih and Jenna McKown for their camaraderie in the lab. Paco Flores for keeping me fit. Wayne Staats for keeping me "tanked up." Jeff Aristoff for setting a lofty standard of what a researcher should be. And to the new crop in the lab - Tim Gruber, Amy Gao, Barry Scharfman and Ben Johnson - for bringing new enthusiasm.

Surely there are many others deserving of acknowledgement, but I can't keep my readers waiting anymore.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The field of fluid dynamics presents tremendous challenges to theoretical, experimental and computational methods owing to the wide-ranging length and time scales, complicated multiphase physics and coupled flow-structure dynamics, among other challenges. The ubiquitous impact of the field on engineering applications ranging from Naval hydrodynamics to renewable energy to biology to microscale devices demands active advances in all methods. With respect to theory, closed form solutions to the governing equations (typically at least Navier-Stokes) are rare, thus for many problems of interest the equations require simplifications in order to be solved. Computational methods are constantly advancing to accommodate more and more complicated situations. In each case, experimental investigation of the problem is often required to validate either the theory or computational method.

It should come as no surprise that fluid dynamics and instrumentation have had an integral relationship for many years. Much like the complete closed form Navier-Stokes solution, or the infinitely resolved computational scheme, the fully four dimensional (3 space + time) instrument capable of measuring all relevant fluid quan-

tities does not yet exist. A variety of instruments have been developed to measure the various quantities of interest in 1-3 dimensions with various levels of resolution; herein, the focus will lie on instrumentation for velocity measurements and multiphase flows. Velocimetry in fluids has seen the evolution from point measurements of velocity using pitot-static tubes [1], hot-wire anemometry [2], or laser doppler anemometry [3] to two-dimensional resolution of two or three components of velocity using particle image velocimetry (PIV) [4] or Stereo-PIV [5]. The developing state-of-the-art in velocimetry is three-dimensional, three-component (3D-3C) velocity field measurements [6, 7, 8, 9, 10]. Measurements of the bubble, spray or droplet properties in multiphase flows followed a similar evolution from point measurements using conductivity or optical probes [11] to two-dimensional measurement of bubble or spray properties using planar shadow imaging [12]. Three-dimensional measurements in multiphase flows are also available [7, 13], but are not as developed as 3D velocimetry. For both velocimetry and multiphase flow measurements, many of the state-of-the-art 3D instruments are based on imaging. However, many of these imaging systems often suffer limitations on resolution and/or total resolvable volume size, possess limited applicability in situations with optical occlusions due to objects in the flow, or often do not extend to measurement in multiphase flow regimes.

This thesis describes a novel three-dimensional imaging system intended to extend laboratory measurement capabilities in complicated flows where knowledge is incomplete. In particular, the imaging system is designed to perform three-dimensional velocimetry in densely seeded flows where object geometry may partially occlude the field as well as to measure and locate bubbles, droplets and particles in three-dimensions in multiphase flows. The target volume sizes have typical macroscale laboratory length scales ($O(10^{-3}m) - O(1m)$).

Before delving into the specifics of the imaging system described in this thesis,

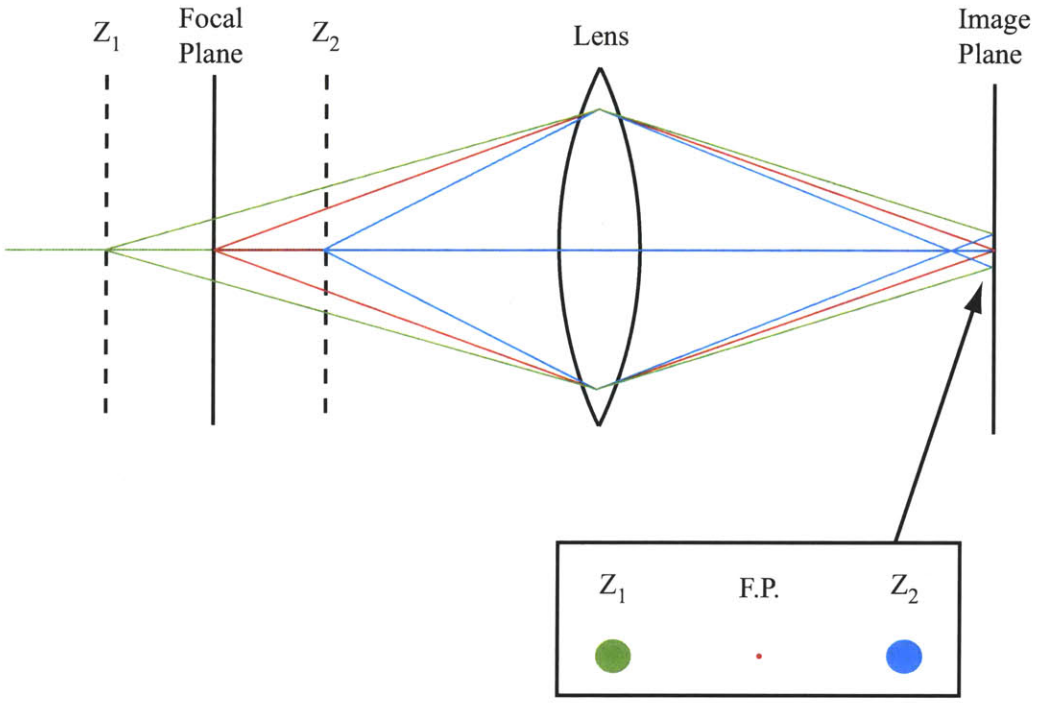
perhaps we should first examine some areas in fluid dynamics where information is lacking. A report by the U.S. National Committee on Theoretical an Applied Mechanics [14] presented key research areas in fluid dynamics. While a single instrument cannot serve every area, efforts for advancing instrumentation to meet needs in a variety of areas outlined in the report are warranted. Turbulence has long been an area of active research and affects many engineering applications related to particle transport [15], air-sea interaction [16] and mixing [17], to name a few. The report [14] cites the advances in understanding of turbulence due to experimental methods, and cites the need for advances in three-dimensional measurement to better understand the physics across multiple scales [14]. Also highlighted in the report are the challenges associated with modeling environmental flows, which span multiple scales. Models often incorporate sensor data [14], and thus understanding processes on a local scale (for example, fundamental physics of air-sea interaction [16, 18]) may serve to improve accuracy in the larger model (e.g., hurricane forecasting). Finally, flow control appears in many engineering settings and involves the interaction of actuators with the fluid [14]. Therefore, understanding the fluid-structure interaction aids in modeling, design and development of control strategies for flow control devices. While important fluid dynamics topics extend far beyond those mentioned here, this thesis will focus on advancing instrumentation in the areas of three-dimensional velocimetry, measurement in three-dimensional multiphase flows and extension to fluid-structure interaction.

The previous discussion motivates two performance requirements of the imaging system: the ability to determine the location of objects in three-dimensional space, and the capacity to image in flows where partial occlusions exist and still faithfully reconstruct occluded objects. The foundation for each requirement is built by considering a classic single lens imaging system. One might expect any 3D imaging system

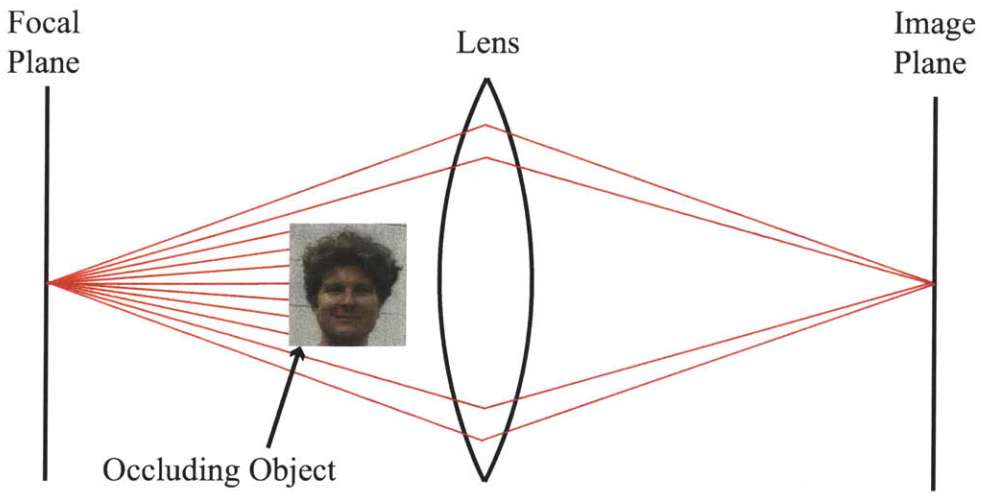
to somehow "encode" depth, as an imaging system in its most basic form compresses a 3D world onto a 2D plane. Consider the schematic of a classic single lens imaging system shown in Figure 1-1(a). If the focal plane is infinitely thin, then rays from objects off the focal plane will intersect the image plane within a circle (neglecting diffraction) resulting in a blurry image; the radius of the circle increases with increasing distance from the focal plane. In this way, even a single lens system encodes depth to a degree; however, the circle is known as the "circle of confusion" [19], as an ambiguity exists as to whether the object lies in front of or behind the focal plane. Furthermore, to extract the depth location of objects from a blurred image requires detecting changes in the blur pattern, which is difficult for a normal circular aperture [20]. To extract 3D information from an imaging system then requires unambiguous encoding of depth, combined with some post capture computation to decode the depth. Therefore, 3D imaging methods typically fall under the field of *computational imaging*, which combines non-standard image capture methods with post-capture processing [19].

Additionally, Figure 1-1(b) schematically shows the ability of a single lens system to "see-through" a partial occlusion if the occluding object is smaller than the lens aperture, as discussed by Levoy [19]. Many rays from objects on the focal plane still reach the image plane to form an image. However, for the fluids flows of interest, the occluding objects may often be as big or bigger than the lens aperture. Also, the see-through effect only allows for formation of images of objects on a small focal plane, but we are interested in obtaining information about objects distributed throughout a volume with large depth dimension.

Synthetic aperture (SA) imaging using camera arrays developed as a means for extending 3D reconstruction and see-through effects to larger scales [19, 21, 22, 23, 24, 25]. As described by Levoy [19], placing several cameras in an array is analogous



(a) circle of confusion



(b) see-through effects

Figure 1-1: Schematic of the "circle-of-confusion" in a traditional camera (a) and ability of a single lens camera to "see-through" partial occlusions (b).

to imaging with a large lens as the cameras sample rays over a large aperture. Each camera captures an image of the scene entirely in focus, and individual images are recombined in software to yield a synthetic image that appears to focus on an isolated plane [21]. Furthermore, the depths of objects are encoded by the parallax between cameras [24], as shown schematically for the nine camera array in Figure 1-2. By establishing the relationship between image and world points, each camera image can be reprojected back onto surfaces in the volume and averaged to form a refocused image [24]. For example, if the images in Figure 1-2 are reprojected onto the plane at Z_1 and averaged, then point A will appear in sharp focus in the refocused image, while point B will appear blurred due to the parallax. Adding more cameras reduces the signal from off plane points. The ability to see-through partial occlusions is inherent in the image refocusing, while the object depth is encoded by parallax and quantified by the image to world point mapping. Figure 1-3 shows an application of the SA refocusing method to images taken of an object placed in a bubbly flow captured using a ProFusion 25 camera array. The bubbles provide a large amount of occlusion of the object in any one image (Figure 1-3(b)), but after application of the SA refocusing algorithm, much of the detail about the object can be seen (Figure 1-3(c)). Refocused images can be generated at several depths to create a focal stack, similar to wide-field microscopy [26, 27], from which a 3D reconstruction is performed.

A brief review of other 3D imaging methods aimed at resolution on the macro scale is now presented. Several other methods exist for resolving 3D scenes, but the methods of data acquisition seem to fall into three broad categories: internal optics alteration, holography and multiple-viewpoints. Internal optics alteration involves placing additional optical element(s) in the optical pipeline to alter the formed images in some controlled fashion. In the case of coded aperture imaging (e.g. [28, 29, 20]),

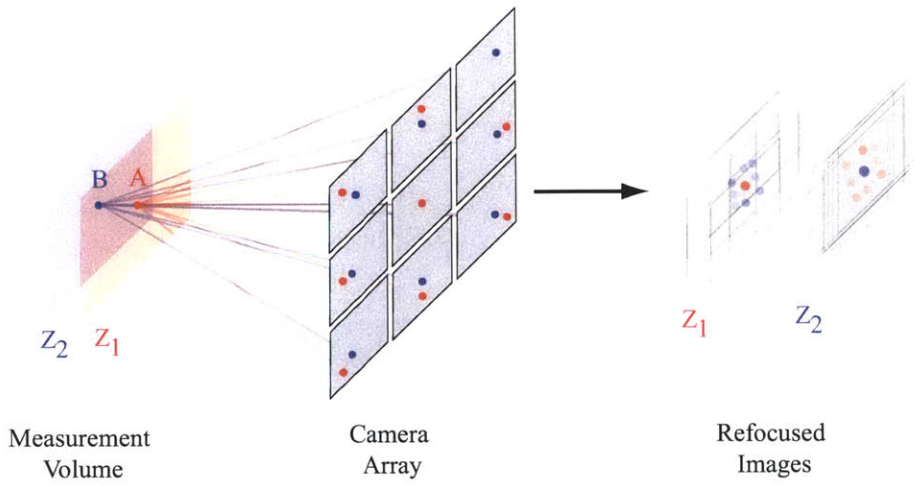
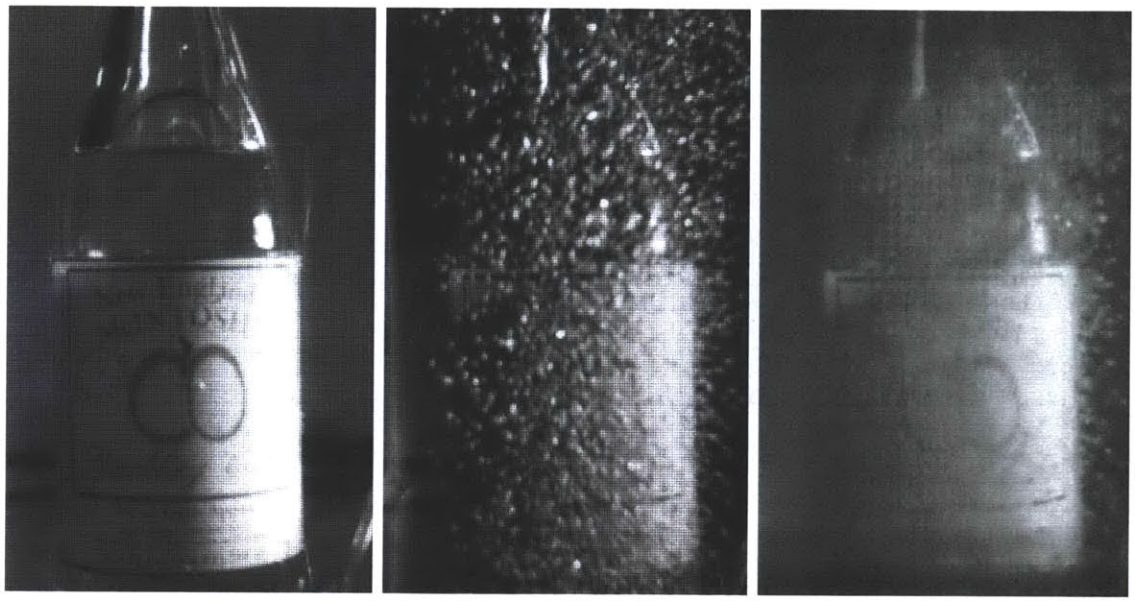


Figure 1-2: Schematic ishowing a 3D depiction of image capture by a 9 camera array, and subsequent refocusing on two planes.

a mask is inserted at the aperture plane, which encodes the blur function such that the deconvolution problem is not ill-posed and allows for digital refocusing at different depths post-capture. However, coded aperture does not solve the problem of reconstructing scenes with partial occlusions. The active wavefront sampling (AWS) technique makes use of an eccentric rotating aperture to generate images from different viewpoints on a single image sensor [30]. Another internal optics alteration involves placing a microlens array after the main lens but before the image sensor plane [31, 32]. Imaging systems of this type enable capture of a light field [19], which allows for many post-capture processing benefits such as generating synthetic views from different viewpoints, refocusing on different depth planes and potentially even reconstructing scenes with partial occlusions [31, 32, 33, 34]. Using a microlens array requires sacrificing some overall X-Y viewable dimensions as the microlenses form images on subregions of the image sensor. Furthermore, it seems as though the size of occluding object that can be "seen through" would be smaller than for SA imag-



(a) original object with no bubbles

(b) object in bubbly flow

(c) SA refocused image

Figure 1-3: Raw image of an object placed in a water tank (a), raw image of the same object placed behind a bubble field (b) and SA refocused image using raw images from all 25 cameras demonstrating the see-through capability of the SA imaging method.

ing, as the baseline over which the sampling points are spaced is smaller, but there appears to be very little literature on refocusing with occlusions using a microlens light field camera. It should be pointed out that SA imaging is a subset of light field imaging [32].

Holography is a technique in which the three-dimensional locations of objects in a volume are deduced from the interference pattern of the light waves emanating from objects and a coherent reference wave that is incident upon the field [8]. The nature of the interference pattern is used to back out information about the phase of light diffracted from objects in the volume, which is related to the distance of the objects from the sensor (i.e., depth in the volume) [35]. Holography has found many application areas in fluid mechanics, especially with the availability of large format digital sensors. Holographic PIV has been developed for resolving 3D velocity fields [36, 8, 37, 38] and is reviewed more thoroughly in Chapter 2. Holography is also used in identification of biological samples in the ocean [39, 40, 41]. Finally, holographic systems are being used to measure the size of bubbles and droplets in multiphase flows as well [13]. Holography is typically limited to measurement in volumes with low void fraction or particle density because occlusions cannot be effectively dealt with [37, 13].

Multiple viewpoint techniques involve either moving cameras or placing cameras over distributed locations, and a variety of algorithms exist for reconstructing 3D scenes from multiple viewpoints. Moving cameras is not an option for fluid measurements where each image must be captured simultaneously due to the motion of the fluid. Stereo imaging is well-established in the computer vision community [42, 43]. As described by Vaish et al. [43], to obtain the see-through effects, multiple cameras must be used in conjunction with appropriate algorithms, which falls under the heading of SA imaging. Synthetic aperture imaging is a form of multi-baseline stereo

imaging, with algorithms that allows for objects to be missing in some views, but still enables reconstruction in 3D [43]. Optical tomography involves solving the inverse problem of determining 3D objects from projections onto image planes [44, 45]. If the sensors are distributed over a large angle ($\geq 180^o$), as for medical imaging, then accurate and efficient algorithms for reconstruction are available [45]. However, in fluids experiments, the sensors are not normally distributed over such a large angle and algebraic reconstruction techniques (ART) are required [45, 46]. Three-dimensional PIV techniques based on ART and variations have developed and are becoming more widely used in the community [46, 9, 47]; tomgraphic-PIV is described in more detail in Chapters 2 and 3. As discussed in Chapters 2 and 3, ART based tomographic techniques do not effectively deal with occlusions. Finally, inspired by coded aperture imaging, digital defocusing particle image velocimetry (DDPIV) generates a single composite image from three or more views and determines object depth by measuring the size of the coded "blur" pattern [48, 7, 49]. DDPIV can accommodate occlusions, but the fields can not be too optically dense as each pattern must be identified by the search algorithm. Bubbly flows have also been resolved with DDPIV systems, typically with low optical density [7, 50].

Although the ability of SA imaging to deal with partial occlusions and encode the 3D location of objects are promising attributes, developing SA imaging into a measurement system requires several considerations that are detailed in this thesis. In particular, we will often aim to extract information from the refocused images (e.g. geometry of a bubble) and determine accurately the plane of best focus for an object. To accurately place extracted information in a world coordinate frame requires accurate calibration. For many fluids studies, calibration requires consideration of the nature of the imaging system, where an air-glass-water transition exists between the cameras and volume of interest. Much of the available literature on SA

imaging focuses on implementation of the refocusing algorithm to produce images for observation only, with particular application to surveillance [22, 23, 24]. The technique has also found applications in display and novel view rendering in the computer vision community [21]. Typically, multi-camera arrays for SA imaging are calibrated for environments where the pinhole model applies [21] or plane-to-plane homographies are used to generate the refocused planes [22, 23, 24]. As will be shown in Chapter 4, linear camera or plane-to-plane models fail to adequately describe the common fluid flow setup involving air-glass-water transition in the camera lines of sight. Determining the plane of best focus is part of the 3D reconstruction problem, and is common in computer vision [51, 43, 52]. Also, many holographic reconstructions apply focus metrics to determine object depth [13, 53, 41]. Vaish et al. [43] discussed methods for 3D reconstruction specifically from SA refocused images based on various metrics (e.g. shape from focus and shape from stereo). In this thesis, metrics for 3D reconstruction and subsequent feature extraction for 3D PIV and SA refocused bubble images are described in Chapters 2 and 5, respectively.

Chapter 2 presents the development of a three-dimensional particle image velocimetry technique based on the synthetic aperture imaging system (dubbed 3D SAPIV) and comes directly from an article published in *Measurement Science & Technology* [10]. The 3D SAPIV technique enables larger seeding densities and thus improved resolution over the current state-of-the-art in 3D PIV. A canonical vortex ring flow is imaged with a low-cost camera array and serves as a benchmark for the 3D SAPIV technique, and results are in excellent agreement with the expected outcome and similar 2D PIV results.

In Chapter 3, several practical aspects of synthetic aperture imaging are discussed. The design of the low-cost camera array used in Chapter 2 is described in further detail, as is a high-speed camera array. Then, comparisons between synthetic

aperture imaging and ART-based optical tomography are drawn, particularly with respect to the computational performance, the ability of each algorithm to handle partial occlusions and the amount of "ghost particles" (or reconstruction artifacts) present in 3D PIV volumes reconstructed with each method. The current implementations of the synthetic aperture imaging algorithms are shown to outperform the tomographic algorithms in both computational performance and in reducing ghost particles, and have the ability to reconstruct volumes with partial occlusions, which the discussed implementations of tomography do not possess.

Chapter 4 describes a novel method of auto-calibration of multiple cameras. Calibration of multiple cameras presents a huge challenge if handled with traditional methods. However, if information is shared across cameras, then the calibration can be simplified greatly. The cameras can be calibrated simply from images of corresponding points and a small amount of reference geometry. In Chapter 4, the calibration model accounts for a common situation in fluids experiments in which the cameras are placed in air and view a scene in water through a wall. Cameras can be in general location and orientation, and the induced refraction is accounted for. The algorithm for auto-calibration of cameras of this nature is developed and represents one of the contributions of this thesis.

Chapter 5 discusses the application of the synthetic aperture imaging technique to the bubbly flow induced by a circular plunging jet. This classic problem presents an ideal test case as the bubble concentration and size is readily varied by altering the jet height. Although the circular plunging jet is not a new problem, certain aspects remain unresolved. In particular, a three-dimensional bubble size distribution has never been reported due to the complexity of the flow and the lack of suitable instrumentation. Results from the experiment demonstrate the capability of the synthetic aperture imaging technique to resolve bubble sizes and locations in three-dimensions

in flows with relatively high air concentration. The resulting three-dimensional size distributions represent the first data sets of their kind, to the knowledge of the author. Some insights are drawn for a range of experimental parameters and compared with previous work on the problem.

Finally, Chapter 6 presents conclusions for the entire thesis, summarizes the SA imaging measurement technique with a "road map" for SA projects and highlights the contributions of the work. The appendices contain several important but peripheral details not included in the main text. Some details of the numerical scheme used to solve part of the calibration in Chapter 4 and additional plots from Chapter 4 are included.

Bibliography

[1] R.G. Folsom. Review of the pitot tube. In *Trans. ASME*, volume 78, pages 1447–1460, 1956.

[2] G Comte-Bellot. Hot-wire anemometry. *Annual Review of Fluid Mechanics*, 8(1):209–231, Jan 1976.

[3] C Tropea. Laser doppler anemometry: recent developments and future challenges. *Measurement Science and Technology*, 6(6), 1995.

[4] R. J. Adrian. Twenty years of particle image velocimetry. *Experiments in Fluids*, 39(2):159–169, 08 2005.

[5] A. K. Prasad. Stereoscopic particle image velocimetry. *Experiments in Fluids*, 29:103–116, 2000. 10.1007/s003480000143.

[6] H. G. Maas, A. Gruen, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(2):133–146, 1993-07-01.

[7] F. Pereira, M. Gharib, D. Dabiri, and D. Modarress. Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. Application to bubbly flows. *Experiments in Fluids*, 29(7):S078–S084, 2000.

[8] K. D. Hinsch. Holographic particle image velocimetry. *Measurement Science and Technology*, 13(7):R61–R72, 2002.

[9] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.

[10] Jesse Belden, Tadd T. Truscott, Michael Axiak, and Alexandra H. Techet. Three-dimensional synthetic aperture particle image velocimetry. *Measurement Science and Technology*, 21(12), 2010.

[11] H. Chanson. Air-water flow measurements with intrusive, phase-detection probes: Can we improve their interpretation? *Journal of Hydraulic Engineering*, 128(3):252–255, March 2002.

[12] D Bröder and M Sommerfeld. Planar shadow image velocimetry for the analysis of the hydrodynamics in bubbly flows. *Measurement Science and Technology*, 18(8):2513, July 2007.

[13] Lei Tian, Nick Loomis, Josè A. DomÌnguez-Caballero, and George Barbastathis. Quantitative measurement of size and three-dimensional position of fast-moving bubbles in air-water mixture flows using digital holography. *Appl. Opt.*, 49(9):1549–1554, 03 2010.

[14] Jerry Gollub. Research in fluid dynamics: Meeting national needs. Technical report, U.S. National Committee on Theoretical and Applied Mechanics, 2006.

[15] Juan P. L. C. Salazar, Jeremy De Jong, Lujie Cao, Scott H. Woodward, Hui Meng, and Lance R. Collins. Experimental and numerical investigation of inertial particle clustering in isotropic turbulence. *Journal of Fluid Mechanics*, 600:245–256, 2008.

[16] W K Melville. The role of surface-wave breaking in air-sea interaction. *Annual Review of Fluid Mechanics*, 28(1):279–321, 1996.

[17] Paul E. Dimotakis. Turbulent mixing. *Annual Review of Fluid Mechanics*, 37(1):329–356, Jan 2005.

[18] H. Chanson, S. Aoki, and A. Hoque. Bubble entrainment and dispersion in plunging jet flows: Freshwater vs. seawater. *Journal of Coastal Research*, 22(3):664–677, June 2006.

[19] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, Aug. 2006.

[20] Anat Levin, Rob Fergus, Frédo Durand, and William T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph.*, 26, July 2007.

[21] Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. Dynamically reparameterized light field. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 297–306, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[22] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. *ACM Trans. Graph.*, 24(3):765–776, 2005.

[23] V. Vaish, G. Garg, E. Talvala, E. Antunez, B. Wilburn, M. Horowitz, and M. Levoy. Synthetic aperture focusing using a shear-warp factorization of the viewing transform. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05') - Workshops*, volume 3, pages 129–129. IEEE Computer Society, June 2005.

[24] V. Vaish, B. Wilburn, N. Joshi, and M. Levoy. Using plane + parallax for calibrating dense camera arrays. In *Proceedings of the 2004 IEEE Computer*

Society Conference on Computer Vision and Pattern Recognition (CVPR04'), volume 1, pages 2–9. IEEE Computer Society, June-July 2004.

[25] Myungjin Cho and Bahram Javidi. Three-dimensional tracking of occluded objects using integral imaging. *Opt. Lett.*, 33(23):2737–2739, 12 2008/12/01.

[26] Peter Shaw. Deconvolution in 3-d optical microscopy. *The Histochemical Journal*, 26:687–694, 1994. 10.1007/BF00158201.

[27] Marc Levoy, Ren Ng, Andrew Adams, Matthew Footer, and Mark Horowitz. Light field microscopy. *ACM Trans. Graph.*, 25:924–934, July 2006.

[28] E. E. Fenimore and T. M. Cannon. Coded aperture imaging with uniformly redundant arrays. *Appl. Opt.*, 17(3):337–347, 02 1978.

[29] Ashok Veeraraghavan, Ramesh Raskar, Amit Agrawal, Ankit Mohan, and Jack Tumblin. Dappled photography: mask enhanced cameras for heterodyned light fields and coded aperture refocusing. *ACM Trans. Graph.*, 26, July 2007.

[30] J. Rohaly and D.P Hart. Monocular 3-d active micro-ptv. In *4th International Symposium on Particle Image Velocimetry*, Gottingen, Germany, September 2001.

[31] E.H. Adelson and J.Y.A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, February 1992.

[32] Ren Ng, Marc Levoy, Mathieu Brédif, Gene Duval, Mark Horowitz, and Pat Hanrahan. Light field photography with a hand-held plenoptic camera. Technical Report CTSR 2005-02, Stanford University, 2005.

[33] Jae-Hyeung Park, Keehoon Hong, and Byoungho Lee. Recent progress in three-dimensional information processing based on integral imaging. *Appl. Opt.*, 48(34):H77–H94, 12 2009/12/01.

[34] Lei Tian. Phase-space representation of digital holographic and light field imaging with application to two-phase flows. Master's thesis, Massachusetts Institute of Technology, 2010.

[35] Joseph W. Goodman. *Introduction to Fourier Optics.* Roberts & Company, Greenwood Village, CO, USA 80111, 3 edition, 2005.

[36] J. Zhang, B. Tao, and J. Katz. Turbulent flow measurement in a square duct with hybrid holographic piv. *Experiments in Fluids*, 23(5):373–381, 11 1997.

[37] Hui Meng, Gang Pan, Ye Pu, and Scott H. Woodward. Holographic particle image velocimetry: from film to digital recording. *Measurement Science and Technology*, 15(4):673–685, 2004.

[38] J. Sheng, E. Malkiel, and J. Katz. Using digital holographic microscopy for simultaneous measurements of 3d near wall velocity and wall shear stress in a turbulent boundary layer. *Experiments in Fluids*, 45(6):1023–1035, 12 2008/12/01/.

[39] Jerome H. Milgram and Weichang Li. Computational reconstruction of images from holograms. *Appl. Opt.*, 41(5):853–864, 02 2002.

[40] J. Katz, P. L. Donaghay, J. Zhang, S. King, and K. Russell. Submersible holocamera for detection of particle characteristics and motions in the ocean. *Deep Sea Research Part I: Oceanographic Research Papers*, 46(8):1455–1481, 1999.

[41] Weichang Li, Nick C. Loomis, Qiao Hu, and Cabell S. Davis. Focus detection from digital in-line holograms based on spectral l1 norms. *J. Opt. Soc. Am. A*, 24(10):3054–3062, 10 2007/10/01.

[42] Qian Chen and G. Medioni. A volumetric stereo matching method: Application to image-based modeling. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 1, Fort Collins, CO, 1999.

[43] Vaibhav Vaish, Marc Levoy, Richard Szeliski, C. L. Zitnick, and Sing Bing Kang. Reconstructing occluded surfaces using synthetic apertures: Stereo, focus and robust measures. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 2 (CVPR'06)*, volume 2, pages 2331–2338, 2006.

[44] Gabor T. Herman and Arnold Lent. Iterative reconstruction algorithms. *Computers in Biology and Medicine*, 6(4):273–294, 1976.

[45] A. C. Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging.* Society of Industrial and Applied Mathematics, 2001.

[46] Callum Atkinson and Julio Soria. An efficient simultaneous reconstruction technique for tomographic particle image velocimetry. *Experiments in Fluids*, 47:553–568, 2009. 10.1007/s00348-009-0728-0.

[47] Gerrit Elsinga, Bernhard Wieneke, Fulvio Scarano, and Andreas Schrder. Tomographic 3d-piv and applications. In *Particle Image Velocimetry*, volume 112 of *Topics in Applied Physics*, pages 103–125. Springer Berlin / Heidelberg, 2008.

[48] C. E. Willert and M. Gharib. Three-dimensional particle imaging with a single camera. *Experiments in Fluids*, 12:353–358, 1992. 10.1007/BF00193880.

[49] Francisco Pereira and Morteza Gharib. Defocusing digital particle image velocimetry and the three-dimensional characterization of two-phase flows. *Measurement Science and Technology*, 13(5):683–694, 2002.

[50] Emilio Castaño Graff, Francisco Pereira, Michele Milano, and Mory Gharib. The capabilities of defocused digital particle image velocimetry. In *6th International Symposium on Particle Image Velocimetry*, Pasadena, California, USA, September 2005.

[51] Yoav Y. Schechner and Nahum Kiryati. Depth from defocus vs. stereo: How different really are they? *International Journal of Computer Vision*, 39:141–162, 2000. 10.1023/A:1008175127327.

[52] S. Baker, R. Szeliski, and P. Anandan. A layered approach to stereo reconstruction. *Computer Vision and Pattern Recognition, 1998. Proceedings. 1998 IEEE Computer Society Conference on*, pages 434–441, June 1998.

[53] Frank Dubois, Cédric Schockaert, Natacha Callens, and Catherine Yourassowsky. Focus plane detection criteria in digital holography microscopy by amplitude analysis. *Optics Express*, 14(13):5895–908, Jun 2006.

Chapter 2

Three-Dimensional Synthetic Aperture Particle Image Velocimetry [1]

2.1 Introduction

Efforts for resolving three-dimensional velocity fields are justified by the need to experimentally resolve flows that are highly three-dimensional and to validate numerical simulations of complex flows. The ability to spatio-temporally resolve flow features from small to large scales in arbitrarily large volumes is the goal of any 3D PIV system. Of course, there have been many roadblocks to achieving all of these goals with a single system, and compromises must be made. Two-dimensional particle image velocimetry (2D PIV) is the most pervasive method for resolving velocity fields, thus it is not surprising that recent efforts to resolve 3D flow fields have extended many of the fundamentals of 2D PIV to the third dimension.

Several methods exist for resolving 3D particle fields, or any 3D scenes for that matter, but the methods of data acquisition seem to fall into three broad categories: multiple-viewpoints, holography and internal optics alteration. The technique described herein falls into the multiple-viewpoint category, and makes use of an algorithm known as *synthetic aperture refocusing* to examine the imaged volume and is thus referred to as synthetic aperture PIV (SAPIV). Herein, we focus on the application of the principles of synthetic aperture imaging to develop a measurement system for resolving three-dimensional fluid velocity fields. The system performance is evaluated theoretically and numerically. The practical utility of SAPIV is demonstrated through an experimental study of a canonical vortex ring.

The evolution of 2D PIV is described by Adrian [2], and is not reviewed here. One of the earliest, but still frequently utilized, methods for 3D PIV is two camera stereoscopic PIV, which is primarily used to resolve the third component of velocity within a thin light sheet [3]. Maas et al. [4, 5] and Malik et al. [6] use a 3D particle tracking velocimetry (PTV) method which resolves the location of individual particles imaged by two, three or four cameras in a stereoscopic configuration. They report measurements in a large volume (e.g. 200 x 160 x 50 mm^3)[1], but with very low seeding density ($\approx$1000 particles). Through precise calibration and knowledge of the imaging geometry, the particle field can be reconstructed. More recently, improvements to PTV methods are presented by Willneff & Gruen [7]. In general, low seeding density is a typical limitation of PTV, yielding low spatial resolution in the vector fields.

Another technique which makes use of multiple viewpoints is defocusing digital particle image velocimetry (DDPIV) [8, 9, 10]. In theory, DDPIV capitalizes on the defocus blur of particles by placing an aperture with a defined pattern (usually

[1]The notation mm^3 will be used throughout this thesis to refer to cubic millimeters.

pinholes arranged as an equilateral triangle) before the lens, which is a form of coded aperture imaging [11]. The spread between three points generated by imaging a single particle corresponds to the distance from the camera along the Z dimension. In practice, the spread between particles is achieved using three off-axis pinhole cameras which causes a single point in space to appear at separate locations relative to the sensor of each camera. As described in [12, 13], the images from all three camera sensors are superimposed onto a common coordinate system, an algorithm searches for patterns which form an equilateral triangle, and based on size and location of the triangle the 3D spatial coordinates of the point can be resolved. A main limitation of this technique appears to be seeding density, because the equilateral triangles formed by individual particles must be resolved to reconstruct the particle field. Pereira & Gharib [9] have reported simulations with seeding density of 0.038 particles per pixel (ppp) in a volume size of 100 x 100 x 100 mm^3, and experiments with seeding density of 0.034 ppp in a volume size of 150 x 150 x 150 mm^3. The technique has also been efficiently implemented with a single camera using an aperture with color-coded pinholes, to measure velocity fields in a buoyancy driven flow in a 3.35 x 2.5 x 1.5 mm^3 volume with seeding density $\approx$0.001 ppp [14].

Tomographic-PIV also uses multiple viewpoints (usually 3-6 cameras) to obtain 3D velocity fields [15, 16]. Optical tomography reconstructs a 3D intensity field from the images on a finite number of 2D sensors (cameras); the intensity fields are then subjected to 3DPIV cross-correlation analysis. The seeding density for tomographic-PIV seems to be the largest attainable of the existing techniques. Simulations by Elsinga et al. [15] show volumetric reconstruction with seeding density of 0.05 ppp, and recent tomographic-PIV experiments typically have seeding density in the range of 0.02-0.08 ppp [16, 17]. The viewable depth of volumes in tomographic-PIV is typically three to five times smaller than the in-plane dimensions [15, 16, 18]. Elsinga

et al. [15, 16] thoroughly characterize the performance of tomographic-PIV, and we have adopted many of the same metrics in evaluating the synthetic aperture PIV method presented herein since both methods are based on 3D cross-correlation of reconstructed intensity fields.

Holographic PIV (HPIV) is a technique in which the three-dimensional location of particles in a volume is deduced from the interference pattern of the light waves emanating from particles and the coherent reference wave that is incident upon the field [19]. The nature of the interference pattern is used to back out information about the phase of light diffracted from objects in the volume, which is related to the distance of the objects from the sensor (i.e. depth in the volume) [20]. Holographic PIV makes use of this principle to image particle-laden volumes of fluids, and extract information about location of particles in the volume. Meng et al. [21] provide an extensive review of film and digital Holographic PIV techniques. In holography, the size of the observable volume is ultimately limited by the size and spatial resolution of the recording device. Zhang et al. [22] have reported very high resolution measurements of turbulent flow in a square duct using film-based HPIV, where particles were seeded to a reported density of 1-8 particles/mm^3 in a volume measuring 46.6 x 46.6 x 42.25 mm^3. Although film has much better resolution and is larger than digital recording sensors, Meng et al. [21] extensively cite the difficulties of film based holographic PIV, which have likely prevented the method from being widely utilized. In contrast, digital Holographic PIV is more readily usable, but is often limited to small volumes and low seeding density [21]. A digital hybrid HPIV method has been implemented by Meng et al. [21] which allows for measurement in volumes with larger depth, but the size of the in-plane dimensions are limited by the physical size of the digital sensor, and seeding density remains low. Sheng et al. [23] have presented recent results of measurements in a turbulent boundary layer with

increased seeding density (0.014 ppp) in a volume measuring 1.5 x 2.5 x 1.5 mm^3.

The synthetic aperture PIV technique is implemented using an array of synchronized CCD cameras distributed such that the fields of view overlap. Images are recombined in software using a refocusing algorithm, commonly applied in synthetic aperture applications [24, 25]. The result is sharply focused particles in the plane of interest (high intensity), whereas particles out-of-plane appear blurred (low intensity). Due to the multiple camera viewpoints and the effective reduction of signal strength of out-of-plane particles in image recombination, particles that would otherwise be occluded can in fact be seen. The 3D intensity field of particle-laden flows can be reconstructed by refocusing throughout the entire volume and thresholding out particles with lower intensities. Typical 3D PIV techniques can then be applied to the intensity fields to extract velocity data. This technique enables larger volumes to be resolved with greater seeding density, yielding higher spatial resolution than prior 3D PIV methods. Additionally, the algorithms are simple and robust and build on established image processing techniques. Results of simulated particle fields show the ability to reconstruct 3D volumes with seeding densities of 0.17 ppp (6.68 particles/mm^3) when the ratio of X-Y to Z dimension is 5:1 (50 x 50 x 10 mm^3 volume), and 0.05 ppp (1.08 particles/mm^3) when the ratio of X-Y to Z dimension is 4:3 (40 x 40 x 30 mm^3 volume). A vortex ring flow field is imposed on each of these simulated volumes, and 3D PIV analysis yields highly-resolved vector fields.

Results are presented from an experimental implementation of SAPIV using a custom built camera array to study a vortex ring in a 65 x 40 x 32 mm^3 volume. Design considerations for experimental 3D SAPIV implementation are discussed throughout the paper. The experimental data presented are benchmarked with 2D PIV, and demonstrate the ability of SAPIV to resolve 3D flow fields, providing a useful and flexible tool for making 3DPIV measurements.

2.2 Synthetic Aperture Methodology

2.2.1 Light Field Imaging

Synthetic aperture PIV is based on the concept of *light field imaging*, which involves sampling a large number of light rays from a scene to allow for scene reparameterization (Isaksen et al. [24]). In practice, one method used by researchers in the imaging community for sampling a large number of rays is to use a camera array [26, 24, 27, 25]. The novelty of the approach presented herein is the application of the reparameterization methods to 3D PIV, and the development of the technique into a measurement system, including the generation of algorithms to reconstruct 3D particle intensity fields from the refocused images. The technique is broken down into sequential components in the list below.

1. **Image Acquisition**: Image capture is performed using an array of cameras typically arranged in a multi-baseline stereo configuration, which view the scene from different viewpoints. The cameras can be placed at arbitrary locations and angles as long as the desired refocused planes are in the field of view (FOV) of each camera. The depth of field of each camera is large enough such that the entire volume of interest is in focus. The multiple viewpoints array captures many more light rays than can be seen with one camera (i.e. *light field imaging* [24]).

2. **Synthetic Aperture Refocusing**: The light fields are reparameterized using synthetic aperture refocusing [24, 27, 25].

3. **3D Intensity Field Reconstruction**: 3D intensity fields generated by a fluid seeded with flow tracers are extracted from images refocused using the

synthetic aperture refocusing method.

4. **3D Cross-Correlation**: Typical 3D intensity field cross-correlation methods similar to those used in Tomographic-PIV [15, 16] operate on the reconstructed 3D intensity fields to extract velocity fields.

2.2.2 Synthetic Aperture Refocusing

In synthetic aperture techniques, images captured by an array of cameras each with large depths of field are post-processed to generate one image with a narrow depth of field on a specific focal plane [24, 26]. Through synthetic aperture refocusing, further post-processing allows the location of the focal plane to be arbitrarily placed within the imaged volume [24, 27, 25]. This technique provides many desirable implications for 3D PIV; namely, a particle-laden volume can be captured at one instant in time and synthetic aperture refocusing allows for the reconstruction of particles at known depths throughout the volume post capture.

In general, the post-processing for synthetic aperture refocusing involves projecting all images onto a focal surface (planar or otherwise) in the scene on which the geometry is known, averaging the projected images to generate one image, and repeating for an arbitrary number of focal planes [24]. The working principle of the synthetic aperture technique is demonstrated with a simplified example. Consider the case where two cameras at different X locations view the same portion of the reference plane in figure 2-1. If the images from each camera are mapped to the reference plane, and then all images (now in reference plane coordinates) are averaged, the point A will be in sharp focus in the averaged image. However, points not on this plane (i.e., at different depths) will appear out of focus because of the parallax between cameras. In the average of the reference plane aligned images, the images

of point B from the two cameras will appear at the points where the gray rays in figure 2-1 intersect the reference plane, which are separated by Δ. By adding more cameras, mapping images to the reference plane and averaging, the signal of point A will grow increasingly larger than the "noise" of the off-plane points. Note that this "noise" does not refer to actual image noise, but the signal from particles not on the plane of interest. The concept is shown schematically in 3D in figure 2-2(d), which shows image capture by a 9 camera array, and subsequent refocusing on two planes. By positioning the cameras on a sufficiently large baseline (larger separation between camera centers of projection (COP's)), some of the cameras can see particles which are occluded in other images [24, 26, 27, 25]. Therefore, the partially occluded particles retain a high signal in the refocused image.

2.2.3 Three-Dimensional Volume Reconstruction

The goal of the synthetic aperture PIV technique is to reconstruct 3D particle intensity fields which are suitable for cross-correlation based 3D PIV processing. The starting point for volume reconstruction is the implementation of the synthetic aperture algorithm to generate refocused images on planes throughout the volume. Thereafter, the actual particle field must be extracted from the refocused images and organized into a volume with quantifiable locations.

To implement synthetic aperture refocusing, relationships between the image coordinates and focal planes in world coordinates must be established (i.e., a set of mapping functions). In the simulations presented herein, cameras are represented with a pinhole model, and the mapping functions can be generated by an algorithm presented in [25], which we refer to as the *map-shift-average algorithm.* This algo-

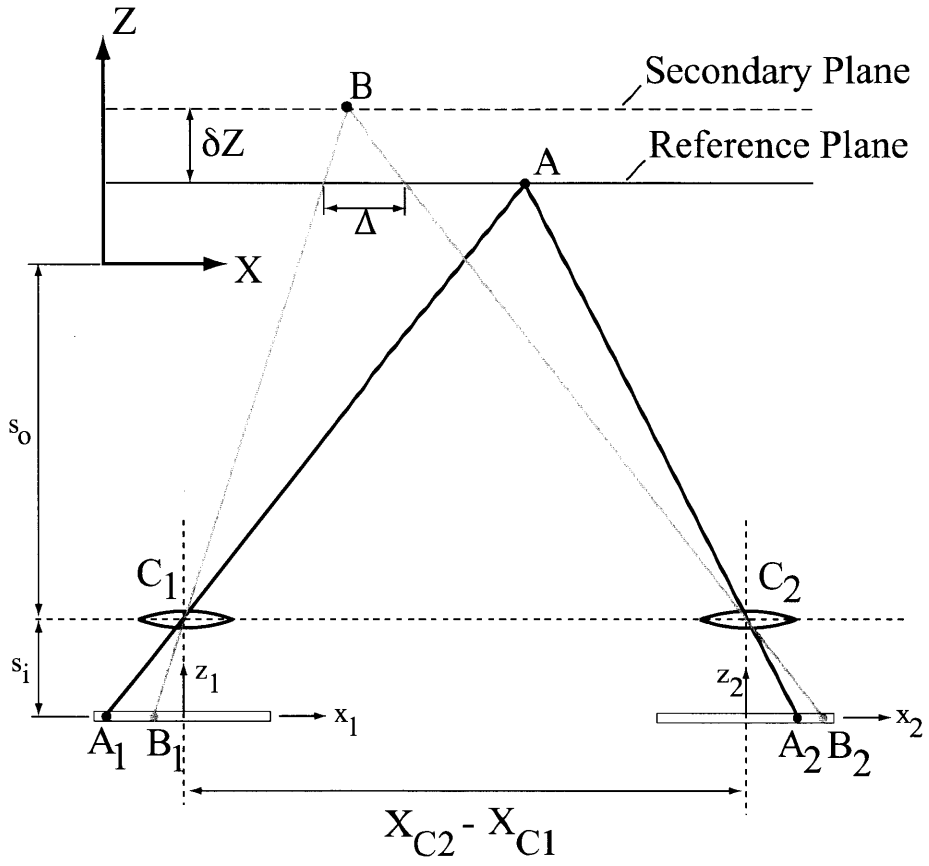


Figure 2-1: Schematic showing the optical arrangement and the concept of parallax. X-Z is the global coordinate system and x-z are local image coordinates.

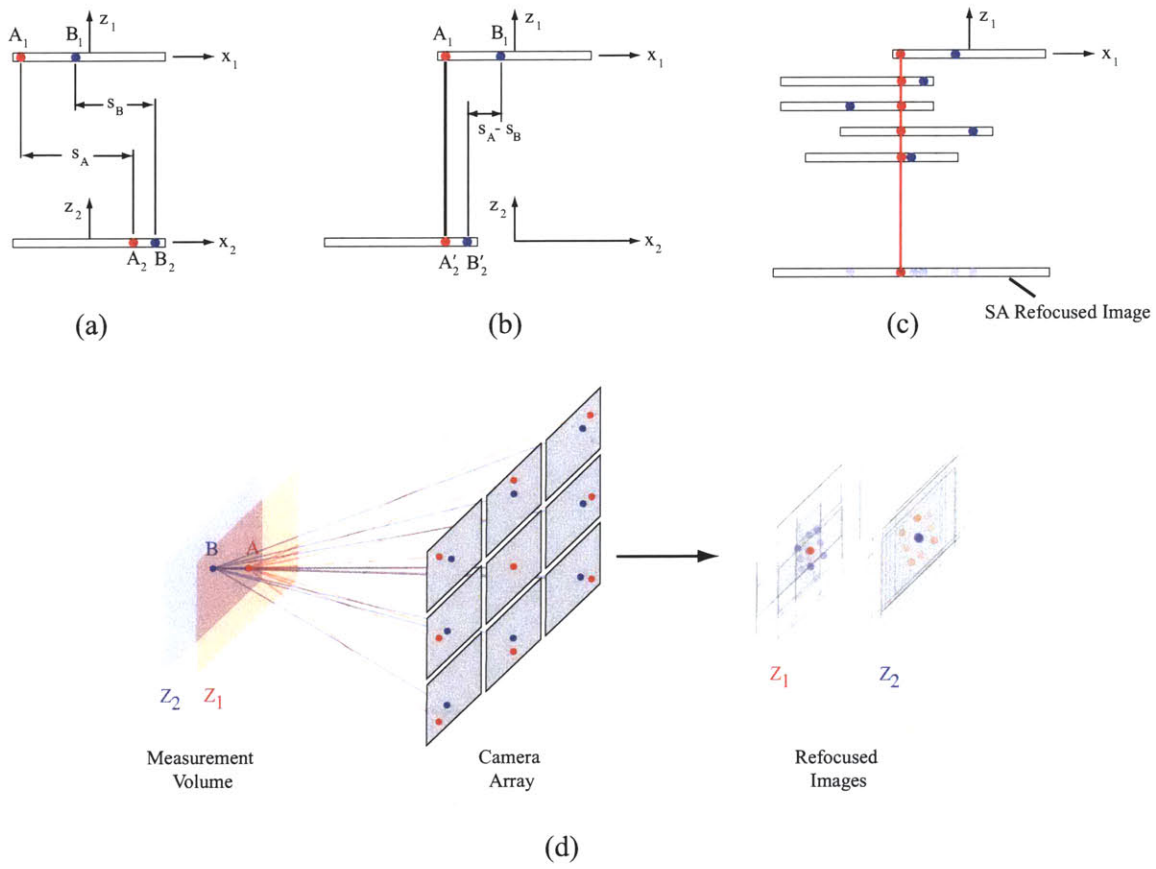


Figure 2-2: Schematic demonstrating the synthetic aperture refocusing method. In (a) the z axes of the image planes are aligned and in (b) image plane 2 has been shifted to align the images of point A. In (c) multiple images are shifted to align on the plane containing point A, and are averaged to generate a refocused image. The schematic in (d) shows a 3D depiction of image capture by a 9 camera array, and subsequent refocusing on two planes.

rithm is suitable for pinhole cameras with no image distortion and no changes in optical media in the line of sight (e.g. air-glass-water transition). In practice, mapping functions that can account for distortion and changes in optical media can be generated by more sophisticated calibration techniques, as is done for the experiment discussed in Section 2.3. For example, as employed by [17], a planar calibration grid can be placed at several planes throughout the volume to generate mapping functions, and error in these functions is reduced using a volume self-calibration technique [28]. The map-shift-average algorithm represents the most basic that is used in synthetic aperture refocusing, but it is helpful in describing the technique and is also implemented in the simulations presented in Section 2.2.4.

Map-Shift-Average Algorithm

The first step in the map-shift-average algorithm is to align all images on a reference plane [25]. The reference plane is an actual plane in the view of the cameras which, in practice, is defined using a planar calibration grid. Images are aligned on the reference plane by applying a *homography* which, as described in [29], is a central projection mapping between two planes given by

$$\begin{pmatrix} bx' \\ by' \\ b \end{pmatrix}_i = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}_i \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}_i \tag{2.1}$$

where b is a constant, $h_{33} = 1$, and the vectors $\mathbf{x}'$ and $\mathbf{x}$ are the homogeneous coordinates of points in the reference plane and the image plane, respectively. In this way the raw image from the *ith* camera, I_i, is mapped to a new image in reference plane coordinates, I_{RP_i}. Hartley & Zisserman [29] describe robust algorithms for

estimating the homography based on the images of several points on the reference plane.

Light field imaging with camera arrays offers the ability to refocus on arbitrarily shaped focal surfaces [24, 27, 25]. For the simulations herein, we restrict the situation to refocusing on fronto-parallel planes where the raw images are captured with a camera array where all camera COP's lie on the same plane. For this restricted case, the mapping that must be applied to refocus on each fronto-parallel plane is simply a shift of the reference-plane aligned images by an amount proportional to the relative camera locations [27]. The coordinates of image points on the new focal plane are given by

$$\begin{pmatrix} x'' \\ y'' \\ 1 \end{pmatrix}_i = \begin{bmatrix} 1 & 0 & \mu_k \Delta X_{C_i} \\ 0 & 1 & \mu_k \Delta Y_{C_i} \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix}_i \tag{2.2}$$

where μ_k is a constant that determines the location of the kth focal plane of the refocused image, and ΔX_{C_i} and ΔY_{C_i} are the relative camera locations of the ith camera. This portion of the algorithm transforms the ith reference plane aligned image, I_{RP_i} to an image on the kth focal plane, $I_{FP_{ki}}$. The relative camera locations can be found using a simple calibration method presented in Vaish et al. [27] which operates on the images from the camera array of points on planes at several depths. No *a priori* knowledge of the camera locations or optical configurations are required to implement this algorithm.

In the final step in the refocusing algorithm, the mapped and shifted images (now all aligned on the kth focal plane) are averaged. The resultant image is referred to

as the refocused image, and is given by

$$I_{SA_k} = \frac{1}{N} \sum_{i=1}^{N} I_{FP_{ki}} \tag{2.3}$$

where images are combined from N cameras [25]. Particles that lie on the plane of the refocused image appear sharp and have high intensity, while particles not on this plane are blurred and have lower intensity. The only difference between the methodology described here and the practical realization comes in the mapping functions used to align all images on the focal planes. Rather than using the linear homography and homology, a higher order mapping will likely be required to deal with distortions and changes in optical media [17, 28].

Focal Plane Spacing and Viewable Depth

Theoretical analysis of the focal plane spacing and viewable depth is presented using the two camera model shown in figures 2-1 & 2-2. Here, we define the focal plane spacing, δZ, as the distance between refocused planes throughout the volume, where Z is the distance from the front of the experimental volume. The viewable depth is the total depth dimension (Z_o) of the reconstructed volume. The simple two camera model is sufficiently general to establish the theoretical limits of the system, and the results can be applied to design and predict the performance of various arrangements. A full scale simulated model is implemented in Section 2.2.4 to examine the effects of parameters not considered in this theoretical treatment, such as particle seeding density, number of cameras and mapping error.

In the model of the imaging system shown in figure 2-1, the image sensors are parallel to each other and the camera centers of projection lie on the same plane.

The geometric imaging optics are described by four parameters: the focal length (f), distance from the lens to the front of the imaged volume (s_o), distance from lens to image plane (s_i) and magnification ($M(Z) = -s_i/(s_o + Z)$). Also, the focal length is related to s_o and s_i through the imaging condition: $1/f = 1/s_i + 1/s_o$.

To examine the effect of camera layout on focal plane spacing, we start by considering the relationship between points in physical space and the images of these points. The coordinates of a general point, A (see figure 2-1), projected onto the imaging plane of one of the cameras is given by

$$x_A = \left[\frac{-s_i}{s_o + Z_A}\right](X_A - X_C) - d_C \tag{2.4}$$

where X_C is the X coordinate of the camera center of projection in global coordinates, and d_C is the displacement of the image sensor from the center of projection (zero as shown). We define the value of "initial" magnification as $M(Z = 0)$. The dimension of each pixel projected into physical space is given by

$$\delta X = \frac{-p}{M(Z)} \tag{2.5}$$

where p is the pixel pitch (width of a pixel).

As described earlier, the first step in the reconstruction is to align the images on a reference plane. In the model in figure 2-1, the reference plane is chosen (arbitrarily) to be the plane in physical space on which point A lies. In figure 2-2, the image sensors are shown with the centers aligned along the z-axes of the local image coordinates. To align the images on the reference plane would require a shift of the image sensor (or equivalently, the digital image) of camera 2 by an amount equal to $s_A = x_{A_2} - x_{A_1}$ (the disparity of the images of point A) in the negative $x-$direction of the camera 2

coordinate system. Since point B does not lie on the reference plane, $s_B = x_{B_2} - x_{B_1}$ does not equal s_A, and thus points at different depths are disambiguated.

The focal plane spacing, δZ, is dictated by the image plane shift required to refocus on another plane. To move from the reference plane to the secondary plane shown in the model, the image plane must be shifted by an amount equal to $s_A - s_B$, which is given by

$$s_A - s_B = \Delta X_C \left[\frac{-s_i}{s_o + Z_B} - \frac{-s_i}{s_o + Z_A} \right] \tag{2.6}$$

where $\Delta X_C = X_{C2} - X_{C1}$ is the separation between the camera COP's. The minimum amount that the image sensor can be shifted by is the width of one pixel (assuming no spatial interpolation), therefore $s_A - s_B >= p$. Letting $Z_A = Z$ and $Z_B = Z + \delta Z$ and imposing the minimum shift requirement yields

$$\Delta X_C \left[\frac{-s_i}{(s_o + Z) + \delta Z} - M(Z) \right] = p \tag{2.7}$$

Solving Equation 2.7 for δZ gives

$$\delta Z = -s_i \left[\frac{\Delta X_C}{p + \Delta X_C \cdot M(Z)} - \frac{1}{M(Z)} \right] \tag{2.8}$$

Dividing the top and bottom of the first term in the bracket by $1/s_o$ yields

$$\delta Z = -s_i \left[\frac{\left(\frac{\Delta X_C}{s_o}\right)}{\frac{p}{s_o} + \left(\frac{\Delta X_C}{s_o}\right) \cdot M(Z)} - \frac{1}{M(Z)} \right] \tag{2.9}$$

Equation 2.9 contains the convenient geometric parameter, $\frac{\Delta X_C}{s_o}$, which is the ratio of the camera spacing to the distance from the front of the imaged volume. For convenience, we will let $D \equiv \frac{\Delta X_C}{s_o}$. This parameter characterizes the baseline of the

system. Equation 2.9 can be further simplified by applying Equation 2.5 at Z to replace p and rearranging,

$$\frac{\delta Z}{\delta X} = \frac{\frac{1}{D} + \frac{Z}{Ds_o}}{1 - \frac{\delta X}{Ds_o}} \tag{2.10}$$

For typical PIV applications, it is reasonable to assume $\delta X \ll Ds_o$; applying this approximation yields

$$\frac{\delta Z}{\delta X} \cong \frac{1}{D} + \frac{Z}{Ds_o} \tag{2.11}$$

Therefore, the ratio $\delta Z/\delta X$ is a linear function of Z with intercept defined by the camera baseline and slope defined by the camera baseline and the imaging system optics. The depth of field in the reconstructed volume is directly related to $\delta Z/\delta X$. When $\delta Z/\delta X$ is small, the camera lines of sight are at large angles to one another, and the physical depth over which particles are in focus is small (small depth of field). Conversely, larger $\delta Z/\delta X$ leads to a larger depth of field, which is manifested as reconstructed particles which are elongated in Z.

In theory, the overall viewable range in X, Y and Z is limited only by the field of view (FOV) and depth of field (DOF) of a single camera of the array. In reality, images from the outer cameras of the array must be shifted with respect to the central camera image to refocus at different depths. The outer edges of the refocused images have a lower signal-to-noise ratio than regions where all images contribute to the average. This effective limitation on the FOV can be characterized by the number of image shifts required to refocus through an entire volume, which is given by

$$N = \frac{Z_o}{\delta Z} \tag{2.12}$$

where Z_o is the depth of the volume. Since images are shifted in integer pixel increments (assuming no interpolation), it is possible to calculate the region of each refocused image to which all images will contribute. As will be seen in Section 2.2.4, this effect does degrade the performance near the outer edges of the larger reconstructed volumes. This effect could be mitigated by excluding the portions of the shifted images which don't contribute from the average, but this adjustment to the refocusing algorithm has not been included here. Nonetheless, the implications of the technique are that the observable range of the system is highly scalable, with the ability to trade-off depth of field for viewable range much as one would trade-off X-Y resolution for FOV with a single camera.

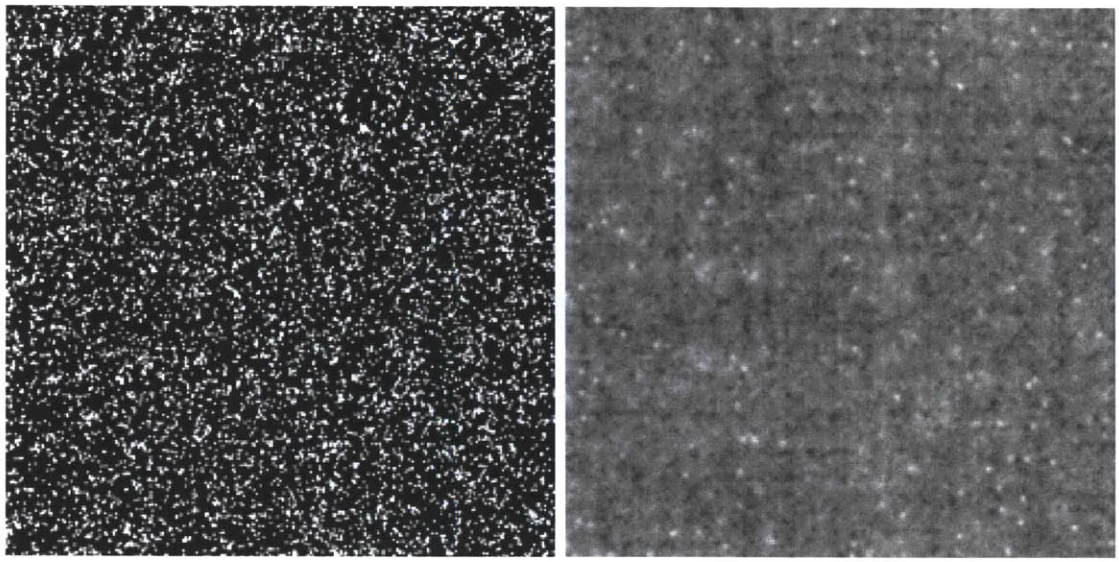
3D Particle Intensity Field Extraction

Once the refocused images have been generated, the images must be processed to extract the actual particles from the blurred, lower intensity background. Figure 2-3(a) shows a zoomed view (250 x 250 pixels) of simulated image from the central camera of the array and figure 2-3(b) shows a refocused image from the 50 x 50 x 10 mm^3 volume simulation which will be described in detail in Section 2.2.4. The refocused image has a higher background "noise" level than the raw image or a typical 2D PIV image due to the averaging of multiple images, however the "noise" is probabilistic. If we consider the intensity fields of the N images aligned on a given focal plane to be independent and identically distributed random variables, $I_1, I_2, ..., I_N$, with means μ and standard deviations σ, then the central limit theorem states that the distribution of the average of the random variables will be Gaussian with mean μ and standard deviation of $\sigma/\sqrt{N}$. Therefore, the intensity distribution of a refocused image (which is an average of the focal plane aligned images) can be

modeled as Gaussian. Figure 2-4(a) shows the intensity histogram for a reference plane-aligned image from one camera of the array and the histogram for the refocused image in figure 2-3(b) is shown in figure 2-4(b). Clearly, the distribution of intensity for the single camera image is not Gaussian, but the shape of the distribution of the refocused image follows a Gaussian distribution quite well, as indicated by the model fit. As more particles are added, the mean of the individual images becomes larger, and thus the mean of the refocused image becomes larger. Actual particles appear with high intensity values, and are thus outliers with respect to the distribution of the refocused image. Intensity thresholding can be applied to retain actual particles and eliminate background "noise" from the images. It was found that a threshold value around three standard deviations above the mean intensity of each refocused image yielded acceptable reconstruction. Figure 2-5 shows the refocused image from figure 2-3(b) now thresholded to reveal the true particles. By refocusing throughout the volume and thresholding the refocused images, the three-dimensional intensity field is reconstructed. Although the thresholding method may require more optimization, it appears that by detecting the outliers in the refocused images the true particle field can be reconstructed, attesting to the simplicity of the SAPIV technique.

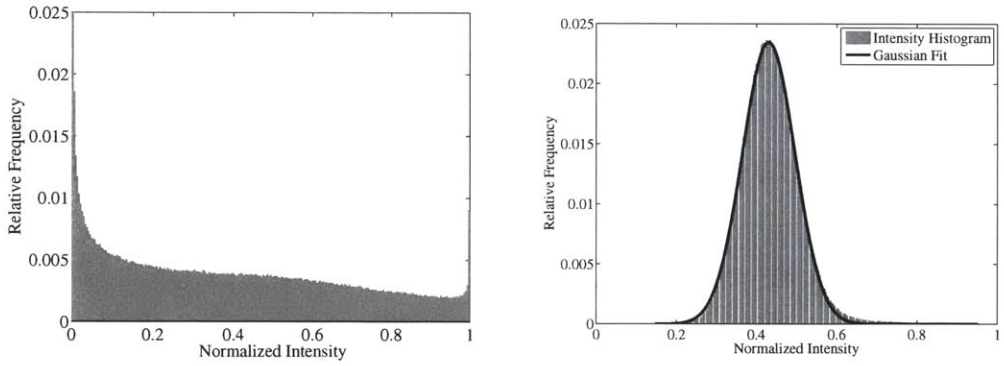
2.2.4 Simulated Camera Array

A 5x5 camera array model is simulated to investigate the system performance as a function of particle seeding density, size of measurement volume and error in the mapping function. The effect of array layout and camera number on reconstruction performance is also investigated by changing the spacing between cameras and removing certain cameras from the array, respectively. Cameras are arranged with COP's on the same plane and equal spacing along X and Y between all camera



(a) (b)

Figure 2-3: Zoomed views (250 x 250 pixels) of a (a) simulated image from the central camera of the array and (b) refocused image using all of the simulated images of the array.



(a) (b)

Figure 2-4: Intensity histograms for a single simulated image aligned on a given focal plane (a) and for a refocused image on that focal plane (b).

Figure 2-5: Zoomed view (250 x 250 pixels) of the thresholded refocused image from figure 2-3(b) revealing the particles and removing the background noise.

COP's (unless otherwise noted). In order to overlap the fields of view, the cameras are angled such that the ray passing through the center of the measurement volume intersects the center of the image sensor. For this arrangement, the map-shift-average algorithm applies. The perspective due to the angling of the cameras is compensated for when the reference plane homographies are applied to the images.

The ability of the system to resolve the particle field in four different measurement volume sizes is examined. The volume sizes (X x Y x Z) are 50 x 50 x 10 mm^3, 40 x 40 x 30 mm^3, 50 x 50 x 50 mm^3 and 100 x 100 x 100 mm^3. For each volume, the system performance is evaluated for several different camera baselines, D, and for each camera baseline the particle seeding density is varied. For all simulations presented herein, the cameras are modeled with 85 mm focal length lenses and the imaging sensors are 1000 pixels x 1000 pixels with a pixel pitch of 10μm. For the

100 x 100 x 100 mm^3 volume, the initial magnification is set to $M(Z{=}0)$ = -0.1 and for the other three volumes the magnification is $M(Z{=}0)$ = -0.2.

Reference plane homographies are calculated for each camera from calibration images of known points on several Z planes with the central camera of the array as the reference camera. The camera positions relative to the central camera are established using the calibration method described in [27]. The calibration images are used to establish the shift required to refocus at each depth in order to define the conversion between voxels and physical units in the Z dimension. Herein, we define a voxel as having the dimensions of a pixel in X and Y, and dimension equal to the focal plane spacing in Z. Because integer pixel shifts are used in the map-shift-average algorithm, a given calibration depth may not correspond exactly to any of the refocused images. Therefore, the actual voxel to Z calibration is approximated by fitting a gaussian curve to the summed intensity from refocused images surrounding each calibration plane and finding the voxel corresponding to the peak of the fit.

Particles are randomly seeded within the volume and imaged using the camera array. Once the image plane coordinates of a point are known, a realistic model of the intensity distribution must be applied. A standard method of simulated image generation is to apply a Gaussian intensity profile as described in [3]. The distribution is applied to each camera image for each pixel location, which forms an image similar to the one presented in figure 2-3(a).

After simulated images have been formed by each camera, the map-shift-average algorithm, followed by intensity thresholding and three-dimensional field reconstruction, is carried out for each numerical experiment. In order to quantify how well the intensity field is reconstructed, the same measure of reconstruction quality, Q, used in Elsinga et al. [15, 16] is applied here:

$$Q = \frac{\sum_{X,Y,Z} E_r(X,Y,Z) \cdot E_s(X,Y,Z)}{\sqrt{\sum_{X,Y,Z} {E_r}^2(X,Y,Z) \cdot \sum_{X,Y,Z} {E_s}^2(X,Y,Z)}} \tag{2.13}$$

where E_r is the reconstructed intensity field and E_s is a synthesized intensity volume based on the known particle locations.

Baseline spacing affects the Z dimension of the voxels such that they represent larger physical sizes than the $X-Y$ dimensions, therefore the intensity distribution in the synthesized field is scaled in Z in voxel space such that in physical space the intensity distribution is spherically symmetric. This ensures that a perfectly reconstructed particle would yield a Q value of 1 when compared to the synthesized field.

The value of Q is calculated for each numerical experiment conducted, and we use the same requirement as in [15] of $Q \geq 0.75$ for the reconstruction to be considered adequate. In all cases other than the 40 x 40 x 30 mm^3 measurement volume, the outer 50 edge pixels were cropped prior to calculating Q because of the effective loss in FOV as described earlier (in the 40 x 40 x 30 mm^3, the images do not fill the entire image sensor and thus the outer pixels of the reconstructed images contain no particles). Figures 2-6(a)-(c) present the reconstruction quality as a function of particle seeding density for various camera baselines in each volume. The number of seeded particles, maximum particle seeding density (C) and number of particles per pixel (ppp) in each case are summarized in Table 2.1. To find the maximum seeding density, simple linear interpolation of the data was used to find the seeding density corresponding to Q=0.75.

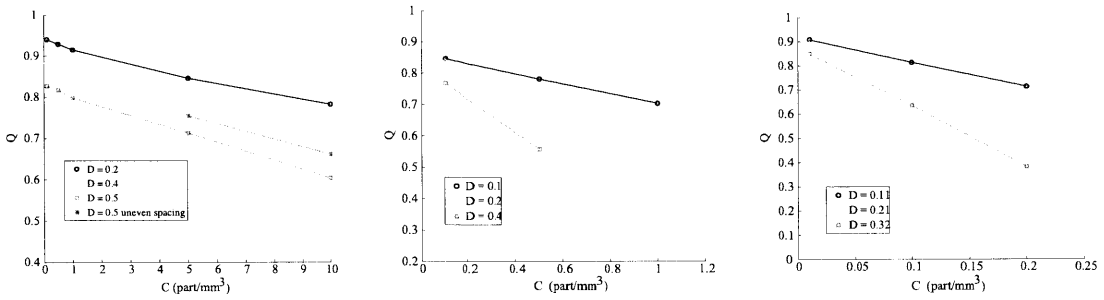
In the case of the 50 x 50 x 10 mm^3 measurement volume, the reconstruction quality falls off with increasing seeding density, and also decreases with increasing camera baselines. For larger seeding density, the reduction in reconstruction quality

is expected; the occurrence of off-plane particle overlap is increased, and the overall signal-to-noise ratio decreases in the refocused images. The reason for reduced reconstruction quality with increasing baseline is less obvious. Investigation of the data reveals that the reason for the degradation in reconstruction quality with increasing baseline is due to the more extreme warping of the particle images imposed by the homography which maps images to the reference plane. The particles become elongated in the mapped image which raises the background noise in the refocused images. This may be mitigated by placing the outer cameras of the array a normalized distance $D/2$ from the inner cameras (which determine the focal plane spacing of the system and would still be placed a normalized distance D from the central camera) but requires interpolation when shifting images from the outer cameras of the array. This has been implemented in the case labeled $D = 0.5$ uneven spacing, and indeed the reconstruction quality is improved for the same seeding density. For this configuration, the achievable seeding density is $C = 5.24$ part./mm^3 and the resultant particles per pixel is 0.13.

Simulations in the 50 x 50 x 50 mm^3 volume show a similar trend as for the 50 x 50 x 10 mm^3 volume, with Q decreasing with increasing C more rapidly for a larger camera baseline. However, the achievable seeding density is lower than for the volume with smaller depth (e.g. $C = 0.37$ part./mm^3 for $D = 0.2$). The reasons for the lower seeding density and the lower actual number of particles that can be seeded as compared to the 50 x 50 x 10 mm^3 volume are four-fold. First, the larger depth of field of each camera requires a larger f-number which increases the particle image diameter; this results in a larger mean intensity in the refocused image. Second, the depth of the volume creates more likelihood of particle overlap on each simulated image, which can, in a sense, decrease the dynamic range of the system by increasing the likelihood of saturated pixels. Third, the larger depth creates a higher likelihood

of overlapping of many different out-of-focus particles in the refocused images. These false particles may be retained in the thresholding if enough images overlap. Finally, the limitation on the field of view imposed by the image shift contributes to the loss in reconstruction quality because some images contribute zero values to the average toward the outer regions of the refocused images, thus true particles have a lower intensity value. This can be mitigated by averaging only the portions of the image which are known to contribute to the refocused image, but that technique has not been implemented here. Tuning the camera array and reconstruction algorithms to enable more seeding in the very large volumes is the subject of ongoing work. By decreasing the dimensions of the volume somewhat, more particles can be seeded, even in volumes where the Z dimension approaches that of the $X - Y$ dimensions, and all are relatively large. Simulations in the 40 x 40 x 30 mm^3 volume are carried out at only one array configuration - $D = 0.4$ uneven spacing - and it was found that a seeding density of $C = 1.08$ part./mm^3 corresponding to 0.05 particles per pixel could be achieved. As shown section 2.2.5, this seeding density allows 3D PIV measurements to be made in this volume with reasonable spatial resolution.

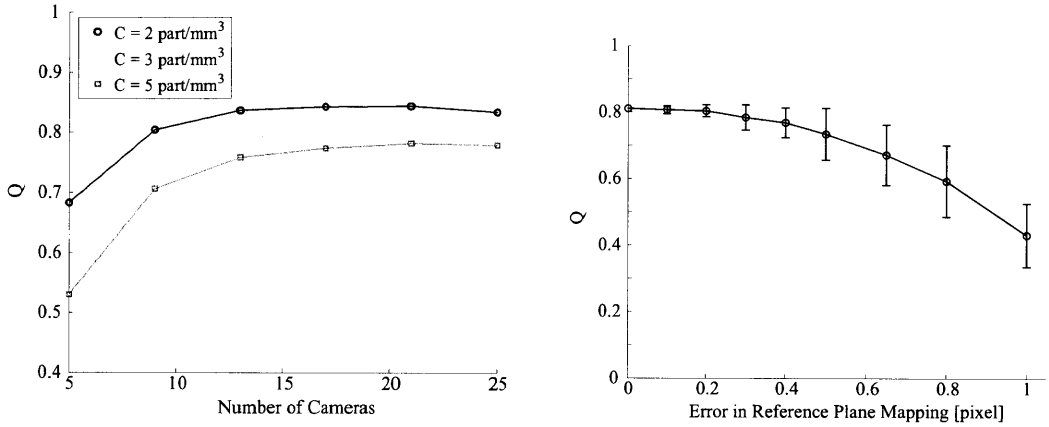
Finally, the camera magnifications are reduced to accommodate the large 100 x 100 x 100 mm^3 measurement volume. The trend for reconstruction quality as a function of seeding density and baseline is similar to that observed for the other volumes studied. The total number of particles that can be seeded is, however, larger than for the 50 x 50 x 50 mm^3 volume for comparable camera baselines. Thus, trading off X-Y resolution allows for more particles to be seeded even with increasing depth dimension. Overall, these results indicate that the synthetic aperture PIV technique is capable of imaging extremely densely seeded volumes where the depth dimension is somewhat reduced, and still quite densely seeded volumes when the Z dimension approaches that of the X-Y dimensions.



(a) (b) (c)

Figure 2-6: Effect of particle seeding density (C, $particles/(mm)^3$) on reconstruction quality, Q, for various camera baselines in a 50 x 50 x 10 mm^3 volume (a), a 50 x 50 x 50 mm^3 volume (b) and in a 100 x 100 x 100 mm^3 volume (c).

The effect of camera number on reconstruction quality is investigated in the 50 x 50 x 10 mm^3 volume by using only some of the cameras in the array. Figure 2-7(a) shows Q as a function of camera number with seeding densities of 2,3 and 5 part./mm^3. Clearly, the reconstruction quality reaches a point of diminishing returns as more cameras are added, and the most efficient number of cameras seems to be in the range of 10-15. Finally, to determine an upper limit to the allowable error in calibration, simulations in the 50 x 50 x 10 mm^3 volume with 13 cameras and 3 part./mm^3 are carried out imposing error in the reference plane homography. This results in the misalignment of mapped images on each focal plane. The total misalignment error is defined as $\epsilon_{RP} = \sqrt{\epsilon_x^2 + \epsilon_y^2}$ and is the same for all cameras, but ϵ_x and ϵ_y are randomized for each camera so as not to introduce any bias. For each value of ϵ_{RP}, ten simulations are carried out such that different random values of ϵ_x and ϵ_y are applied to each mapping for each simulation; the mean value of Q is then found for each ϵ_{RP}. Figure 2-7(b) shows the mean value of Q as a function of ϵ_{RP}; the error bars represent three standard deviations from the mean. The value of



(a) (b)

Figure 2-7: Reconstruction quality, Q, as a function of camera number for various particle seeding densities (a) and error in reference plane mapping (b) in the 50 x 50 x 10 mm^3 volume.

ϵ_{RP} corresponding to a mean value of Q = 0.75 was found to be 0.45 pixels. Volume self-calibration techniques such as that described in [28] are capable of reducing mapping errors to less than 0.1 pixels; therefore, we expect the synthetic aperture PIV technique to be robust to the error levels that will exist in mapping functions applied in actual experiments.

2.2.5 Synthetic 3D Flow Fields

Two synthetic flow fields are simulated to assess the ability of the synthetic aperture PIV method in reconstructing 3D intensity fields that are suitable for cross-correlation based 3D PIV analysis. In each simulation, the fluid motion is prescribed by the same equation for a vortex ring as used in Elsinga et al. [15, 16], where the

Table 2.1: Summary of resolution and particle seeding density for simulated measurement volumes.

Measurement Volume (mm^3)	D	$\delta X(mm)$	$\delta Z/\delta X$	C (part./mm^3)	*part./pixel (ppp)*	*# part.*
50 x 50 x 10	0.2	0.05	5.00	>10	>0.25	>250 000
	0.4	0.05	2.50	6.68	0.17	166 960
	0.5	0.05	2.00	3.26	0.08	81 537
	0.5 (uneven)	0.05	2.00	5.24	0.13	131 040
40 x 40 x 30	0.4 (uneven)	0.05	2.51	1.08	0.05	51 988
50 x 50 x 50	0.1	0.05	10.24	0.70	0.09	87 371
	0.2	0.05	5.13	0.37	0.05	46 229
	0.4	0.05	2.56	0.14	0.02	17 159
100 x 100 x 100	0.11	0.1	9.40	0.16	0.16	164 240
	0.21	0.1	4.70	0.10	0.10	95 028
	0.32	0.1	3.13	0.05	0.05	51 659

velocity magnitude is given by

$$|V| = \frac{8KR}{l} e^{-\frac{R}{l}} \tag{2.14}$$

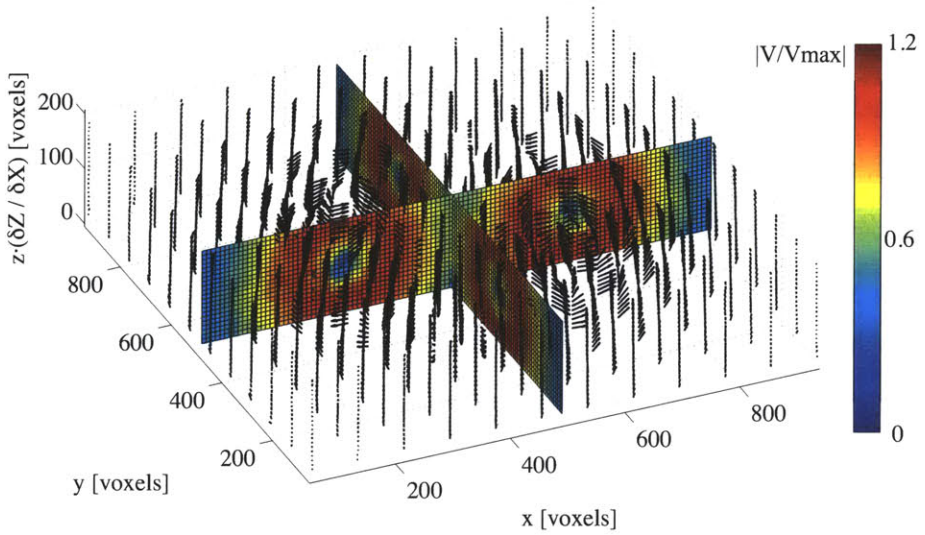
where R is the radial distance of a point from the toroidal axis of the vortex ring and l is a length scale determining the size of the vortex ring (chosen to be 4 mm). The constant, K, is a conversion factor from voxel to physical units and is required since, in the present case, the synthetic particles are seeded in physical space and imaged by the model camera array. The toroidal axis forms a circle of diameter 20 mm. The first synthetic experiment is carried out in a 50 x 50 x 10 mm^3 volume where the central axis of the toroid is parallel to the Z-axis. The volume is seeded with 125,000 particles (C = 5 part/mm^3), and images are simulated in a 21 camera array with spacing D = 0.5 (uneven spacing, see Table 2.1) and magnification $M(Z{=}0)$ = -0.2. This results in a reconstruction quality of Q = 0.76. The maximum displacement in

this flow field is 0.3 mm which corresponds to 6 voxels in X and Y and 2.9 voxels in Z.

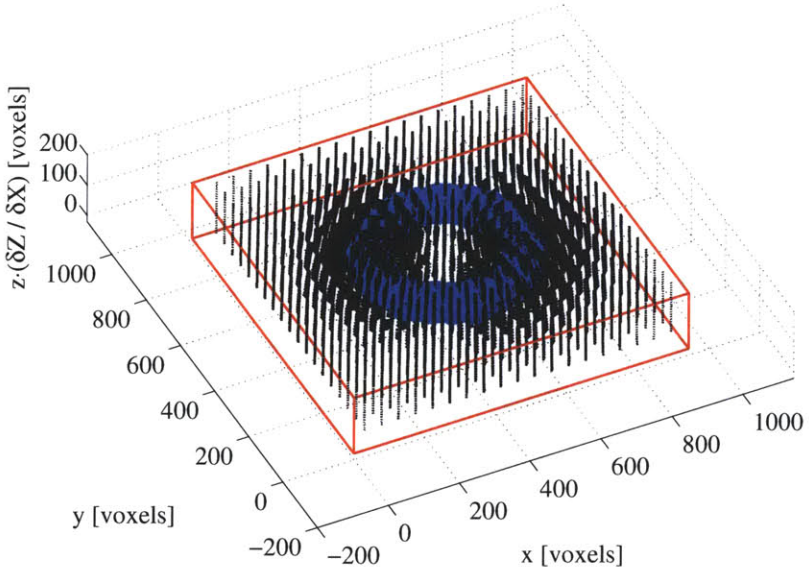
Since the reconstructed volumes are intensity fields, a cross-correlation based PIV calculation is suitable for calculating vector fields. In the present study, we have adapted an open-source 2D PIV code, matPIV [30], for 3D functionality. A multipass algorithm with a final interrogation volume containing 32 x 32 x 16 voxels and 75% overlap generates 327448 vectors (122 x 122 x 22 vectors). The Z dimension of the interrogation volumes in voxel units is half that of the X-Y dimension because the focal plane spacing is twice the pixel size for this camera configuration. Each interrogation volume contains approximately 20 particles, based on the gross particle seeding density.

Figure 2-8(a) shows the resultant velocity field with every tenth vector plotted in X and Y and every vector plotted in Z. The Z voxel and velocity values are multiplied by $\frac{\delta Z}{\delta X}$ to create the correct aspect ratio for plotting purposes. The fields are plotted in voxel units; if converted to physical units, the data set would not be cubical. Two slices are shown with normalized velocity magnitude contours revealing the vortex ring structure and symmetry. The maximum velocity magnitude in the exact known field is used to normalize the velocity magnitude in the processed field. Figure 2-8(b) shows the vector field and a vorticity iso-surface (0.15 voxels/voxel) with every sixth vector plotted in X and Y and every vector plotted in Z.

To quantitatively evaluate the performance, both the reconstructed 3D intensity fields and the synthesized 3D intensity fields are processed using the 3D adaptation of matPIV, and each is compared to the exact velocity field. The error is defined as the difference between the processed and exact field at every vector location. By comparing the PIV results for both fields, error due to the PIV algorithm itself can be identified. Both the synthesized and reconstructed volumes are processed using



(a)



(b)

Figure 2-8: Three-dimensional vector field resulting from PIV processing of the reconstructed intensity volumes using SAPIV in the 50 x 50 x 10 mm^3 volume. Figure 2-8(a) shows two cuts with normalized velocity magnitude contours and figure 2-8(b) shows the vector field and a vorticity iso-surface (0.15 voxels/voxel).

exactly the same window sizes, PIV code and filtering routines. We will refer to the vector fields resulting from PIV processing of the reconstructed 3D intensity fields and the synthetic 3D intensity fields as the reconstructed vector field, and synthesized vector field, respectively.

Figure 2-9 shows scatter plots of the error in each vector component for the reconstructed vector field (a & b) and the synthesized vector field (c & d). The error is further characterized by calculating the 90% precision interval for the error in each vector component. This is summarized in table 2.2 along with the maximum magnitude of velocity in the exact field. The fourth column is the 90% precision interval for error between the reconstructed and exact field, and the fifth column is the 90% precision interval for the error between the synthesized and the exact field. The precision intervals seem rather large, but comparison between the error in the reconstructed and synthesized vector fields shows that the difference between the precision interval values are small compared to the actual error magnitude. This indicates that most of the error is due to the PIV algorithm itself, and a much smaller percentage is due to the actual intensity field reconstruction. This is supported by the error scatter plots in figure 2-9, which show only slightly more spread in the error for the reconstructed velocity fields. To examine this further, the numerical experiment was repeated with the velocity magnitude increased by 3 fold everywhere in the flow field. As shown in Table 2.2, the precision interval magnitude increases for both the reconstructed and synthesized vector fields, but the difference between the precision intervals essentially remains the same. This further points to the PIV algorithm as the largest source of error in the vector field. This is not surprising since the 3D PIV algorithm is not very sophisticated, and we would expect a reduction in error magnitude with a more advanced 3D PIV algorithm.

In the second simulated flow, a vortex ring of the same size is oriented with the

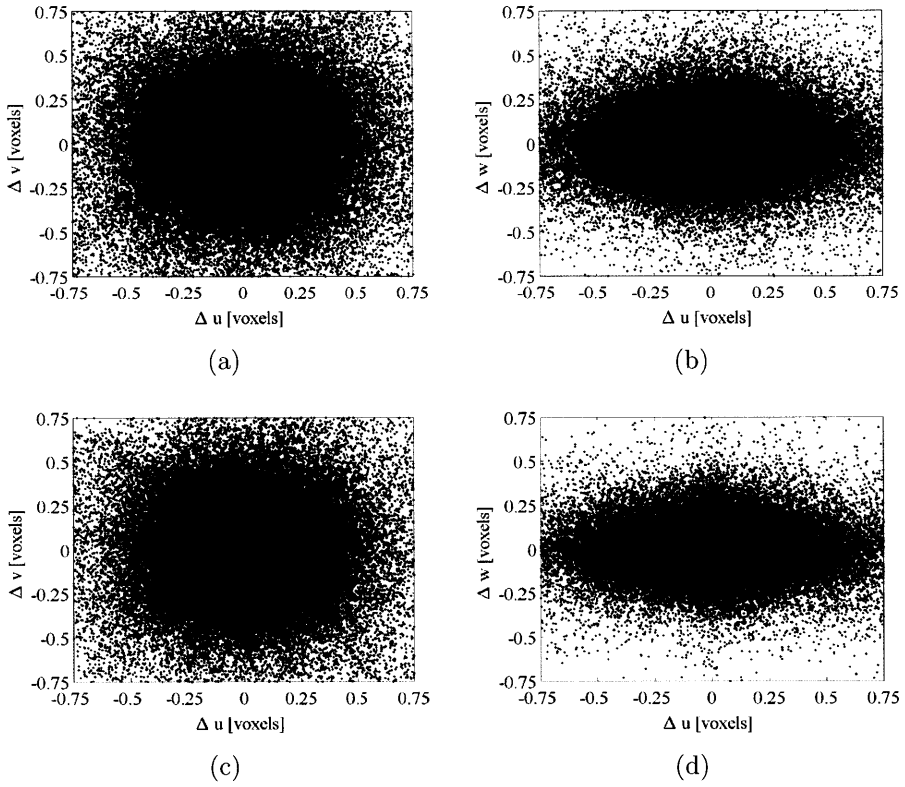
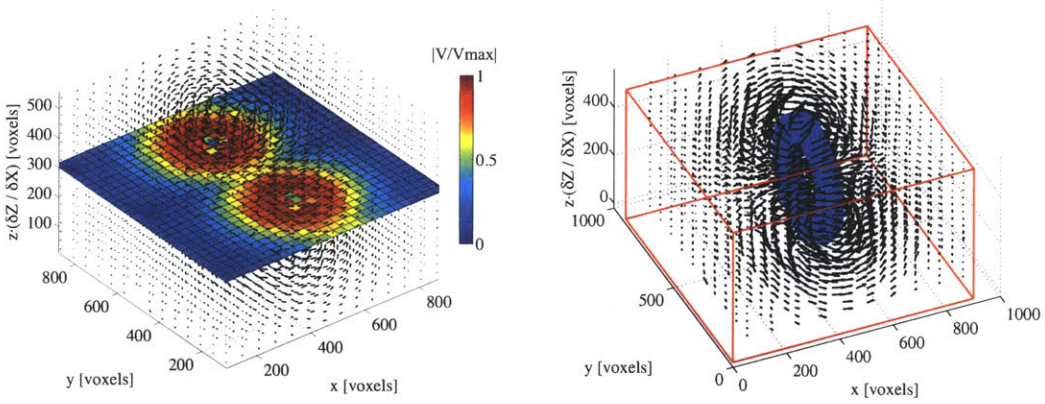


Figure 2-9: Scatter plots of the error in the reconstructed vector field (a - u, v components) & (b - u, w components), and error in the synthesized vector field (c - u, v components) & (d - u, w components). All plots are for the vortex ring in the 50 x 50 x 10 mm^3 volume.

Table 2.2: Summary of PIV error in three simulated flow fields. The fourth column is the 90% precision interval for error between the reconstructed and exact field, and the fifth column is for the error between the synthesized and the exact field.

Measurement Volume (mm^3)	Component	$\|V_{max}\|$ (voxels)	90% precision interval (voxels) - recon.	90% precision interval (voxels) - syn.
50 x 50 x 10	u	6	$\pm$ 0.37	$\pm$ 0.33
	v	6	$\pm$ 0.38	$\pm$ 0.34
	w	2.9	$\pm$ 0.20	$\pm$ 0.14
50 x 50 x 10	u	18	$\pm$ 1.69	$\pm$ 1.62
	v	18	$\pm$ 1.72	$\pm$ 1.62
	w	8.7	$\pm$ 0.90	$\pm$ 0.80
40 x 40 x 30	u	7.3	$\pm$ 0.54	$\pm$ 0.48
	v	7.3	$\pm$ 0.43	$\pm$ 0.36
	w	2.9	$\pm$ 0.19	$\pm$ 0.14

central axis of the toroid parallel to the X-axis, such that the ring spans deeper into the flow in the Z dimension. The 5x5 model camera array is used with spacing $D = 0.4$ (uneven spacing, see Table 2.1) and the magnification is set to $M(Z = 0) = -0.2$. The maximum displacement in the flow field is 0.37 mm which corresponds to 7.3 voxels in X and Y and 2.9 voxels in Z. A particle seeding density of $C = 1$ part/(mm)^3 in a 40 x 40 x 30 mm^3 volume results in a distribution of 48,000 particles (resulting in $Q = 0.76$). The lower seeding density requires larger interrogation volume sizes in order to contain an appropriate number of particles, therefore a final interrogation volume containing 60 x 60 x 24 voxels is used with each containing 27 particles based on the seeding density. Using 50% overlap in the multipass 3D PIV calculation yields 18,432 vectors (32 x 32 x 18), which includes the imaged area that contained no seeding particles (with a magnification of -0.2, images of the seeded volume did not span the entire imaging sensor). Figure 2-10 shows the vector fields results, with figure 2-10(a) revealing velocity magnitude on a slice in an $X - Y$ plane



(a) (b)

Figure 2-10: Three-dimensional vector field resulting from PIV processing of the reconstructed intensity volumes using SAPIV in the 40 x 40 x 30 mm^3 volume. Figure 2-10(a) shows slice in an $X - Y$ plane with normalized velocity magnitude contours and figure 2-10(b) shows the vector field and a vorticity iso-surface (0.15 voxels/voxel).

with every second vector plotted in X and Y and every vector plotted in Z. Figure 2-10(b) shows the vector field with every third vector plotted in X and Y and every vector plotted in Z and 0.15 voxels/voxel vorticity iso-surface.

Figure 2-11 shows the error scatter plots resulting from the difference in velocity components between the reconstructed and exact vector fields (a & b) and the synthesized and exact vector fields (c & d). Table 2.2 summarizes the 90% precision interval for this experiment as well. The data again indicate that the largest percentage of the error is due to the PIV algorithm. The reason for the difference in errors in the X and Y dimensions for this case is likely due to the orientation of the vortex ring, which induces larger spatial gradients in the X-Z plane than in the Y-Z plane.

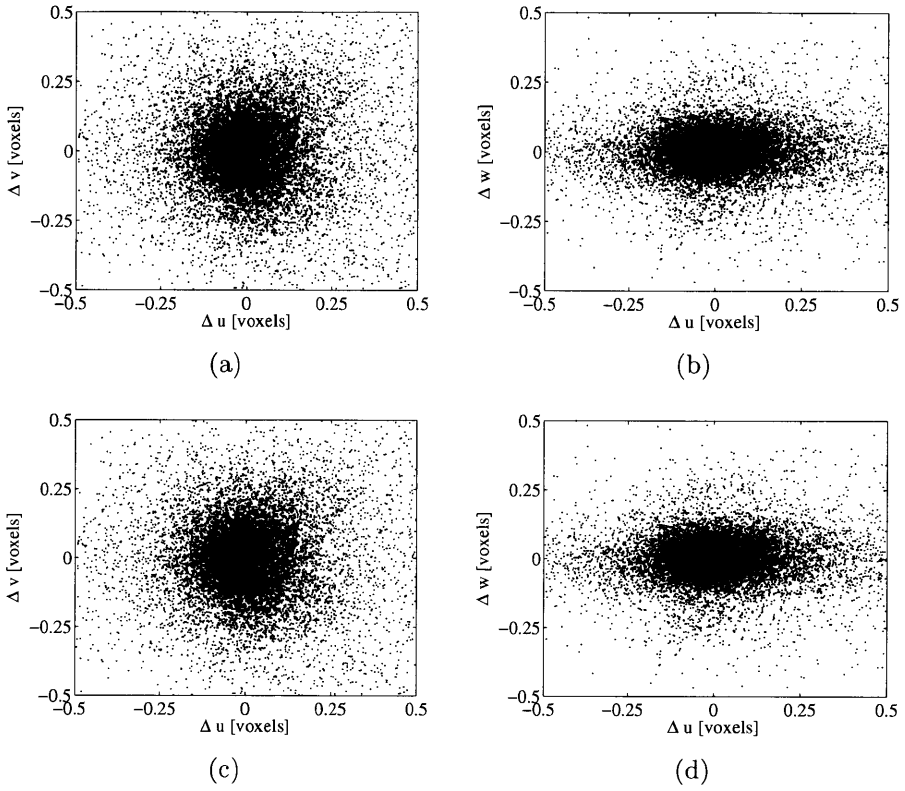


Figure 2-11: Scatter plots of the error in the reconstructed vector field (a - u, v components) & (b - u, w components), and error in the synthesized vector field (c - u, v components) & (d - u, w components). All plots are for the vortex ring in the 40 x 40 x 30 mm^3 volume.

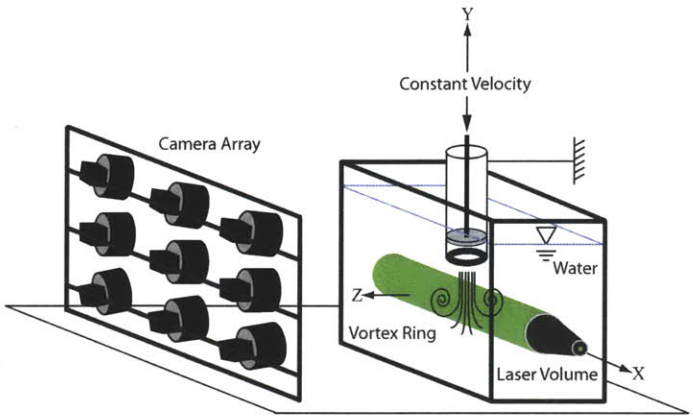


Figure 2-12: Experimental setup with camera array imaging a vortex ring illuminated with a laser volume.

2.3 SAPIV Experimental Implementation

2.3.1 Experimental Apparatus

To illustrate the capabilities of the SAPIV technique in practice, a canonical 3D flow field is captured experimentally using an array of eight cameras. Instantaneous 3D-3C SAPIV and classic 2D PIV velocity data of a piston-generated vortex ring are acquired for comparison.

The experiment is conducted in a glass tank (504 x 254 x 280 mm^3) seeded with neutrally buoyant particles (50 μm diamter) yielding an estimated seeding density of $C = 0.23$ particles/mm^3. A cylindrical piston-driven vortex ring generator (40 mm inner diameter) is mounted in the center of the tank. The outlet orifice diameter of the cylinder is 30 mm and the piston stroke length and velocity are 10 mm and 15 mm/s respectively. Based on the total stroke length (as opposed to integral of piston velocity over time), the ratio of stroke length to outlet orifice diameter is $L/D_o = 0.33$. The camera array images from the side of the tank (figure 2-12)

with all cameras mounted on a common frame. Initially, the array included nine cameras; however, the camera in the upper left corner of figure 2-12 had outdated firmware that did not support the appropriate triggering and therefore is not used in this study.

The cameras are Point Grey Research, Inc. Flea2 cameras (FL2-08S2M/C), arranged in a rectangular array mounted on 80/20 ® aluminum rails (figure 2-13), with ≈150 mm spacing between cameras. Cameras are angled in order to overlap the fields of view. The array is placed approximately 760 mm from the center of the water tank. As the number of cameras in the array increases, the hardware cost associated with the array will increase. Thus it is important to consider the minimum number of cameras necessary for successful volume reconstruction. Simulations showed that the reconstruction quality reaches a point of diminishing returns as more cameras are added, and that the most efficient number of cameras seems to be in the range of 10-15, as shown in figure 2-7(a).

All eight cameras in the array capture 1024 x 768 pixels, 8 bit, monochromatic images at 10 frames per second. Each camera is equipped with a Nikon Nikkor 50 mm lens. At the given frame rate, each camera requires 8 megabytes per second (MBps) of bandwidth, totaling 64 MBps to capture the entire event. The cameras use the IEEE 1394b serial bus interface standard which is rated to support 80 MBps for each bus, which provides ample bandwidth for the eight cameras. The cameras are connected to a single computer which records the data onto three hard drives. A custom C++ program using PointGrey's SDK libraries was developed in-house to interface with the cameras. The data are stored on one of three hard drives in AVI files with a raw encoding to maintain information integrity (i.e., no image compression). After completion of the capture sequence, the software reorganizes the data to place all of the necessary files into a single location.

Figure 2-13: Photograph of the camera array. Nine cameras are mounted with 50 mm lenses on an extruded aluminum frame. Only eight cameras are used in the study.

While the cameras record at 10 frames per second (fps), classic PIV frame straddling timing is used to obtain appropriate image pair time spacing. A Litron 532nm, 180 mJ/pulse, dual cavity Nd:YAG laser is used to illuminate the particle volume. The laser pulse duration is 4 ns, and inter-frame straddling time is 8 ms. A 10X beam expander creates a cylindrical laser volume with a 40 mm diameter, and a 10 mm fanning optic spreads the laser beam into a 1 mm thick laser sheet for the 2D PIV images. A Berkeley Nucleonics Corporation timing box is used to synchronize the cameras and laser. While the focus of this effort is 3D-3C PIV using SAPIV, 2D PIV images of the vortex ring are used for benchmarking and comparison.

2.3.2 Particle Volume Reconstruction

To achieve proper focus in synthetic aperture images, accurate mapping between image and world coordinates is required. The mapping is found by imaging a preci-

sion machined calibration plate traversed through the target volume with Z location increments of 2 mm. Since the SAPIV technique involves reprojecting the images onto several planes throughout the volume, a suitable coordinate system must be established on the calibration plate. Here, we use the average calibration in pixels/mm from the center camera image of the plate at the Z location farthest from the cameras to convert the reference geometry of the calibration plate from mm to pixels. Second-order polynomial fits are used to map image coordinates to reference coordinates on each Z calibration plane, and linear interpolation is used to find the polynomial fits on Z planes between each calibration plane. This approach follows that of prior Tomographic-PIV studies where polynomial fits are used to deal with the distortion introduced when imaging through an air-glass-water transition [28, 17]. As described in [28], even with higher order fits, the errors in mapping functions can be in excess of 0.5 pixels, but volume self-calibration can be used to reduce the errors to 0.1 pixels or less. As discussed in section 2.2.4, the error in the mapping functions should be less than 0.45 pixels for adequate reconstruction in SAPIV. Volume self-calibration is not implemented in the present experiment, yet reconstruction still yields a volume which is suitable for 3DPIV. We expect that implementing volume self-calibration will improve greatly the particle yield rate in the reconstruction.

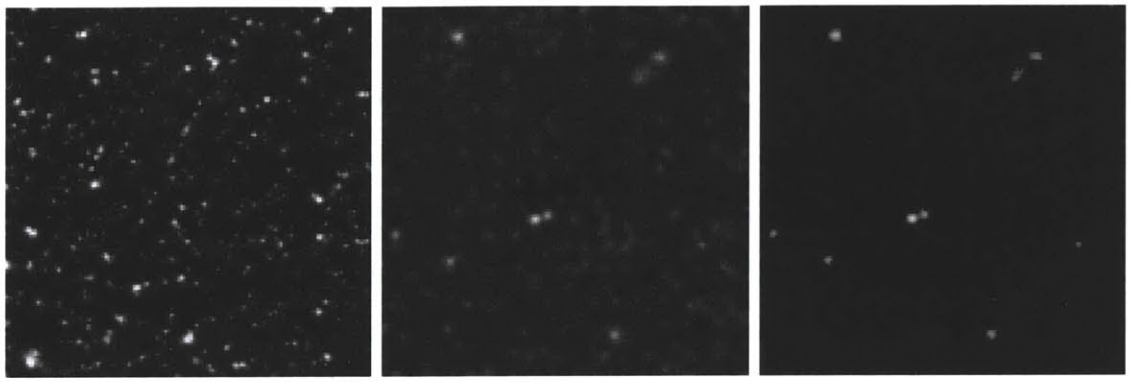
The synthetic aperture images are formed with focal plane spacing of 0.2 mm in Z. Theoretically, the number of focal planes that can be generated within an illuminated volume is infinite, but the information on each plane will not necessarily be unique, as the sharpness of a refocused object is determined by the degree to which mapped images of the object overlap. For example, it is possible to generate focal planes very close to each other by shifting all images by a very small amount. However, if the shift size is much smaller than the refocused object, the two refocused images will be essentially indistinguishable, in which case the information is redundant. Therefore,

focal plane spacing (which determines voxel size) should be made large enough so as not to retain redundant information. The depth over which objects are in focus can be controlled by changing the camera baseline, D; for the present experiment, D = 0.2. Ultimately, smaller focal plane spacing should yield better resolution in the Z dimension of the reconstructed fields, and thus the vector fields. The influence of focal plane spacing and camera baseline on the accuracy and resolution of 3DPIV vector fields is the subject of ongoing work.

Other volumetric PIV studies have discussed the need for image preprocessing to deal with non-uniformities in laser profiles and pulse intensities, as well as to remove background noise [31, 28]. Prior to refocusing, images in this study are subjected to the following preprocessing steps:

1. Subtract sliding minimum (window size = 10 pixels)
2. Convolve with 3x3 Gaussian kernel
3. Equalize histograms to histogram of image with highest contrast
4. Increase contrast by trimming the bottom and top 0.1% of intensity values
5. Subtract sliding minimum (window size = 10 pixels)

After preprocessing, the images are mapped to each plane throughout the volume, averaged to form the synthetic refocused image, and thresholded to retain particles to generate an intensity volume for each time instant. Because the mapping functions are not simple linear homographies, interpolation is required to re-project the images; here, a bilinear interpolation is used. Figure 2-14 shows a preprocessed image from the array, a synthetic aperture refocused image at one depth (Z = 5 mm) and a thresholded image at the same depth. It is estimated from the preprocessed images



(a) (b) (c)

Figure 2-14: Zoomed views (140 x 140 pixels) of a preprocessed 3DPIV image from single camera of the array (a), a refocused image at $Z = 5$ mm (b), and thresholded image at the same depth (c).

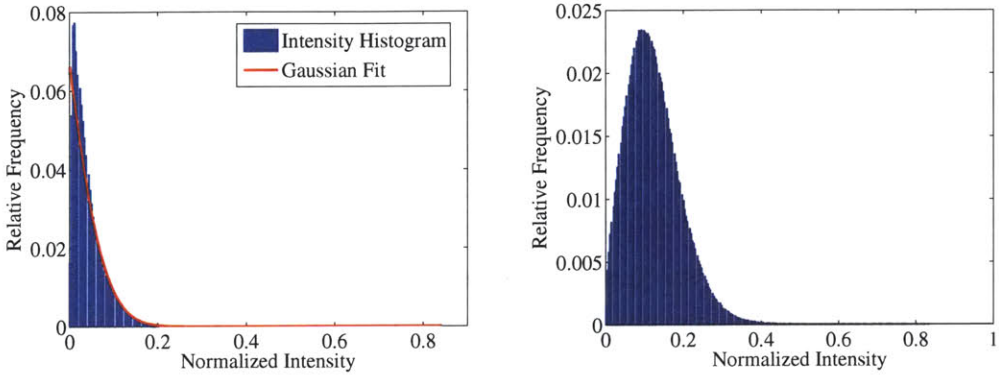
that the seeding density is 0.026 partices/pixel (ppp) ($C = 0.23$ particles/$(\text{mm})^3$), and the actual number of particles in the reconstructed volumes is 9863 particles for the first time instant and 9890 particles for the second time instant ($C = 0.13$ particles/$(\text{mm})^3$). The reason the yield rate is low in the reconstructed volumes is due to errors in the calibration which have not been reduced through the use of volume self-calibration. Images of particles that span two pixels or less in the original camera images tend to not be properly reconstructed in 3D space because the calibration error can be on the same order as the particle size.

Figure 2-15(a) shows the intensity histogram and Gaussian fit for a refocused image from the 8-camera array experiment. The mean intensity value is very low, but actual particles in the refocused image have high intensity; therefore the signal-to-noise ratio in the refocused image is very large and the in-focus particles can readily be determined (figure 2-14(c)). Figure 2-15(b-d) show the histograms from simulations in a 50 x 50 x 10 mm^3 volume imaged with 9 cameras and seeding density of 0.05 ppp (b), 0.075 ppp (c) and 0.125 ppp (d). With increasing seeding

density, the mean and standard deviation of the intensity of the refocused images increases, which reduces the signal-to-noise ratio of actual particles. Despite the reduced signal-to-noise ratio, the simulations produced adequate reconstruction of the intensity volume for the 0.075 ppp case (reconstruction quality, Q = 0.77) and also provided reasonable reconstruction for the 0.125 ppp case (Q = 0.71). The signal-to-noise ratio of actual particles for each of these simulated cases is much lower than for the SAPIV experiment, indicating that seeding density can be greatly increased in future experimental studies.

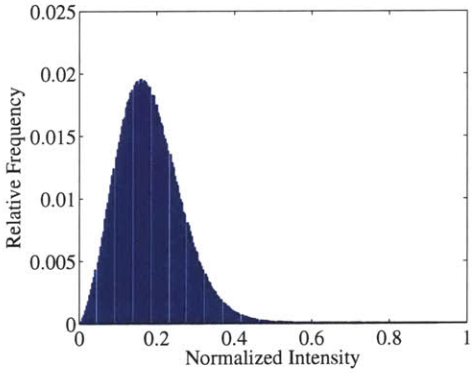
Once reconstructed, the intensity volumes are ready for cross-correlation based 3DPIV analysis; the adapted version of matPIV is again employed. A multi-pass algorithm with one pass at an initial interrogation volume size of 128 x 128 x 64 voxels and two passes at an final interrogation volume size of 64 x 64 x 32 voxels and 50% overlap generates a 23 x 31 x 11 vector field. Each 64 x 64 x 32 voxels interrogation volume contains approximately 15 particles. The resultant vector field resolution is 2.1 mm in X and $Y$and 3.2 mm in Z. Post-processing consists of a filter based on signal-to-noise ratio of the cross-correlation peak, a global filter which removes vectors five standard deviations above the mean of all vectors, and a local filter which removes vectors which deviate by more than three standard deviations from the median of a 3 x 3 x 3 vector window. The filtered field is interpolated using linear interpolation and smoothed with a 3 x 3 x 3 gaussian filter. At this point some mention should be made of the overall processing time. The time required to reconstruct the two volumes used to generate the 3D vector field is 18% of the time required for the 3DPIV processing of the fields. Therefore, the limiting time factor in processing is the 3DPIV analysis, which demonstrates the relative efficiency of the synthetic aperture refocusing technique.

Processing of the 2D PIV data is also performed with matPIV; a multi-pass

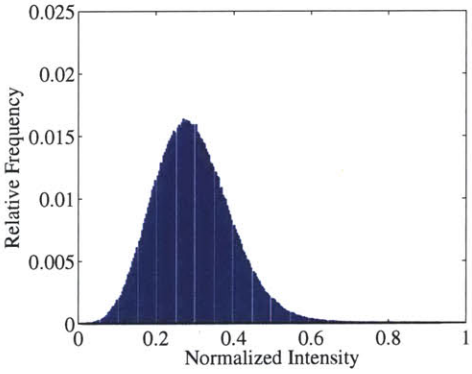


(a) Experiment: 0.026 ppp

(b) Simulation: 0.05 ppp



(c) Simulation: 0.075 ppp



(d) Simulation: 0.125 ppp

Figure 2-15: Intensity histograms and Gaussian fit for a refocused image from from the 3D SAPIV experiment with 8 cameras (a), and intensity histograms for refocused images from simulations in 50 x 50 x 10 mm^3 volume with 9 cameras and seeding density of 0.05 ppp (b), 0.075 ppp (c) and 0.125 ppp (d).

algorithm with one pass using initial interrogation window size of 64 x 64 pixels and two passes with a final window size of 32 x 32 pixels and 50% overlap is used to create a vector field with vector spacing of 1.03 mm. Post-processing for the 2D vector fields also consists of the signal-to-noise, global and local filters, as well as linear interpolation and smoothing with a 3 x 3 gaussian kernel.

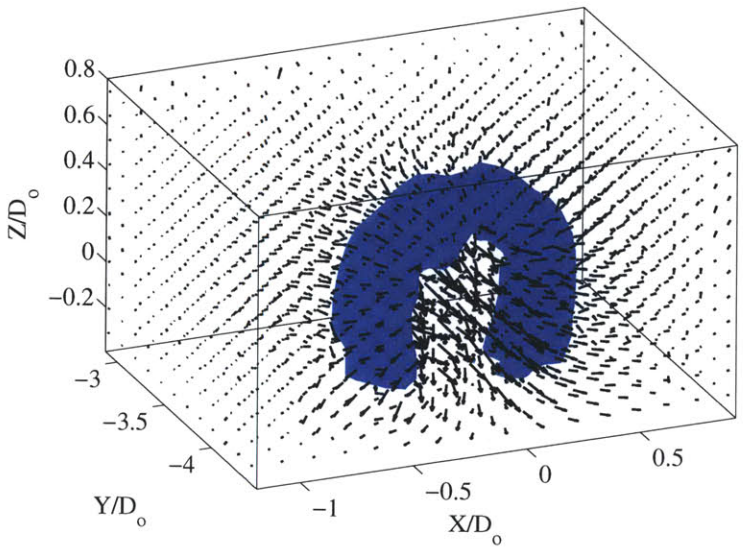
2.3.3 Experimental Results

Experimental results for the instantaneous 3D velocity data of the vortex ring are shown in figure 2-16(a); the resultant 3D vector field and an iso-vorticity contour (magnitude 9 s^{-1}) are plotted at one time instant. For ease of comparison between the 3D SAPIV and 2D PIV data, we normalize all lengths by the orifice diameter ($D_o = 30$ mm), as is done in Weigand & Gharib [32]. The origin of the X-Y-Z global coordinate system is placed approximately at the center of the outlet orifice of the vortex generator, with Y positive up and Z decreasing in the direction away from the cameras. In figure 2-16(a) the ring has propagated to a distance $Y/D_o \approx 3.72$ below the orifice outlet. Every vector in Z is plotted and every second vector is plotted in X and Y directions. The measured SAPIV volume is only limited by the volume which is illuminated by the laser. The iso-vorticity contour shows an incomplete ring, due to the fact that the ring is not centered in the laser volume and part of the ring is outside of the illuminated region. Cross sectional slices of vorticity, with planes at $Z/D_o = 0.003$ and $X/D_o = 0$, are plotted in figure 2-16(b). As expected, isolated regions of vorticity are located where the ring passes through the planar cut in the 3D volume. The vorticity magnitude on the X-Z slice is slightly lower than for the X-Y, which is likely due to the lower resolution of the vector field in the Z-dimension. By locating the peaks in the normal vorticity component in each core

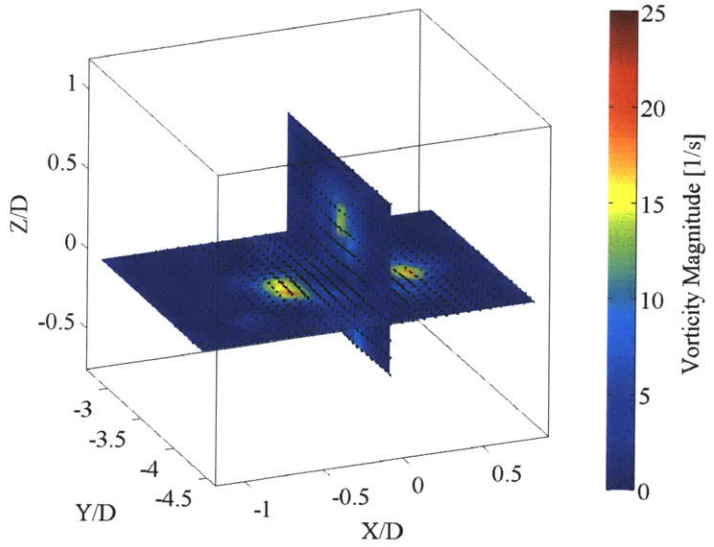
cross-section on the plane passing through $Z/D_o = 0.003$, the normalized diameter of the vortex is found to be $D/D_o = 0.77$.

Data from the 2D PIV experiments are used to compare and benchmark the 3DPIV results. Figure 2-17(a) shows the velocity and vorticity on an X-Y plane of the 3D SAPIV data at $Z/D_o = 0.11$, and figure 2-17(b) shows velocity and vorticity resulting from a 2D PIV experiment, where the laser sheet is located at approximately $Z/D_o = 0.15$. The slice of the 3D data at $Z/D_o = 0.11$ is the location in the vector field nearest to the 2D laser sheet. Although the 2D and 3D experiments could not be performed simultaneously, the experimental vortex ring generator is designed to be repeatable, and the vortex ring is at approximately the same Y-location for the 2D and 3D vector fields shown in figure 2-17. Qualitatively, the topology of the vortex ring compares well between the slice from the 3D and 2D data. The distance between the positive and negative cores of the vortex ring is $D/D_o = 0.82$ for the 2D data, and $D/D_o = 0.77$ for the 3D plane at $Z/D_o = 0.11$.

For quantitative comparison, the velocity, vorticity and circulation are examined and compared for the 3D and 2D data. In figure 2-18(a), u and v velocity profiles from the 2D vector field are plotted against X/D_o at Y/D_o = -3.75. The profiles show good qualitative agreement with those presented in [32]. In addition to the 2D profiles, two profiles from the 3D data are plotted. The u and v velocity profiles are plotted against X/D_o at Y/D_o = -3.72 and $Z/D_o = 0.11$ (X-Y plane), and the v and w velocity profiles are plotted against Z/D_o at Y/D_o = -3.72 and $X/D_o = 0$ (Y-Z plane). The Y-Z velocity profile contains no data for $Z/D_o < -0.32$, due to the fact that the ring is not entirely in the illuminated volume and thus not resolved beyond $Z/D_o < -0.32$. Profiles from the 3D data show good quantitative agreement with the 2D profiles. The maximum negative v velocity from the 3D X-Y and Y-Z profiles are 6% and 8% below the maximum negative v velocity for the 2D profile,



(a)



(b)

Figure 2-16: (a) Experimental SAPIV velocity vector field for the vortex ring with an iso-vorticity contour of magnitude 9 s^{-1} plotted in blue. (b) X-Y and Y-Z cross-sectional cuts of vorticity through the vortex ring center, with superimposed velocity vectors.

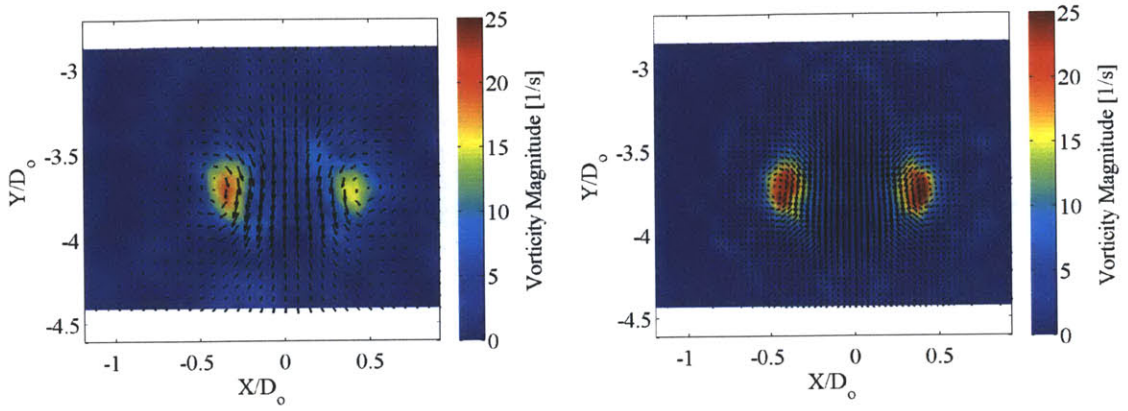


Figure 2-17: Vorticity contours and vector fields from (a) 3DPIV cut and (b) 2DPIV instantaneous slice through the vortex ring.

respectively. The profiles from the 3D data also capture the sign reversal in v velocity as the profile moves through the core, and the spatial decay in velocity moving away from the core.

Along each of the same profiles, the component of vorticity normal to the plane containing the profile is plotted in figure 2-18(b). The trends in all profiles agree very well, but the vorticity calculated from the 3D data underestimates the peak magnitude as compared with the vorticity from the 2D data. This is likely due to the lower spatial resolution in the vector fields (2D data was processed with smaller interrogation windows), which results in a spatial averaging of velocity gradients during the cross-correlation calculation. Also, the smoothing implemented in the post-processing of the vector field is more likely to remove gradients in the 3D data, because the gaussian kernel is three-dimensional, and thus the smoothing is based on more neighboring vectors.

Finally, to serve as another quantitative measure for benchmarking the 3D SAPIV

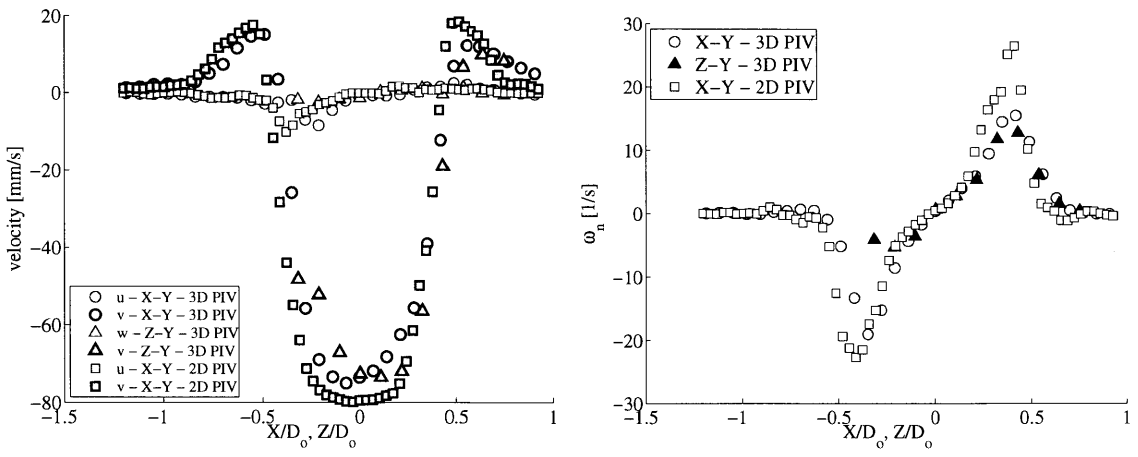
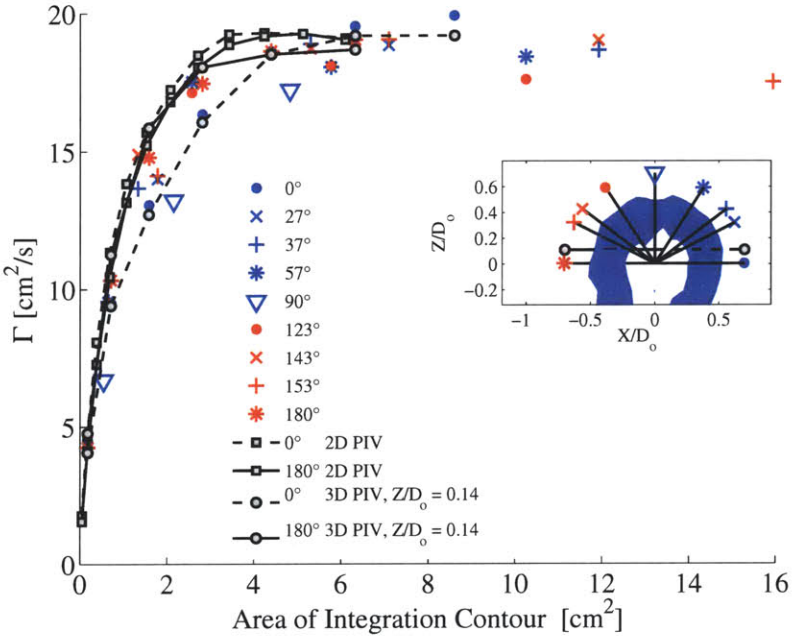


Figure 2-18: (a) Velocity profiles (u,v & w) along axes passing through the center of the vortex ring for 2D and 3D cuts. (b) Vorticity profiles along same planes used in (a) for 2D and 3D cuts normalized by the diameter of the vortex generator orifice (D_o).

system the circulation, Γ, is calculated on a variety of planes. The circulation is calculated by taking the line integral of velocity around a rectangular contour on a particular plane. Here, we calculate the circulation in the 2D data, as well as on several planes for the 3D data, using rectangular contours of increasing size to encompass up to one half of any 2D slice (i.e., encompassing one vortex core). Each contour is centered around the location of maximum surface-normal vorticity on the plane under consideration. The inset of figure 2-19 shows the location and angle of each plane on which circulation is computed. For the 3D data, nine planes are chosen with angles varying between $0° - 180°$, as well as one additional plane located at $Z/D_o = 0.11$, on which the circulation is calculated for both cuts through the vortex ring core. Figure 2-19 shows the circulation plotted against area enclosed by the integration contour for each plane.

For a symmetric vortex ring, the circulation on half of any one plane containing



(a)

Figure 2-19: Distribution of circulation as a function of area enclosed by integration contour on planar slices for several angles of the vortex ring. Planes of interest and their positions are shown in the inset and labeled in the legend.

the axis passing through the ring center should be constant. From figure 2-19, it can be seen that the magnitude of circulation remains relatively constant regardless of plane angle, for a given integration contour size. The maximum difference in peak circulation is 2.69 cm^2/s (13.5 % of maximum) for the angled planes. On the plane passing through $Z/D_o = 0.11$, as well as for the 2D data (laser plane at $Z/D_o = 0.15$), the peak circulation is reached at a smaller integration contour size than for the planes passing through the central ring axis, because the bisected cross-sections are offset from the center of the ring. Nonetheless, the maximum circulation magnitude on the offset planes for both the 2D and 3D data is within 7% of the maximum found on the planes which pass approximately through the central axis of the ring.

The quantitative agreement between the 3D SAPIV data and the 2DPIV data confirms the viability of the SAPIV technique for making accurate measurements in 3D volumes. Although the simulations show the ability to reconstruct very densely seeded fields, the seeding density was kept rather low (0.026 ppp in the raw images) in this experiment to ensure proper reconstruction. However, it is expected that increased seeding density can be achieved in practice.

2.4 Conclusion

Synthetic aperture PIV offers a novel and exciting method for imaging complex 3D flow-fields with high seeding densities and partial occlusions. SAPIV draws on the concept of light field imaging, which involves sampling and encoding many rays from a 3D scene, and is practically implemented with an array of cameras. Recombining the camera array images using synthetic aperture refocusing provides many desirable capabilities for 3D fluid flow measurement; namely, the ability to digitally refocus

on isolated planes post-capture, to effectively "see-through" partial occlusions by exploiting the multiple camera viewpoints and to capture volumetric information at a single time instant. We expect the capabilities of the synthetic aperture system to be flexible enough to measure in other flow environments, such as multi-phase and bubbly flows or flows with partial obstructions.

Simulations showed that a single array arrangement allowed for measurement within volumes with depth ranging from 10mm to 50mm. Altering the optics on the cameras enables further scalability of the measurement range, as was shown in the simulation of the 100 x 100 x 100 mm^3 volume. In this manner, the behavior of the camera array is similar to the behavior of a single-lens camera: we have control over the viewable depth for a given magnification and can change the FOV by changing the magnification. Two simulated flow fields demonstrated the performance of the technique in resolving vector fields with high resolution and in a relatively large volume. The focal plane spacing of the system in the Z dimension, which is related to the depth of field, was theoretically derived for the simple model of two coplanar image sensors. The observed focal plane spacing in the simulations agreed extremely well with that predicted by the theory, despite the fact that the camera image sensors in the simulated model were not coplanar (the cameras were angled). This shows that the concise theory derived is an accurate and useful tool for predicting the depth of field of the refocused images as a function of camera baseline and optics.

The results of the 3D PIV experiment indicate that SAPIV is a viable technique for efficiently and accurately resolving a range of 3D flow fields. In practice, the hardware implementation successfully captured an instantaneous 3D velocity vector field for a vortex ring, with only eight cameras, in a volume with an aspect ratio (Z:X-Y) that is comparable to some of the largest found in the literature. 3D SAPIV results compared well with the 2D PIV experimental data for a similar vortex

ring. The signal-to-noise ratio of actual particles for each of the simulated cases was much lower than for the SAPIV experiment, indicating that seeding density can be greatly increased in future experimental studies, which will allow for increased vector resolution.

The simulations indicate that synthetic aperture PIV can resolve particle fields with seeding density and volume sizes competitive with the state-of-the-art; however, we have made no mention of the efficiency with which the system performs. We have reported the number of particles per pixel based on the number of pixels in one camera, perhaps a measure of efficiency is to divide the particle number by the *total* number of pixels (e.g. 13 cameras $\times$ 1000 pixels $\times$ 1000 pixels). Table 2.3 compares this measure of efficiency for several volumetric studies, including the $D = 0.4$ configuration for the 50 x 50 x 10 mm^3 simulation using 13 cameras and the 8 camera experiment from the present study, as well some representative results from Tomographic-PIV, Holographic-PIV, Defocusing DPIV and PTV. The comparison is not intended to give a concrete answer as to the most efficient method, rather to test whether the efficiency of the synthetic aperture PIV technique lies in the realm of other techniques, as it does.

Regardless of the notion of efficiency, performance is the bottom line for a 3D PIV system, if the cost is acceptable. Thus, a practical consideration that naturally arises is how to deal with such a high number of cameras needed for synthetic aperture PIV. Certainly, with a large number of cameras the cost of a system becomes a concern. We have developed an eight camera array which performs double-pulsed frame straddling PIV at 10 frames per second with a total system cost (excluding laser) of less than $15,000, in present-day dollars. An array of high speed cameras for fully time-resolved experiments will be more costly, but SAPIV offers the ability to trade-off individual camera sensor size for a technique that synthesizes smaller

Table 2.3: Comparison of efficiency of some 3D PIV techniques.

Technique	*Volume Size (mm^3)*	*# cameras*	*# part.*	*part./pixel*	*part./ total pixel #*
SAPIV (simulation)	50 x 50 x 10	13	125000	0.125	0.01
SAPIV (experiment)	65 x 40 x 32	8	9860	0.015	0.002
Tomo-PIV [15]	35 x 35 x 7	4	24500	0.05	0.013
Tomo-PIV [16]	37 x 36 x 8	4	114500	0.08	0.02
Tomo-PIV [17]	100 x 100 x 20	4	100000	0.024	0.006
HPIV [33]	10 x 10 x 10	1	2000	0.0015	0.0015
HPIV [23]	1.5 x 2.5 x 1.5	1	56250	0.014	0.014
DDPIV [9]	100 x 100 x 100	3	40000	0.038	0.013
DDPIV [34]	150 x 150 x 150	3	35000	0.034	0.011
PTV [4]	200 x 160 x 50	3	3000	0.0038	0.0013

sensors, potentially making a high-speed system affordable for 3DPIV.

Another form of cost is the required computation time; all data processing was performed on a Macintosh Power Mac G5 computer with a 3 GHz Quad-Core Intel Xeon Processor and 2 GB of RAM. Reconstruction and 3D PIV analysis was implemented in Matlab, and the codes are not optimized at this point. However, the computation time to implement the map-shift-average algorithm, refocus and threshold the images, and assemble them into the reconstructed volume for two timesteps in the simulated 40 x 40 x 30 mm^3 volume required 15% of the time taken to compute the vector fields with 3 passes and 50% overlap (67 minutes to reconstruct the two timesteps, and 446 minutes to perform the PIV processing). For the SAPIV experiment, the time required to reconstruct the two volumes used to generate the 3D vector field (62 minutes) was 18% of the time required for the 3DPIV processing of the fields (414 minutes). This attests to the relative simplicity of the refocusing algorithm. Therefore, the actual 3D PIV calculations will dominate the computation time for synthetic aperture PIV.

In order to fully realize the capabilities of SAPIV, further work is required to

address several practical issues and challenges, and is ongoing. For example, volume self-calibration can greatly improve the image reconstruction quality and allow for increased seeding densities. The ability of the SAPIV technique to reconstruct the intensity fields without the use of volume self-calibration in the present study underscores the capability of the method. In addition, increasing the camera baseline spacing is expected to increase Z resolution. By increasing the baseline, the depth-of-field can be reduced allowing for more distinction between particles in the Z direction, which we expect will yield higher resolution in Z. In practice, this requires further investigation to determine acceptable minimum and maximum baseline spacing limits, as well as the maximum flow volume size that can be resolved for a given baseline spacing.

Ultimately, Synthetic Aperture PIV (SAPIV) provides a new and novel method for 3D-3C, quantitative flow velocimetry, which offers the ability to reconstruct very dense flow fields in relatively large volumes for a wide range of applications.

Acknowledgements

The authors would like to thank Dr. Ramesh Raskar for introducing us to the subject and for many fruitful discussions and Roderick R. LaFoy for his help with initiating the project. We gratefully acknowledge funding provided by the Office of Naval Research Contract N00014-09-1-1167 managed by Steven J. Russell and the Office of Naval Research (ONR) In-house Laboratory Independent Research funds.

Bibliography

[1] Jesse Belden, Tadd T. Truscott, Michael Axiak, and Alexandra H. Techet. Three-dimensional synthetic aperture particle image velocimetry. *Measurement Science and Technology*, 21(12), 2010.

[2] R. J. Adrian. Twenty years of particle image velocimetry. *Experiments in Fluids*, 39(2):159–169, 08 2005.

[3] Markus Raffel, Christian E. Willert, and Jurgen Kompenhans. *Particle Image Velocimetry: A Practical Guide.* Springer-Verlag, Berlin Heidelberg, 1998.

[4] H. G. Maas, A. Gruen, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(2):133–146, 1993-07-01.

[5] Hans-Gerd Maas and Armin Gruen. Digital photogrammetric techniques for high-resolution three-dimensional flow velocity measurements. *Optical Engineering*, 34(7):1970–1976, July 1995.

[6] N.A. Malik, Th. Dracos, and D.A. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(4–5):1432–1114, September 1993.

[7] Jochen Willneff and Armin Gruen. A new spatio-temporal matching algorithm for 3D-particle tracking velocimetry. In *9th of International Symposium on Transport Phenomena and Dynamics of Rotating Machinery*, Honolulu, Hawaii, February 2002.

[8] F. Pereira, M. Gharib, D. Dabiri, and D. Modarress. Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. Application to bubbly flows. *Experiments in Fluids*, 29(7):S078–S084, 2000.

[9] Francisco Pereira and Morteza Gharib. Defocusing digital particle image velocimetry and the three-dimensional characterization of two-phase flows. *Measurement Science and Technology*, 13(5):683–694, 2002.

[10] Sang Youl Yoon and Kyung Chun Kim. 3d particle position and 3d velocity field measurement in a microvolume via the defocusing concept. *Measurement Science and Technology*, 17(11):2897–2905, 2006.

[11] E. E. Fenimore and T. M. Cannon. Coded aperture imaging with uniformly redundant arrays. *Appl. Opt.*, 17(3):337–347, 02 1978.

[12] L Kajitani and D Dabiri. A full three-dimensional characterization of defocusing digital particle image velocimetry. *Measurement Science and Technology*, 16(3):790–804, 2005.

[13] R L Grothe and D Dabiri. An improved three-dimensional characterization of defocusing digital particle image velocimetry (ddpiv) based on a new imaging volume definition. *Measurement Science and Technology*, 19(6):065402–065414, 2008.

[14] Wei-Hsin Tien, Patrick Kartes, Toru Yamasaki, and Dana Dabiri. A color-coded backlighted defocusing digital particle image velocimetry system. *Experiments in Fluids*, 44(6):1015–1026, 06 2008.

[15] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.

[16] Gerrit Elsinga, Bernhard Wieneke, Fulvio Scarano, and Andreas Schrder. Tomographic 3d-piv and applications. In *Particle Image Velocimetry*, volume 112 of *Topics in Applied Physics*, pages 103–125. Springer Berlin / Heidelberg, 2008.

[17] Fulvio Scarano and Christian Poelma. Three-dimensional vorticity patterns of cylinder wakes. *Experiments in Fluids*, 47(1):69–83, 07 2009.

[18] Dirk Micahealis, Bernhard Wieneke, Christian Tiedemann, and Rolf Henke. Visualization of small three dimensional turbulent structures using thin volume tomographic piv. In *13th International Symposium on Flow Visualization*, July 2008.

[19] K. D. Hinsch. Holographic particle image velocimetry. *Measurement Science and Technology*, 13(7):R61–R72, 2002.

[20] Joseph W. Goodman. *Introduction to Fourier Optics.* Roberts & Company, Greenwood Village, CO, USA 80111, 3 edition, 2005.

[21] Hui Meng, Gang Pan, Ye Pu, and Scott H. Woodward. Holographic particle image velocimetry: from film to digital recording. *Measurement Science and Technology*, 15(4):673–685, 2004.

[22] J. Zhang, B. Tao, and J. Katz. Turbulent flow measurement in a square duct with hybrid holographic piv. *Experiments in Fluids*, 23(5):373–381, 11 1997.

[23] J. Sheng, E. Malkiel, and J. Katz. Using digital holographic microscopy for simultaneous measurements of 3d near wall velocity and wall shear stress in a turbulent boundary layer. *Experiments in Fluids*, 45(6):1023–1035, 12 2008/12/01/.

[24] Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. Dynamically reparameterized light field. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 297–306, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[25] V. Vaish, G. Garg, E. Talvala, E. Antunez, B. Wilburn, M. Horowitz, and M. Levoy. Synthetic aperture focusing using a shear-warp factorization of the viewing transform. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05') - Workshops*, volume 3, pages 129–129. IEEE Computer Society, June 2005.

[26] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. *ACM Trans. Graph.*, 24(3):765–776, 2005.

[27] V. Vaish, B. Wilburn, N. Joshi, and M. Levoy. Using plane + parallax for calibrating dense camera arrays. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR04')*, volume 1, pages 2–9. IEEE Computer Society, June-July 2004.

[28] Bernhard Wieneke. Volume self-calibration for 3D particle image velocimetry. *Experiments in Fluids*, 45:549–556, 2008.

[29] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, Cambridge, United Kingdom, 2000.

[30] J. K. Sveen. An introduction to matpiv v.1.6.1. eprint no. 2, ISSN 0809-4403, Dept. of Mathematics, University of Oslo, 2004. http://www.math.uio.no/~jks/matpiv.

[31] G. Elsinga, B. Wieneke, F. Scarano, and B. van Oudheusden. Assessment of tomo-piv for three-dimensional flows. *6th International Symposium on Particle Image Velocimetry*, September 21-23, 2005.

[32] A. Weigand and M. Gharib. On the evolution of laminar vortex rings. *Experiments in Fluids*, 22(6):447–457, 04 1997.

[33] Juan P. L. C. Salazar, Jeremy De Jong, Lujie Cao, Scott H. Woodward, Hui Meng, and Lance R. Collins. Experimental and numerical investigation of inertial particle clustering in isotropic turbulence. *Journal of Fluid Mechanics*, 600:245–256, 2008.

[34] Francisco Pereira and Morteza Gharib. A method for three-dimensional particle sizing in two-phase flows. *Measurement Science and Technology*, 15(10):2029–2038, 2004.

Chapter 3

Practical Aspects of Synthetic Aperture Imaging

3.1 Introduction

This chapter presents several practical aspects of synthetic aperture (SA) imaging ranging from hardware realizations to computational performance. First, the design and operation of two multi-camera SA systems is presented. One system is the low-cost array discussed in Chapter 2, where details of the timing and control software are expounded upon. The other system consists of high-speed Photron cameras and is applied to a multiphase flow in Chapter 5. Some comparisons are then drawn between SAPIV and Tomographic-PIV, particularly with respect to computational performance, the ability to deal with occlusions and the issue of "ghost" particles.

3.2 Synthetic Aperture Imaging System Design

3.2.1 Low-Speed

The first application of SA imaging for fluid flows was for 3DPIV, as described in Chapter 2. Fully time-resolved PIV requires a high-speed camera(s) and laser, but frame-straddling PIV still allows for exploration of a variety of flows with an order of magnitude lower hardware cost. Therefore, the first camera array for SA imaging in this thesis was constructed with the goal of performing frame-straddling PIV [1] and is referred to as the low-speed camera array; the details of frame-straddling PIV are reviewed later in this section.

The low-speed camera array constructed for 3D SAPIV comprised machine vision cameras; the Flea 2 model FL2-08S2M/C from Point Grey [2]. The cameras, shown in Figure 3-1, are attractive due to their small size, low cost and C-mount threads, which allows for a variety of lenses to be used. Each cameras captures 1024 $\times$ 768 pixels (4.65 $\times$ 4.65 μm pixel size), 8 bit, monochromatic images. The cameras provide no on-board memory and stream data via the IEEE 1394b (FireWire) interface standard, which supports 800 megabits/second (Mb/s) transfer rate (equivalently 100 megabytes/second (MB/s)). Therefore, the host computer for this type of camera array must be configured to provide ample bandwidth for data transfer. A Dell Precision^{TM} 690 Workstation served as the host computer for the low-speed array.

Figure 3-2 displays a schematic of the data flow for the low-speed camera array. The array configuration strategy involves connecting cameras to PCI-e 1394b cards installed in the computer. Point Grey offers dual bus PCI-e 1394b cards that fit into a single PCI-e slot in the computer [2]. For the low-speed array in this thesis, at most two cameras were connected to each PCI-e 1394b bus with each bus supporting 80

MB/s transfer rate. A portion of the computer RAM is partitioned in a RAM disk configuration, which acts as a temporary secondary storage to allow for writing of the streamed images. The RAM disk is necessary because the write speed of a standard computer hard disk drive is typically too slow to support the total throughput of the camera array in real time. For the type of hard drive found in the host computer (SATA 7200 RPM) the typical write speed is 129 MB/s [3]. The images are instead written to RAM disk and transferred to the hard disk drive post-capture. The data throughput for each camera is given by

$$\left(\frac{0.8 \text{ MB}}{\text{frame}}\right) \times \left(\frac{F \text{ frames}}{\text{sec}}\right). \tag{3.1}$$

If N cameras are connected to each PCI-e bus and operated at the full frame rate of the Flea 2 model FL2-08S2M/C cameras (30 frames/sec) then the upper limit on cameras per bus is

$$\begin{aligned} N\left(\frac{0.8 \text{ MB}}{\text{frame}}\right) \times \left(\frac{30 \text{ frames}}{\text{sec}}\right) &\leq 80 \text{ MB/sec} \\ N &\leq 3 \text{ cameras.} \end{aligned} \tag{3.2}$$

For the low-speed array used in this work, only two cameras (or less) were used per bus. The limit on total number of cameras in the array then becomes either the RAM disk write speed or the number of PCI-e slots available on the computer. Six PCI-e slots are available on the Dell Precision^{TM} 690 Workstation, of which four are fitted with PCI-e cards. With four cameras per card (two cameras per bus, two buses per card) the total data throughput at full frame rate is (16)(0.8 MB/frame)(30 frames/sec) = 384 MB/sec. The write speed of the RAM of the host computer (DDR2 667 MHz (PC2-5300)) is 5333 MB/sec. Thus, the number

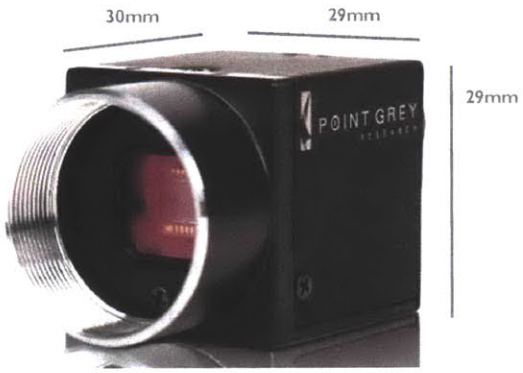


Figure 3-1: Picture of the Flea 2 model FL2-08S2M/C from Point Grey [2].

of PCI-e slots generates the limiting factor in total data throughput. As a final consideration for the data capture, the RAM disk partition limits the duration of the recording. Therefore, the total number of frames that can be acquired is a function of the amount of RAM available on the host computer, and the RAM disk must be smaller than the total amount of RAM.

A custom C++ program designed in house and based on the Point Grey Flycapture SDK initially managed the image acquisition[1]. However, the support software has since been switched to the commercial Streampix 5 software offered by Norpix [4]. The software facilitates setting of camera parameters, initiation of capture, recording onto the RAM disk and subsequent writing to the hard disk drive. These are the key aspects of the image acquisition software, and any software capable of performing these tasks should be adequate.

Another crucial consideration of the camera array not yet addressed concerns the timing of cameras during image capture. To properly refocus the data using the SA algorithms requires accurate synchronization of image capture across all cameras. While Point Grey's MultiSync^{TM} software synchronizes cameras across different PCI-

[1]Michael Axiak was responsible for the programming and testing of the code.

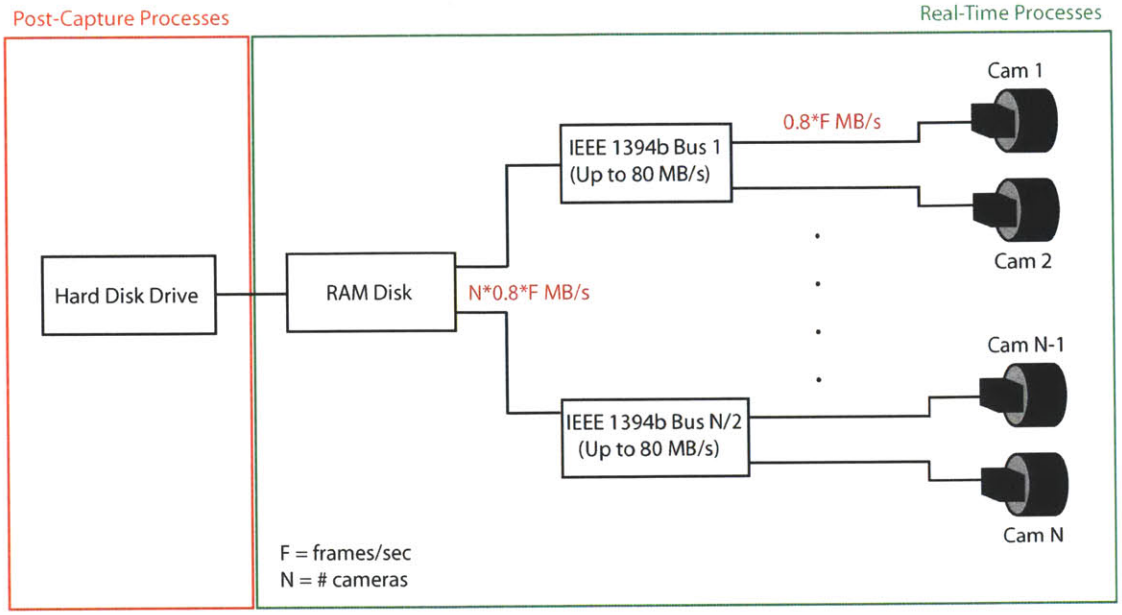


Figure 3-2: Data flow schematic for low-speed camera array.

e buses, the deviation is $\pm$ 125μs, which is too large for applications of SA imaging for fluid flows. Therefore, all cameras are hardware triggered using an external timing signal generator that provides much smaller latency than the software timing signal. The timing signal is generated with a Berkeley Nucleonics Model 505 Pulse Generator. Each camera is set to a mode (trigger mode 14) that starts frame exposure with a rising or falling pulse edge, exposes for a pre-set amount of time, and overlaps the frame readout with the next frame exposure.

Because one of the main applications of the low-speed camera array is frame-straddling PIV, the interframe time is also of critical importance. Figure 3-3 shows a timing diagram for a frame-straddling PIV setup. The interframe time, t_{if}, is the time between the end of one exposure and the beginning of the next. In frame-straddling PIV, image pairs are generated by pulsing one laser at the end of one

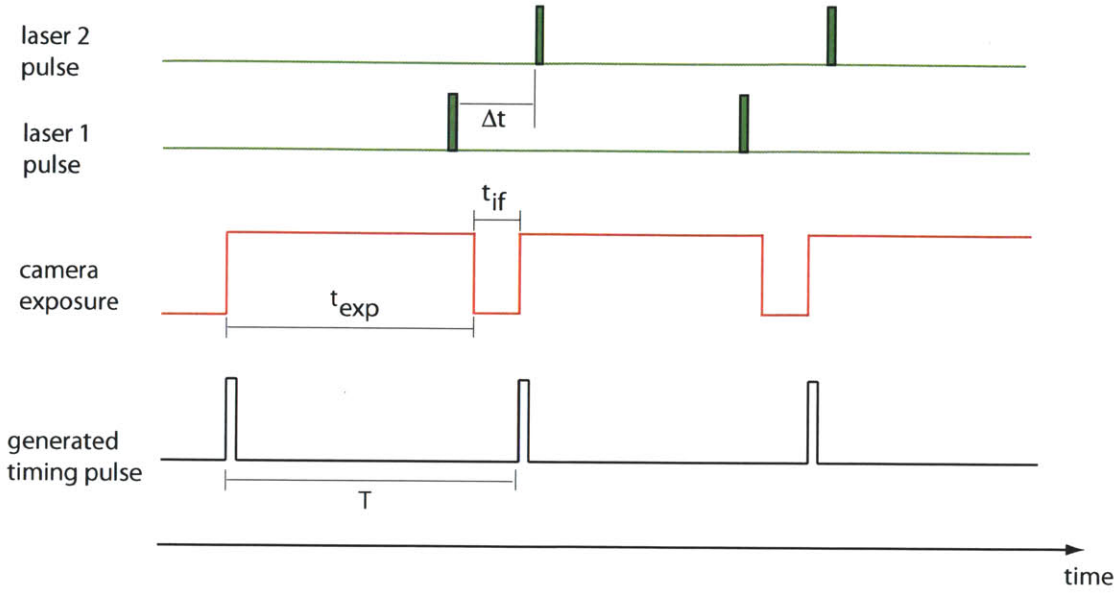


Figure 3-3: Frame-straddling PIV timing diagram for low-speed SAPIV camera array.

frame exposure and a second laser at the beginning of the next exposure, thereby generating a small time gap, Δt, between images. The shortest time scale of the flow field that can be measured is limited by Δt and therefore ultimately limited by the interframe time. For the Flea camera array, the minimum interframe time that worked reliably without dropping frames was 250μs, which is short enough to provide resolution for most water flows.

A final practical consideration for the low-speed array involves mounting the cameras on a frame that allows for enough degrees of freedom such that different volume size can be viewed with different camera arrangements. Figure 3-4 shows the frame used for the low-speed array. Aluminum 80/20® members provide the structure of the frame. Each camera is bolted to an aluminum L-bracket which is attached to one of three cross members, and thus has two degrees of freedom:

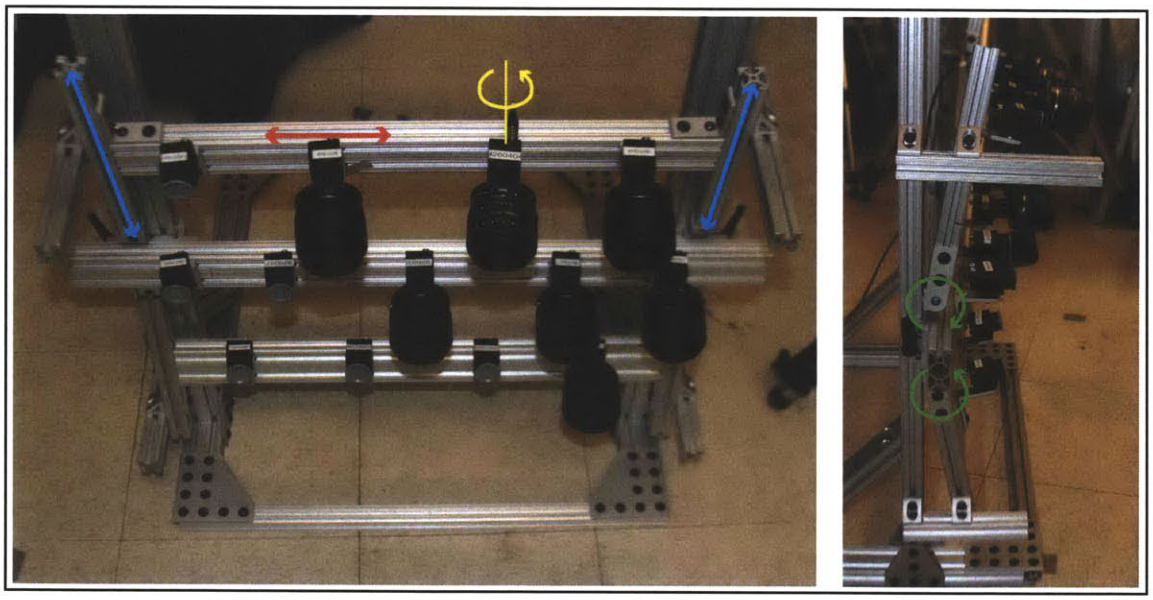


Figure 3-4: Degrees of freedom of the cameras and 80/20® frame.

rotation about the bolt axis and translation along the cross member, shown by the yellow and red arrows in Figure 3-4, respectively. Also, each cross member can be translated along the rails to which it is attached (shown by the blue arrows). Finally, the attachment rails for the top and bottom cross members can be rotated to angle the cameras, as shown by the green arrows. From experience, this frame design affords enough freedom of movement to spread the cameras out in a desired array configuration and overlap the fields of view (FOV) of all cameras.

Typically, alignment of camera FOVs involves placing a target in the center of the volume of interest and moving the cameras until the target appears in the center of the image. Each camera is then focused by fully opening the aperture (smallest f-number) to reduce the depth of field (DoF) and focusing on the target. Next, the aperture is partially closed (larger f-number) and the target moved to the front and back of the volume to ensure that the entire volume lies within the DoF. If the target

ever appears out of focus, the aperture is closed further.

3.2.2 High-Speed

The high speed camera array, shown in Figure 3-5, consists of nine Photron cameras. High speed cameras offer far more capability than low speed machine vision cameras for at least an order of magnitude higher cost. Among the advantages offered by the high speed cameras is full temporal resolution (up to $O(1000$ Hz$)$), improved sensitivity and a larger format image sensor. Also, the Photron cameras contain at least 2 GB of on-board memory, which removes the need for the careful data flow considerations associated with the low speed camera array. Data is downloaded from each camera post-capture via gigabit ethernet. Camera synchronization is accomplished by setting one camera to be the "master camera" and outputting a timing pulse from the master to the other cameras in the array. This camera array is used for the plunging jet experiment described in Chapter 5.

3.3 Comparison of SAPIV and Tomographic-PIV

This section draws further comparison between SAPIV and Tomographic-PIV based on the work described in Chapter 2. First, a comparison of the computational performance of each algorithm is discussed, along with the ability (or lack there-of) of each algorithm in dealing with partial occlusions in the flow. Then, the amount of "ghost particles" present in the reconstructed particle fields is evaluated.

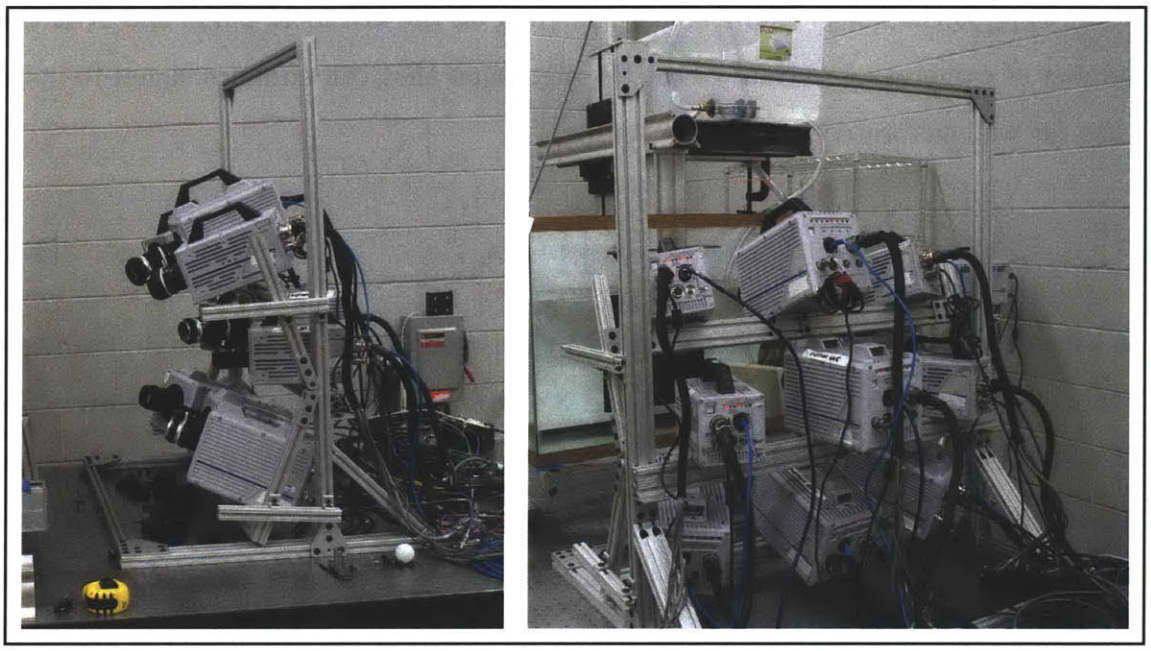


Figure 3-5: Array of high-speed Photron cameras.

3.3.1 Algorithm Performance

Some subtleties of the SA algorithm along with several similar Tomographic-PIV algorithms are now discussed. First, the computational performance of several algorithms is presented. After discussing the algorithms, comparisons of computational performance for typical experimental configurations highlights the relative efficiency of the SA method. Then the ability (or lack there-of) of each algorithm in dealing with partial occlusions in the flow is discussed.

As presented in Chapter 2, the formation of SA images is described mathematically as

$$I_{SA_k} = \frac{1}{N} \sum_{i=1}^{N} I_{FP_{ki}} \tag{3.3}$$

where $I_{FP_{ki}}$ is the image from camera i aligned on the kth focal plane and N is the number of cameras. A variant of the SA algorithm that can enhance signal-to-noise ratio for well calibrated images is given by

$$I_{SA_k} = \prod_{i=1}^{N} (I_{FP_{ki}})^n \tag{3.4}$$

as suggested by [5], where n is an exponent between 0 and 1. Henceforth, the algorithms making use of Equations 3.3 and 3.4 will be referred to as the additive SA (ASA) and multiplicative SA (MSA) algorithms, respectively. In order to determine the computational performance, the number of operations in each step of the processing is accounted for. With respect to the SA software, the number of operations associated with each processing step from application of the calibration to reproject images through the construction of the refocused volume is considered. To reproject images onto focal planes requires a transformation function derived from the camera calibration for each plane. Appendix A derives the transformation functions for a variety of calibrations; therefore, the number of operations corresponding to reprojection with pinhole camera models is simply stated here. To form the matrix H requires six operations, inversion of H requires 53 operations and applying scale and shift requires 27 operations resulting in 86 operations for each plane and each image. For all cameras and the entire volume, $86Nn_z$ operations are required, where N is the number of cameras and n_z is the number of voxels in the depth direction (or, equivalently, the number of focal planes in the volume). Each pixel must then be transformed to the focal plane, which requires nine operations for matrix multiplication and two more for conversion from homogeneous to actual coordinates. Therefore, application of the transformations to each pixel in all cameras for all focal planes necessitates $11p_{in}Nn_z$ operations, where p_{in} is the number of pixels in the

input images (assumed to be the same for all cameras). In general, the projections of the input image pixels onto the output image plane will not correspond exactly to the discretized output image, and the output image pixel intensity values must be interpolated from the projections of the input image, and described by Pratt [6]. A bilinear interpolation function defined by [6] as

$$\begin{aligned} I_o(i',j') &= (1-a)\left[(1-b)I(i,j) + bI(i,j+1)\right] + \\ &\quad a\left[(1-b)I(i+1,j) + bI(i+1,j+1)\right] \end{aligned} \tag{3.5}$$

is used to define the output image (I_o) where (a,b) is the displacement in x and y between the pixel (i',j') in the output image and the projected pixel (i,j) from the input image. For each pixel in all cameras for all focal planes, $14p_{out}Nn_z$ operations are required, where p_{out} is the number of pixels in the output image. Finally, with all images reprojected onto the n_z focal planes, application of either Equation 3.3 or 3.4 yields the refocused volume. For the additive SA algorithm, this final step requires $pn_z(N+1)$ operations to sum the intensity values at each voxel and divide by the total number of contributions, assuming now that $p_{out} = p_{in} = p$. For the multiplicative SA algorithm, $2pNn_z$ operations are needed to raise each contribution to the power n and multiply each resulting value. In summary, the total number of operations required to refocus one volume using the additive SA algorithm is

$$
\begin{aligned}
N_{o_{ASA}} &= 86Nn_z + 11pNn_z + 14pNn_z + pn_z(N+1) \\
N_{o_{ASA}} &= pn_z\left[(26N+1) + \frac{86N}{p}\right] \\
N_{o_{ASA}} &\approx pn_z\,(26N+1) = N_{vox}\,(26N+1)
\end{aligned} \tag{3.6}
$$

where $N_{vox} = pn_z$ is the total number of voxels in the reconstructed volume and the $86N/p$ is dropped because p is typically $\geq O(10^6)$. Furthermore, calculation of the transformations is typically only required once, as the same transformations are applied to each time step. The total number of operations corresponding to the multiplicative SA algorithm is

$$
\begin{aligned}
N_{o_{MSA}} &= 86Nn_z + 11pNn_z + 14pNn_z + 2pNn_z \\
N_{o_{MSA}} &\approx 27pNn_z = 27NN_{vox}
\end{aligned} \tag{3.7}
$$

Attention is now turned to the number of operations associated with several Tomographic-PIV algorithms; namely, the multiplicative algebraic reconstruction technique (MART) algorithm discussed by Elsinga et al. [7], a variant of MART with a multiplicative first guess (MFG) presented in Worth & Nickels [8] and two algorithms described in Atkinson & Soria [9], each making use of a multiplied line-of-sight (MLOS) input. In multi-detector imaging systems, it is assumed that the intensity on the detector is given by the line integral of a three-dimensional intensity field along the line-of-sight of the detector [9]

$$P_i = \int_{l_i} I(x, y, z) dl_i \tag{3.8}$$

where, for multiple cameras, P_i represents the ith pixel intensity. The problem of optical tomography involves solving the inverse of this equation; in other words, estimating the three-dimensional intensity field $I(x, y, z)$. Typically, variations of algebraic reconstruction techniques have been used for Tomographic-PIV because they are most suitable to a limited number of viewpoints [10, 9]. The MART algorithm is an iterative technique where the intensity of each voxel is updated according to

$$I_j^{k+1} = I_j^k \left(\frac{P_i}{\sum_j W_{ij} I_j^k} \right)^{\mu W_{ij}} \tag{3.9}$$

and k refers to the iteration, i refers to the pixel, j is the current voxel, μ is a relaxation parameter and W_{ij} is a weighting matrix defining the intersection of each pixel line-of-sight (LOS) with each voxel [7]. Equation 3.9 is evaluated for each pixel, and in order for a voxel to retain a high intensity value, the corresponding pixel(s) in each camera must "see" the voxel [7]. The implication of this algorithm is that if one camera has an occluded view of an object (zero intensity), then the object will not be reconstructed. This effect is in contrast with the SA algorithms, and will be discussed in further detail shortly. Also, in accordance with the integral of Equation 3.8, the summed projected intensity along the LOS should equal the pixel intensity; therefore, if one LOS intersects with many other LOS from the other cameras, the intensity will be divided among these voxels (and, in many cases, form ghost particles. According to [8], the total number of operations applied in MART is given by

$$N_{o_{MART}} = 4k_t N \bar{N_{vox}} (8\bar{L} + 4) \tag{3.10}$$

where $\bar{N_{vox}}$ is the average number of voxels considered in each iteration, $\bar{L}$ is the average length of each pixel LOS and k_t is the total number of iterations (typically five [7]). The large computational cost associated with MART stems from the need to iterate over every pixel and every voxel, and also the storage of W_{ij}, which is very large.

Worth & Nickels [8] proposed a method for reducing the number of voxels considered by initializing the iterations with a multiplicative first guess (MFG). The MFG involves multiplying the intensity of each pixel back-projected through the volume, and raising the result in each voxel to the power $1/N$. This is very similar to the multiplicative SA algorithm defined in Equation 3.4, with three notable differences. First, any pixel with an occluded view (zero intensity) will cause the object to not be reconstructed. Note that this is also the case for the multiplicative SA algorithm, except that an adjustment is made in the SA algorithm whereby zero pixels are replaced with small values, which do not reduce the reconstructed voxel intensity to zero when raised to the small exponent n [5]. In contrast with the SA methods that only require storage of the transformations between each image and each focal plane, the MFG method still requires storage of the large W_{ij} matrix. The third difference is that a small number of cameras is still used in the MFG method. Interestingly, Worth & Nickels determined that the reduction in accuracy in a 3D simulated velocity field was very slight when using simply the fields reconstructed by MFG compared to the fields reconstructed by MFG followed by MART. This indicates that the small improvement in accuracy as a result of MART may not be worth the huge computational penalty. In particular, if the initial reconstruction could be further improved

in accuracy - say, by the effective use of more cameras in conjunction with an SA algorithm - then the difference in accuracy would likely be very small and not worth the increase in computational cost. The number of operations of the MFG method is

$$N_{o_{MFG}} = N_{vox}(8N + 1) \tag{3.11}$$

plus the number of operations defined in Equation 3.10; the reduction in cost comes from iterating over fewer voxels.

A similar first guess proposed by Atkinson & Soria [9] called multiplied line-of-sight (MLOS) also reduces the computational cost of the tomographic algorithm. The approach is essentially the same as the MFG algorithm, except that the camera calibration equations combined with interpolation are used to project the pixel lines of sight into the volume, making the algorithm even more similar to the multiplicative SA. Again, an object must be seen in all cameras in order to be reconstructed using the MLOS approach. The number of operations of the MLOS step is $O(120NN_{vox})$ The number of operations of the tomographic reconstruction is reduced by considering only voxels containing non-zero values. Atkinson & Soria make use of two different iterative tomographic algorithms following the initial MLOS guess: a simultaneous algebraic reconstruction technique (SART) and a simultaneous multiplicative algebraic reconstruction technique (SMART). The number of operations for each algorithm are

$$N_{o_{MLOS-SART}} = (24N + 4)N_{n.z.vox} \tag{3.12}$$

$$N_{o_{MLOS-SMART}} = (24N + 3)N_{n.z.vox} \tag{3.13}$$

Table 3.1: Comparison of computational performance of several 3DPIV algorithms.

Algorithm	N_o	N	$N_{n.z.vox}$	$\bar{N_{vox}}$	$\bar{L}$	k_t	Initial N_o	Total N_o
ASA	$N_{vox}\,(26N+1)$	9	–	–	–	0	$8.81(10^9)$	$8.81(10^9)$
MSA	$27NN_{vox}$	9	–	–	–	0	$9.11(10^9)$	$9.11(10^9)$
MART	$4k_tN\bar{N_{vox}}(8\bar{L}+4)$	4	–	$1.1(10^7)$	30	5	0	$2.15(10^{11})$
MFG -MART	$N_{vox}(8N+1)$ $+4k_tN\bar{N_{vox}}(8\bar{L}+4)$	4	–	$6(10^6)$	20	5	$1.238(10^9)$	$8.00(10^{10})$
MLOS -SART	$120NN_{vox}$ $+(24N+4)N_{n.z.vox}$	4	$3.38(10^5)$	–	–	5	$1.80(10^{10})$	$1.81(10^{10})$
MLOS -SMART	$120NN_{vox}$ $+(24N+3)N_{n.z.vox}$	4	$3.38(10^5)$	–	–	5	$1.80(10^{10})$	$1.81(10^{10})$

where $N_{n.z.vox}$ is the number of non-zero voxels in the volume.

Table 3.1 compares the theoretical computational performance for each algorithm assuming a setup described in [8] that has $N_{vox} = 3.75(10^7)$ and $ppp = 0.05$. Four cameras are assumed for each tomographic algorithm, and nine cameras for each SA algorithm (in Chapter 2, nine cameras adequately reconstructed a field with $ppp = 0.05$ for the additive SA algorithm). The number of non-zero voxels ($N_{n.z.vox}$) is calculated assuming each reconstructed particle is represented by a 3x3x6 voxel patch, and the values of $\bar{N_{vox}}$ and $\bar{L}$ are given in [8]. The SA algorithms and the two Tomographic-PIV algorithms with enhanced initial fields each improve upon the computational performance of MART significantly. The mulitplicative SA algorithm requires 16.3, 6.1 and 1.4 times fewer operations than MART, MFG-MART and MLOS-S(M)ART, respectively. This indicates that the SA algorithms offer an efficient method of reconstructing intensity fields, particularly when the number of cameras must be increased to accommodate larger seeding density.

Perhaps the more important contrast between the SA and Tomographic types of

algorithms lies in the ability to deal with partial occlusions. While not as relevant for 3D PIV applications containing simply particle images, occlusions may be far more prevalent in multiphase (e.g., bubbly) flows or if the volume of interest contains objects (e.g., cylinder, propeller). In this case, as alluded to throughout the previous discussion, Tomographic algorithms will not reconstruct objects occluded from any one view. However, the additive SA readily reconstructs objects present in enough views to have sufficient signal over the background noise. Furthermore, by making an adjustment to raw images whereby zero pixels are replaced with small values, which do not reduce the reconstructed voxel intensity to zero when raised to the small exponent n, the multiplicative SA algorithm also enables reconstruction of partially occluded objects [5]. In Chapter 5, the multiplicative SA algorithm is applied to a densely occluded bubble field with excellent results.

Finally, because the Tomographic algorithms impose the constraint that the line integral of intensity must equal the pixel intensity, the intensity value assigned to particle locations is reduced when the instances of LOS intersection are increased (some intersections are true particles, some are ghosts particles [7]). Presumably, this is the reason why most Tomographic-PIV studies investigate volumes with a much smaller depth dimension than X-Y dimensions (e.g., [11]), as increasing the depth dimension increases the instances of LOS intersections. In contrast, the SA algorithms do not divide intensity among intersection points, but rather allows for the number of LOS intersecting each voxel to determine the relative magnitude of voxel intensity. This allows for the depth dimension of the volumes to increase. The next subsection concerns the amount of "ghost particles" generated in SA reconstructions.

3.3.2 Ghost Particles

In particle tracking velocimetry (PTV) [12], Tomographic-PIV [7] and SAPIV, "ghost particles" occur when the lines of sight from different particle images overlap in the reconstruction volume. In Tomographic-PIV, this results in a false positive particle [7] and similarly in SAPIV, some (usually not all) camera lines of sight will combine to result in a false particle with reduced intensity relative to the true particles. As reported by Elsinga et al. [13], for velocimetry techniques that make use of cross-correlation analysis, the ghost particles influence the measurement only when correlated in adjacent time steps. They showed this to occur in regions of small depth-wise flow gradient (i.e., small $\partial \mathbf{u}/\partial z$); the ghost particles caused underestimation of the local flow velocity. The results of their study indicate that not only do ghost particles result in local error in velocity estimation, but the error is dependent of local gradients, and thus non-uniform throughout the field. Novara et al. [14] proposed a method for reducing the number of ghost particles by estimating the velocity field using two or more reconstructed volumes, deforming one according to the velocity field and averaging to reduce the incoherent (or unpaired) ghost particles. Although they showed this method to improve the signal to noise ratio, coherent ghost particles in low gradient regions would tend to remain in the volumes and bias the error according to the study by Elsinga et al. [13]. Furthermore, the method of [14] is very computationally intensive as multiple iterations of the original Tomographic-PIV algorithm are required.

The amount of true and ghost particles present in the thresholded reconstructed volumes for all the simulated cases presented in Chapter 2 are calculated here. To determine the number of true and ghost particles reconstructed requires the identification of possible particles. First, each reconstructed volume is thresholded so that

the smallest 10% of intensity values are removed and the volume is then converted to binary. Then, particles are identified using a three-dimensional connectivity algorithm and the centroid of each particle is calculated. To distinguish between true and ghost particles, any particle with centroid within a 1.5 voxel radius of a known particle position is considered a true particle; this was the method used in [7]. Any particle not within a 1.5 voxel radius is considered to be a ghost particle, as are any duplicate particles. In other words, only one particle can be associated with each known location and all others are ghost particles. As was done for the simulations presented in Chapter 2, the outer 50 pixels (in X-Y) around the volume were discounted due to the effective reduction in field of view.

Figure 3-6 shows the results of the ghost particle study applied to the simulated $50 \times 50 \times 10$ $(mm)^3$ volume, with camera baseline $D = 0.4$, and varying number of cameras. Figure 3-6(a) shows the percentage of actual particles accurately reconstructed as a function of number of cameras, while Figure 3-6(b) shows the number of ghost particles expressed as a percentage of accurately reconstructed particles, versus number of cameras. Note that the y-axis on Figure 3-6(b) is log scale. As expected, adding cameras serves to reduce the number of ghost particles, to a very low percentage, even for high seeding density (e.g. 0.125 particles/pixel). The percentage of accurately reconstructed particles actually peaks between 9-13 cameras, then levels off with increasing camera number. This effect is most likely attributable to the fact that in the original simulations (Chapter 2), the threshold was adjusted to maximize reconstruction quality (Q) for each test, and therefore was not uniform for all tests. The percentage of accurately reconstructed particles is higher, but the number of ghost particles is also higher, and therefore the most accurate overall reconstruction still occurs for larger camera numbers. Note that for 25 cameras, the percentage of accurately reconstructed particles does begin to decrease, and this is

due to the same reasons that caused a decrease in reconstruction quality as discussed in Chapter 2.

The additional data points come from available data in the literature; E refers to data from Elsinga et al. [7] and A1, A2 and A3 refer to data from Atkinson & Soria [9]. In [7], the data come from application of the MART algorithm to a simulated $35 \times 35 \times 7$ $(mm)^3$ with ppp = 0.05 viewed with four cameras. In [9], the reconstruction is simulated in a 2D volume in voxel space of size $1000 \times 1 \times 200$ voxels and equivalent 3D seeding denisty of ppp = 0.0167 viewed with 3 cameras. The cases A1, A2 and A3 refer to application of one iteration of MLOS-SART, one iteration of MLOS-SMART and five iterations of MART, respectively. The cameras have similar angles with respect to the optical axis in all cases. For reference, the SAPIV volume size in voxels is $1000 \times 1000 \times 82$ (due to application of integer shifts described in Chapter 2) and in [7] the size is $700 \times 700 \times 140$ voxels. The lower number of voxels in the SAPIV case may reduce the number of estimated ghost particles somewhat, but this would need more investigation. If the SAPIV ppp = 0.05 curve were extended to fewer camera numbers, the percentage of ghost particles would still be an order of magnitude lower than found in [7]. However, the percentage of accurately reconstructed particles would be approximately 10% larger for the Elsinga data. Although no curve with similar seeding density is presented for the SAPIV data, the Atkinson data points clearly produce a larger number of ghost particles. If a reduction in ghost particles is the goal, clearly a larger number of cameras coupled with the SAPIV algorithm provides a good solution.

Several authors (e.g. [15, 14]) have cited a theoretical prediction for the ratio of ghost particles to actual particles defined as

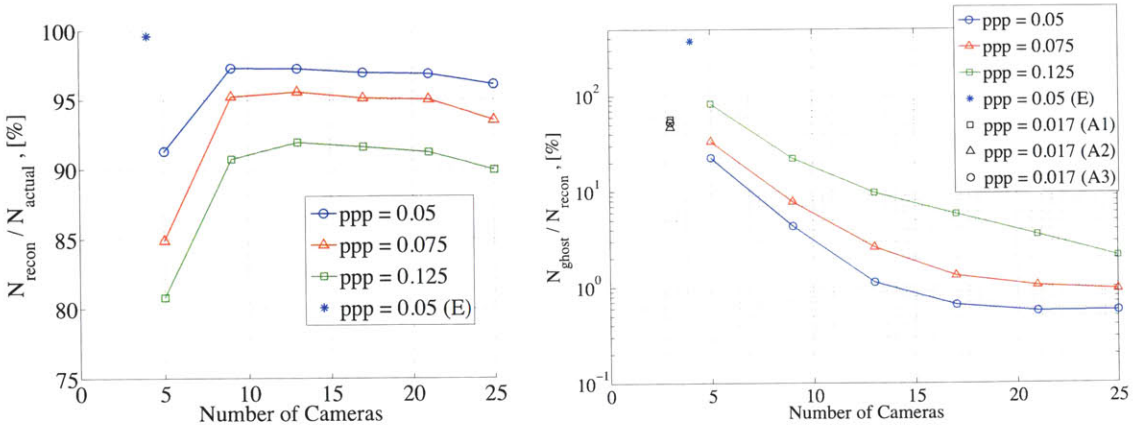


Figure 3-6: Percentage of actual particles accurately reconstructed as a function of number of cameras (a) and number of ghost particles expressed as a percentage of accurately reconstructed particles, versus number of cameras (b).

$$\frac{N_g}{N_p} = ppp^{N-1} \cdot A_p^N \cdot l_z \tag{3.14}$$

where N is the number of cameras, A_p is the particle image area in pixels and l_z is the volume depth dimension in voxels. Presumably, because SAPIV and Tomographic-PIV are closely related, the theory should apply to SAPIV data. The theoretical prediction using the parameters for the simulations plotted in Figure 3-6(b) are shown in Figure 3-7 to dispel the notion that this theory gives an accurate prediction of the ratio of ghost to actual particles. Clearly, the theory does not properly predict the dependence of number of ghost particles on camera number. Also, the prediction of values for the Tomographic-PIV setups are orders of magnitude of from the reported data. This brings into question whether this theory - often cited as a design tool in Tomographic-PIV - should be used at all.

Figure 3-8 summarizes the percentage of actual particles accurately reconstructed

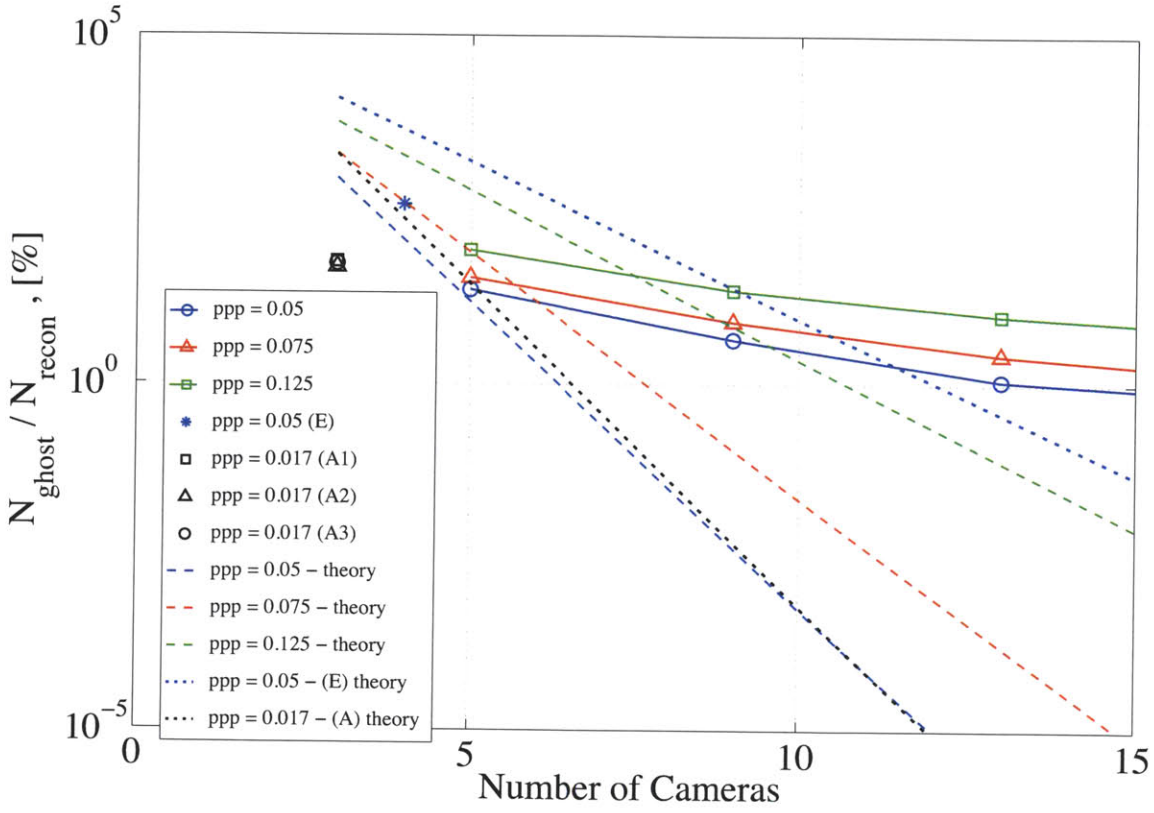


Figure 3-7: Percentage of actual particles accurately reconstructed as a function of number of cameras (a) and number of ghost particles expressed as a percentage of accurately reconstructed particles, versus number of cameras (b).

as a function of number of actual particles (Figures 3-8 (a),(c) and (e)) and number of ghost particles expressed as a percentage of accurately reconstructed particles (Figures 3-8 (b),(d) and (f)), versus number of actual particles in each simulated volume. For all cases, 25 simulated cameras were used. The percentage of particles accurately reconstructed decreases linearly with increasing seeding density, while the number of ghost particles increases non-linearly. This indicates that, beyond a certain seeding density, the noise will begin to rapidly wash out the signal. Figure 3-8 (f) shows what appears to be an anomalous data point for $D = 0.2$ in the $50 \times 50 \times 10$ $(mm)^3$ volume; the number of ghost particles actually peaks below the maximum seeding density. This is again likely due to using a threshold that retained more accurately reconstructed particles (as indicated in Figure 3-8 (e)), but retained too much noise.

To summarize, the SAPIV algorithm provides a relatively computationally efficient method of reconstructing volumes from many cameras. In turn, the use of many cameras dramatically reduces the instances of ghost particles and increases the number of accurately reconstructed particles, which should ultimately lead to more accurate velocity vector fields.

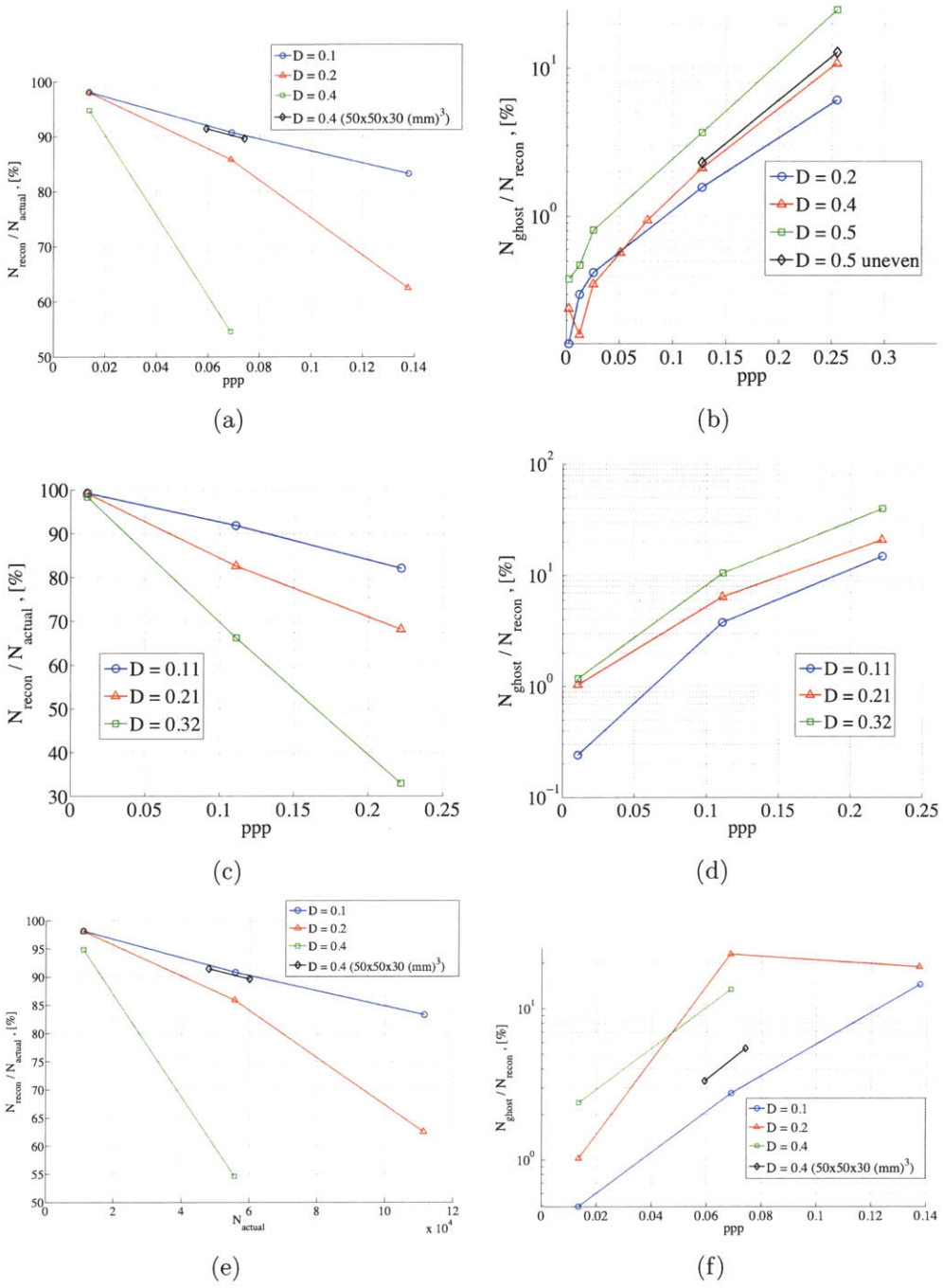


Figure 3-8: Percentage of actual particles accurately reconstructed as a function of number of actual particles in $50 \times 50 \times 10$ $(mm)^3$ volume (a) and number of ghost particles expressed as a percentage of accurately reconstructed particles, versus number of actual particles (b) in $50 \times 50 \times 10$ $(mm)^3$ volume. Plots (c-d) and (e-f) show the same quantities for the $50 \times 50 \times 50$ $(mm)^3$ volume and $100 \times 100 \times 100$ $(mm)^3$ volume, respectively. For all cases, 25 simulated cameras were used.

Bibliography

[1] R. J. Adrian. Particle-imaging techniques for experimental fluid mechanics. *Annual Review of Fluid Mechanics*, 23(1), 01 1991.

[2] http://www.ptgrey.com.

[3] Performance considerations. http://www.seagate.com, 2011.

[4] Streampix 5 digital video recording software. http://www.norpix.com.

[5] Roderick R. LaFoy. Personal Communication, 2010.

[6] William K. Pratt. *Digital Image Processing: PIKS Scientific Inside.* John Wiley & Sons, Inc., 4th edition, 2007.

[7] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.

[8] N. Worth and T. Nickels. Acceleration of tomo-piv by estimating the initial volume intensity distribution. *Experiments in Fluids*, 45:847–856, 2008. 10.1007/s00348-008-0504-6.

[9] Callum Atkinson and Julio Soria. An efficient simultaneous reconstruction technique for tomographic particle image velocimetry. *Experiments in Fluids*, 47:553–568, 2009. 10.1007/s00348-009-0728-0.

[10] A. C. Kak and Malcolm Slaney. *Principles of Computerized Tomographic Imaging.* Society of Industrial and Applied Mathematics, 2001.

[11] Gerrit Elsinga, Bernhard Wieneke, Fulvio Scarano, and Andreas Schrder. Tomographic 3d-piv and applications. In *Particle Image Velocimetry*, volume 112 of *Topics in Applied Physics*, pages 103–125. Springer Berlin / Heidelberg, 2008.

[12] H. G. Maas, A. Gruen, and D. Papantoniou. Particle tracking velocimetry in three-dimensional flows. *Experiments in Fluids*, 15(2):133–146, 1993-07-01.

[13] G. Elsinga, J. Westerweel, F. Scarano, and M. Novara. On the velocity of ghost particles and the bias errors in tomographic-piv. *Experiments in Fluids*, pages 1–14, 2010. 10.1007/s00348-010-0930-0.

[14] Matteo Novara, Kees Joost Batenburg, and Fulvio Scarano. Motion tracking-enhanced mart for tomographic piv. *Measurement Science and Technology*, 21(3), 2010.

[15] G. Elsinga, B. van Oudheusden, and F. Scarano. Experimental assesment of tomographic-piv accuracy. In *13th Int Symp on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, 26-29 June 2006.

Chapter 4

Camera Calibration

4.1 Introduction

One of the major challenges inherent in multi-camera setups is calibration. Camera calibration generally refers to the need to establish a relationship between points in the world viewed by the camera and points in the image plane of the camera. Moreover, in the context of quantitative imaging, the calibration must be a metric calibration such that measurements made in the images are true measurements of the world.

A conventional approach to camera calibration involves imaging a known set of points or object. Many researchers have taken the approach of traversing a precision machined calibration plate throughout a volume and taking images at each location(e.g. [1, 2]). However, this classic approach of using images of an accurately placed object to calibrate the camera becomes unreasonable when multiple cameras must be calibrated throughout an entire volume. This method places a burden on the user to very accurately align the calibration plate at several locations and even

still the calibration points may not truly coincide with the assumed known world points. Furthermore, any changes to the cameras (voluntary or not) nullifies the calibration.

This chapter presents a method for calibrating a multi-camera system that requires explicit knowledge of only a small number of world reference points. In particular, the method accounts for a typical situation encountered in experimental fluids studies, in which an air-glass-water transition exists between the cameras and objects of interest. Given enough cameras, points in 3D space are over constrained and the relative location of world points can be determined exclusively from images of the points in each camera. Subsequently, the points can be aligned with a world frame using a minimal amount of reference geometry. Cameras are modeled based on the physics of the imaging system, rather than an empirical approach, making the calibration more flexible. Because explicit knowledge of the world points is not required, the initial calibration can be updated from point correspondences at any time if any changes are made to the cameras (assuming some reference geometry is still available, or the calibrations have not deviated too far from the true calibration; this will be discussed in detail).

Several researchers in the computer vision community have investigated the problem of calibrating cameras with unknown world points, primarily for imaging systems where the camera is assumed (with good accuracy) to have a single viewpoint (i.e., pinhole camera models) [3, 4, 5, 6, 7, 8, 9]. As will be shown, direct application of a pinhole model to the imaging system with an air-glass-water transition is inaccurate because the system ceases to have a single viewpoint (this was also discussed in [10]). Nonetheless, calibration methods of this type provide a useful framework for the model developed later in this chapter. Zhang [11] developed an algorithm for calibration of a single camera using images of a planar grid at different arbitrary

locations. The procedure returns the intrinsic camera parameters and is simple to implement; however, only the extrinsic parameters defining the relation of the camera to each plane are calculated. In order to make accurate measurements, the location of the camera(s) relative to a Euclidean world frame is required. Sturm [12] describes a method of self-calibration using the motion of a single camera, which is not as relevant here.

Of more interest are the multi-camera self-calibration methods, which involve solving for camera parameters and world point locations using only images of the world points in many cameras. Hartley & Zisserman [4] describe the general framework for the multi-camera self-calibration problem with pinhole cameras, and many other authors have solved the pinhole self-calibration problem in a variety of ways [3, 5, 6, 7, 8, 9]. Svoboda et al. [9] developed a particularly useful mutli-camera self-calibration method and Matlab code [13] which involves solving the problem as outlined in [4], but allows for point correspondences to be missing in some views. The calibration provides cameras and world points (both previously unknown) in a Euclidean coordinate system that is within a similarity transform of a reference world coordinate system. Some known reference geometry must be supplied to align cameras and world points with the reference coordinate system; this is the case for all self-calibration codes for the traditional pinhole camera models [3, 5, 6, 7, 9]. However, the reference geometry is much easier to provide than accurately controlled calibration points throughout an entire volume as far fewer points are required.

The first step in a multi-camera calibration is to establish an appropriate model for the cameras. As mentioned previously, most existing auto-calibration techniques assume the pinhole camera model, which is accurate when all world scenes and cameras are in the same medium. The technique presented here accounts for the typical situation encountered in experimental fluids studies, in which an air-glass-water tran-

sition exists between the cameras and objects of interest. A classic pinhole model is assumed for the cameras, but the refraction at the interfaces is accounted for. Other authors (e.g. [14]) have used empirical models (namely, third order polynomials) to map planes in space to images to account for the non-linearity introduced by refractive interfaces, and subsequently reduced the error in the mapping functions through an iterative procedure. However, this method pre-supposes relatively well controlled placement of calibration planes and is not as flexible as a physics based modeling approach. The refractive model developed herein incorporates (up to) the 11 parameters of a pinhole model (6 exstrinsic, 5 intrinsic) and assumes a planar wall with accurately measured thickness. However, the wall thickness and geometry could be left as free parameters in the model. The cameras and points are initialized either from educated guesses based on knowledge of the setup, or via a linear auto-calibration commonly applied in computer vision settings [9]. As will be shown, the initial guesses are not required to be very accurate. Then, the world coordinates of each point are adjusted to minimize the summed reprojection error across all cameras using a non-linear minimization procedure. Using the new world points, the camera model parameters are adjusted to minimize the summed reprojection error across all points. This procedure is iterated until the reprojection error is reduced to an acceptable level.

This chapter first discusses a framework for calibrating multi-camera systems for a general relationship between world and image points. The framework allows for choice of camera model based on the physics of the system. Then, the details of the classic pinhole camera model are presented, followed by the description of the refractive model. Results are presented from a thorough study of the multi-camera refractive model calibration using a simulated array of cameras. Comparisons are drawn between the accuracy of the refractive model and direct application of a pin-

hole model. Also presented is the accuracy and convergence rate of the calibration procedure as a function of the number of cameras, error in the image point measurements and error in the initial conditions. Finally, the method is applied to real SAPIV experiments to demonstrate the accuracy, and the improvement in results over traditional calibration methods.

4.2 Framework for Calibration

In the most general sense, a camera images 3D points in the world onto a 2D image plane. We can represent the general action of a camera mathematically as,

$$\mathbf{u}_i^k = F\left(\mathbf{X}_i \; ; \; p^k\right) \tag{4.1}$$

which refers to the projection of the i^{th} point into the k^{th} camera. In equation 5.5, $\mathbf{u}_i^k$ is the 2x1 vector of the i^{th} image point coordinates in the k^{th} camera, $[u_i, v_i]^T$, $\mathbf{X}_i$ is the 3x1 vector of the i^{th} world point coordinates, $[X_i, Y_i, Z_i]^T$ and p^k is a set of parameters defining the model of the k^{th} camera. The specific form of the model and the parameters will be discussed in later sections.

In the conventional calibration approach, the world points are assumed known and the corresponding image points are measured. Each image coordinate provides an independent equation; therefore, we have $2N$ equations for b unknown parameters for each camera, where N is the number of points. This gives the requirement that $N \geq b/2$, but in practice more points than required are used and the parameters are found using a least-squares minimization due to measurement noise. In this case, the measurement noise is assumed to be Gaussian and is manifested as error prone

image point coordinates, and therefore the "best-fit" solution minimizes the sum of squared geometric reprojection errors. The geometric reprojection error is defined as,

$$e_r = ||\mathbf{u} - \hat{\mathbf{u}}||_2 = ||\mathbf{u} - F\left(\hat{\mathbf{X}}\ ;\ \hat{p}\right)||_2 \tag{4.2}$$

where $\mathbf{u}$ are the measured image coordinates, and $\hat{\mathbf{u}}$ and $\hat{p}$ are the reprojected image coordinates and camera parameters estimated in the minimization, respectively. Typically, a Levenberg-Marquardt nonlinear minimization algorithm is used to estimate the camera parameters [4].

Now consider the case where all camera parameters *and* world point coordinates are unknown. All that is known are the image point correspondences across cameras; in other words, it is assumed that the image of each world point can be paired with the corresponding images of the same point in each camera. In this situation, there are $2N \times M$ equations for all image coordinates in M cameras. The number of unknowns is $3N$ for the coordinates of the world points and bM for the camera parameters. Therefore, we have the requirement that

$$2NM \geq bM + 3N. \tag{4.3}$$

Rearranging this expression gives more insight into the requirements

$$N \geq \frac{bM}{2M - 3}\ . \tag{4.4}$$

Immediately it becomes clear that at least two cameras are required, which we knew intuitively. This expression also gives a requirement for the number of points that must be imaged given a certain number of cameras. Again, many more than the minimum number of points should be imaged to get a best-fit of the parameters and world points.

As will be discussed in the next section, linear models for multi-camera systems allow for the camera parameters and world points to be estimated via matrix factorization, followed by Euclidean stratification. This is the subject of auto-calibration in the computer vision community [3, 5, 6, 7, 8, 9]. A nonlinear minimization is usually only applied as the final step to model adjustment [4]. With a nonlinear camera model, a direct linear solution is not an option and an iterative nonlinear minimization procedure must be used with some initial starting point. The approach adopted here is essentially an interleaved bundle adjustment [4], except the model for the cameras can be nonlinear in general. The method is shown in figure 4-1. Initial estimates for the camera model parameters and world point coordinates can come from a linear auto-calibration algorithm, or simply from educated guesses based on the experimental setup (that are reasonable based on the volume of interest). As will be shown in section 4.4.2, the final results of the iterative algorithm are not affected by the initial conditions, although the convergence rate may be slower. However, a practical calibration method will be presented which enhances the accuracy and speed of the algorithm.

Once initialized, the world points are projected into each camera, and the reprojection error calculated for each point. The iterative procedure is carried out while some statistical measure of the reprojection error (e.g., mean over all points in all cameras) exceeds a desired tolerance, or until there is no reduction in the reprojection error. If the reprojection error exceeds the tolerance, the algorithm goes into a

loop for all cameras (1 to M), with the camera model parameters being adjusted to minimize the summed reprojection error due to all points. Then, a loop is run that uses a nonlinear minimization method to minimize the summed reprojection error in all cameras by adjusting the world point coordinates but keeping the camera models constant. The loop runs for all points (1 to N). Using updated cameras and points, the reprojection error is re-evaluated and the procedure continues.

The question of how to construct the camera model is addressed in the next two sections, followed by application of the calibration algorithm to simulated and real data to demonstrate the suitability for multi-camera setups.

4.3 Pinhole Camera Model

A large body of work has been published in the computer vision community centered around calibration of the pinhole camera model [4, 15, 16]. Much of this section follows directly from Hartley & Zisserman [4], but development of the pinhole model is required before discussion of the refractive model.

Figure 4.3 depicts a typical imaging configuration for a pinhole camera in an multi-camera setup. The camera center is, in general, removed from the origin of the world coordinate system and is angled arbitrarily. X-Y-Z is the fixed world coordinate system, X_{cam}-Y_{cam}-Z_{cam} is the camera coordinate system with origin located at the camera center, and x-y are the coordinates at the image plane (in physical units, not pixel units).

The pinhole camera model is developed assuming points are represented using homogeneous coordinates [4]. A homogeneous 4-dimensional vector defines a world point,

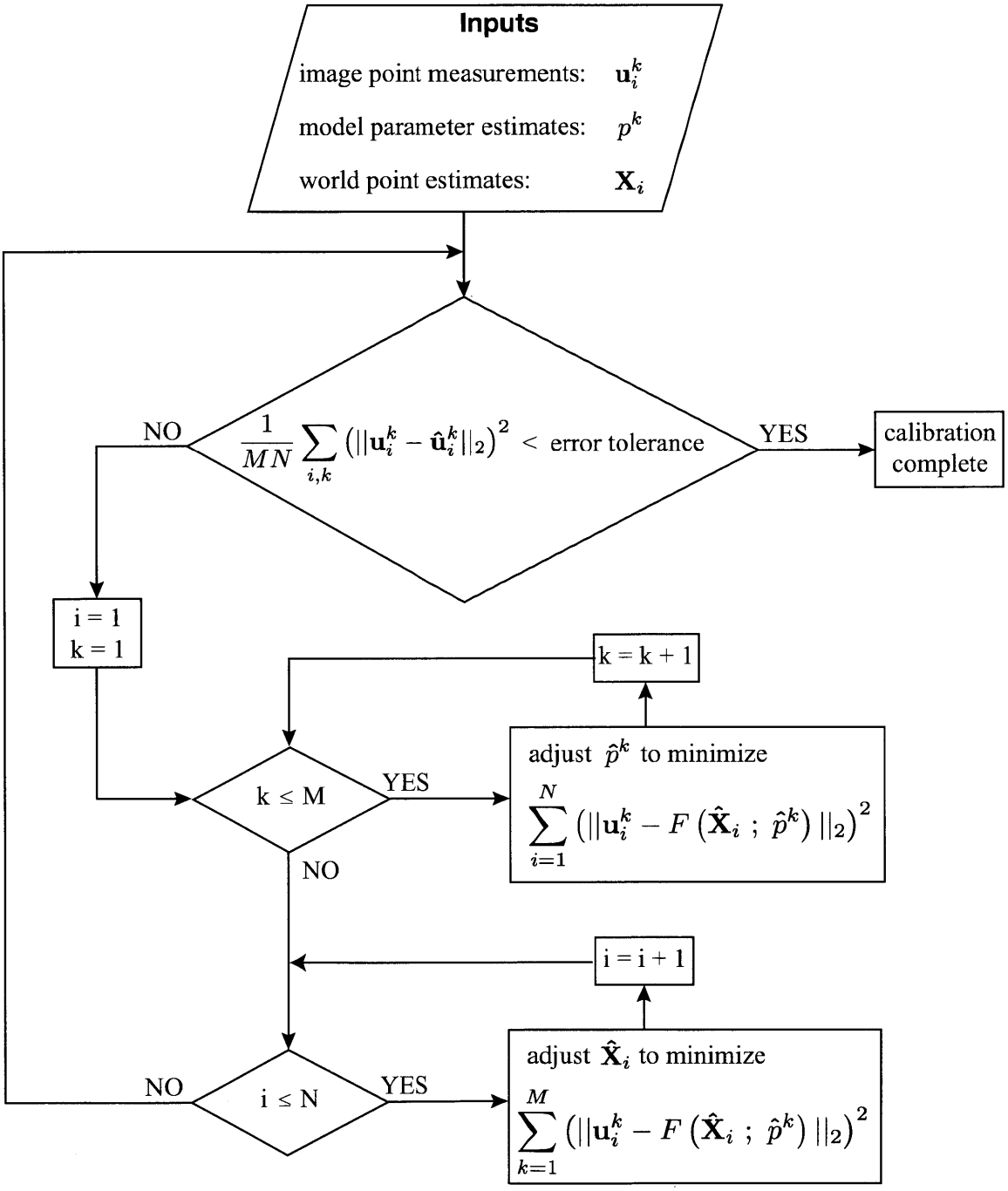


Figure 4-1: Flowchart of the iterative calibration procedure.

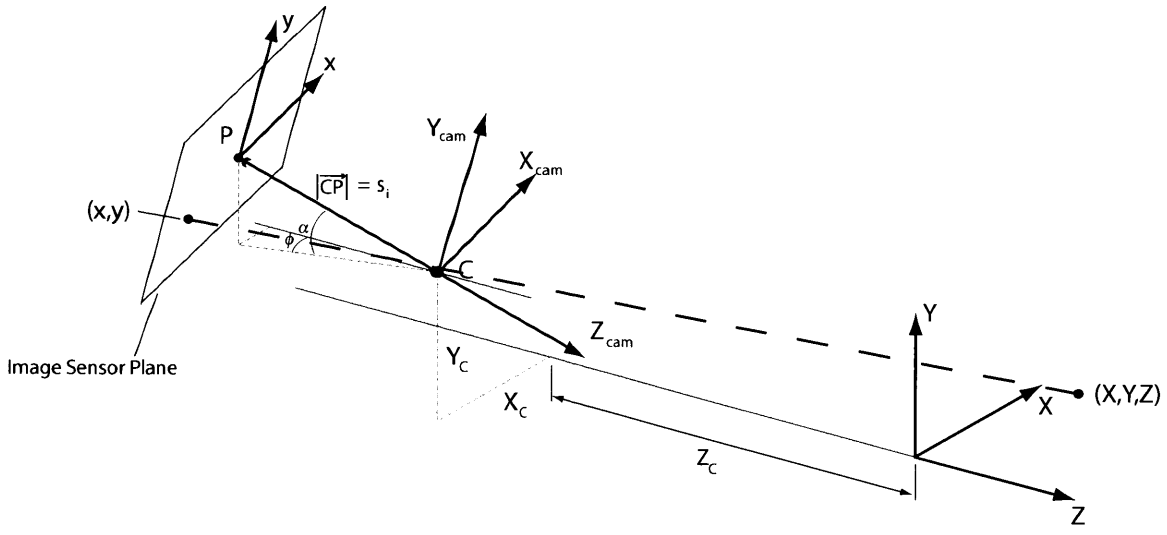


Figure 4-2: Coordinate systems for pinhole camera in air.

$$\mathbf{X} = [wX, wY, wZ, w]^T = [X_1, X_2, X_3, X_4]^T \tag{4.5}$$

where w is an arbitrary scale factor. The Cartesian coordinates of the point are found by

$$X = \frac{X_1}{X_4} \ , \ Y = \frac{X_2}{X_4} \ , \ Z = \frac{X_3}{X_4} \ . \tag{4.6}$$

Similarly, a homogeneous 3-dimensional vector defines a point on the image plane: $\mathbf{x} = [bx, by, b]^T = [x_1, x_2, x_3]^T$. Writing points in homogeneous coordinates allows for the projection of world points onto the image plane to be written mathematically as a matrix equation

$$\mathbf{x} = \underline{\underline{P}}\mathbf{X} \tag{4.7}$$

where $\underline{\underline{P}}$ is the 3x4 pinhole camera matrix, which can be written as

$$P = KR[I \mid -\mathbf{X}_{cam}] \tag{4.8}$$

where K is the 3x3 camera calibration matrix, R is the 3x3 rotation matrix, I is the 3x3 identity matrix, $\mathbf{X}_{cam}$ are the Cartesian coordinates of the camera center expressed in the world coordinate frame, and the double underlines have been dropped from matrix variables. The rotation matrix and translation of the camera center define what are referred to as the *extrinsic* parameters, because they describe the camera orientation in the world frame. In general, six extrinsic parameters define the camera: three rotation angles (ϕ, α, θ) and three coordinates of the camera center $(X_{cam}, Y_{cam}, Z_{cam})$. The rotation matrix can be defined in terms of rotations about three axes (according to Euler's rotation theorem); one possible set of rotation angles to take the world coordinate frame to the camera coordinate frame are shown in figures 4-3(a)-4-3(c). Each rotation can be written as a 3x3 matrix

$$R_1 = \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}$$

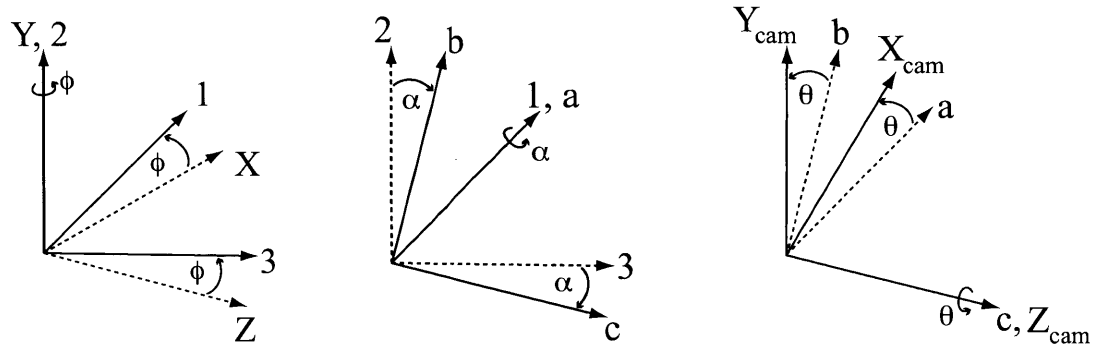
$$R_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha \\ 0 & -\sin\alpha & \cos\alpha \end{bmatrix} \tag{4.9}$$

$$R_3 = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The total rotation matrix is found by multiplying the individual matrices in the correct order

$$R = R_3 R_3 R_1 = \begin{bmatrix} \cos\theta\cos\phi + \sin\theta\sin\alpha\sin\phi & \sin\theta\cos\alpha & -\cos\theta\sin\phi + \sin\theta\sin\alpha\cos\phi \\ -\sin\theta\cos\phi + \cos\theta\sin\alpha\sin\phi & \cos\theta\cos\alpha & \sin\theta\sin\phi + \cos\theta\sin\alpha\cos\phi \\ \sin\phi\cos\alpha & -\sin\alpha & \cos\phi\cos\alpha \end{bmatrix}. \tag{4.10}$$

The *intrinsic* camera parameters define the camera calibration matrix, K



(a) R_1 (b) R_2 (c) R_3

Figure 4-3: Three rotations required to describe the orientation of the camera coordinate system.

$$K = \begin{bmatrix} -m_u s_i & s & m_u t_x \\ 0 & -m_v s_i & m_v t_y \\ 0 & 0 & 1 \end{bmatrix} \tag{4.11}$$

where s_i is the distance from the camera center to the image plane, s is a skew factor, m_u and m_v are scalar constants which convert physical units to pixel units, and t_x and t_y are offsets that account for a origin of the image coordinates that is not in the center of the image plane, as is often the case in practice and is shown in figure 4.3. The reason for using two scalars, m_u and m_v, is to account for the situation in which the aspect ratio of the camera pixels is not equal to one. Ignoring this rare case, and also excluding the skew factor which is not often necessary in practice, the camera calibration matrix becomes

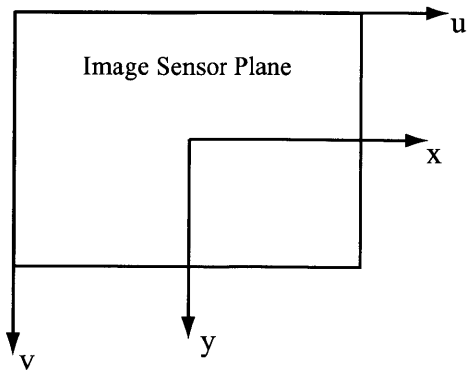


Figure 4-4: Coordinate systems for pinhole camera in air.

$$K = \begin{bmatrix} -b & 0 & u_o \\ 0 & -b & v_o \\ 0 & 0 & 1 \end{bmatrix} \tag{4.12}$$

where m and s_i have been replaced with one constant, b and we have also replaced mt_x and mt_y with u_o and v_o, respectively. Therefore, in most common applications found in the laboratory setting, there are 9 parameters (3 intrinsic, 6 extrinsic) that define a pinhole camera; however, as many as 11 parameters can be used in general.

4.4 Refractive Imaging Model

A very common configuration in fluids experiments is for the cameras to image from the air through a glass or acrylic tank full of water or another fluid. In this case, direct application of the pinhole model between world and image points is not valid as the refraction of light should be accounted for in the model of the system. As shown in Figure 4-5, if the rays emanating from the world points are drawn along the path taken in water (as would be assumed in a pinhole model), they do not meet

in a single point in the air. Even with exactly known world and image points, the resulting pinhole model would contain error as discussed by Treibitz et al. [10].

Before proceeding with a description of a model that does accurately describe this situation, it is useful to quantify how inaccurate the pinhole model is. Using the method for projection through refractive media that will be described in 4.4.1, points randomly located in a simulated volume of water are projected into cameras in one of the simulated arrays from Chapter 2. A five mm thick glass wall is also place between the water and the cameras. Application of the pinhole calibration to the exactly known world and image points yields an errant model. Figure 4-6 shows the histogram of reprojection errors for one camera from the array: the mean reprojection error (which in this case is also the true error) is 0.411 pixels and the standard deviation of the reprojection error is 0.269. The maximum reprojection error using the model described in this section (which accounts for refraction) is 2×10^{-8} pixels. As discussed in Chapter 2, the synthetic aperture refocusing technique begins to rapidly degrade in refocusing quality with true error in the mapping function above 0.4 pixels, particularly for PIV images. Therefore, even with zero measurement noise the pinhole model is insufficient for SAPIV.

In order to make the multi-camera approach relevant for fluids related studies, an accurate model which accounts for the refraction in the line of sight is required. Treibitz et al. [10] proposed a camera model that accounts for refraction but assumes the image plane is parallel to the container wall and requires placement of an object at accurately known depth. These constraints were acceptable for their application, but not for the present case where cameras can be in arbitrary position relative to the container wall and little or no requirement should be placed on locating calibration objects accurately. With these goals in mind, this section aims to develop a model suitable for accurate auto-calibration of multi-camera systems with refraction.

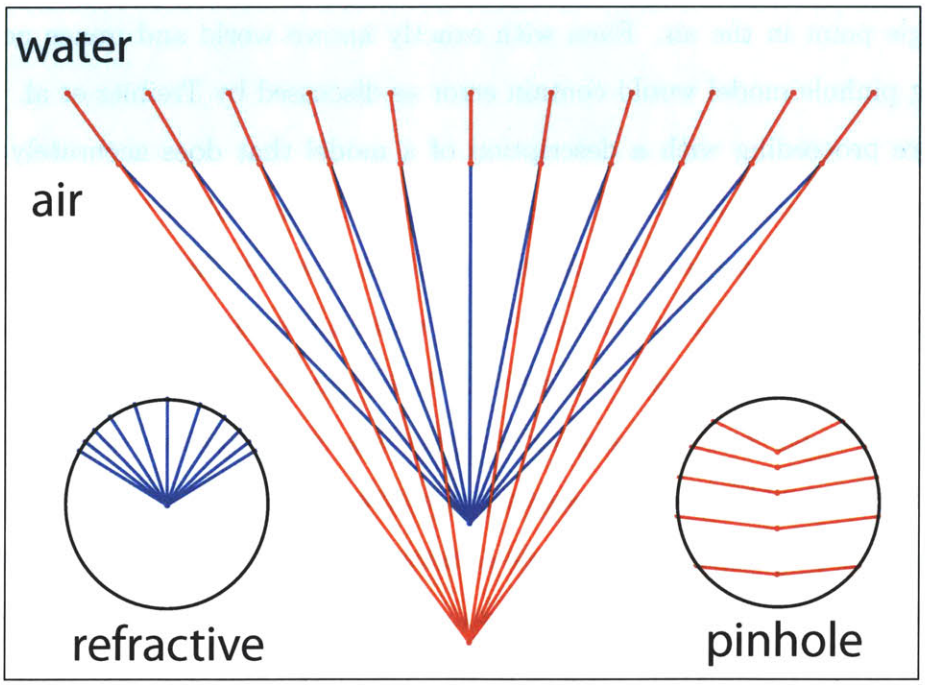


Figure 4-5: Schematic of imaging through changing medium. Modeling the refraction results in the (blue) rays correctly meeting in a single point. Application of a pinhole model is not valid as the (red) rays do not meet in a single point.

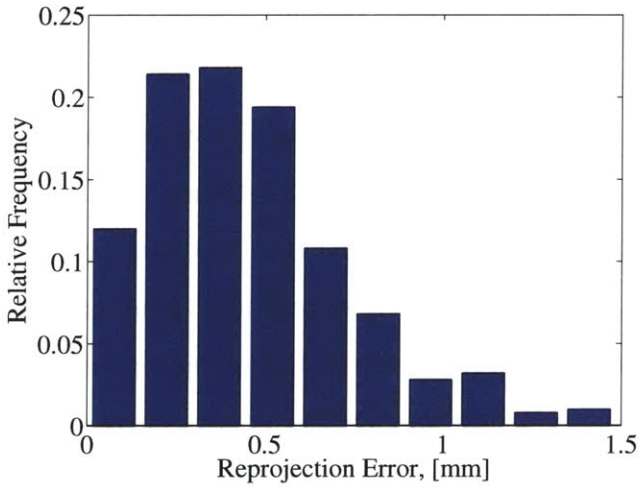


Figure 4-6: Reprojection error in one simulated camera emanating from application of the pinhole model to a refractive imaging system.

One assumption made in the model is that the wall of the tank is planar, which is true in many practical situations. However, the geometry could be altered or even left as an adjustable parameter if the situation required it. A three-dimensional view of the imaging system model is depicted in Figure 4-7(a), while Figure 4-7(b) shows the two-dimensional view in the plane of the ray. The angle of a light ray at each interface is governed by Snell's law

$$n_1 \sin\theta_1 = n_2 \sin\theta_2 \tag{4.13}$$

where n is the index of refraction of the medium (assumed known in the model) and θ is the angle the ray makes with the normal to the interface. Applying Snell's law at the air-tank interface, and defining the angles in terms of lengths shown in Figure 4-7(b) gives

$$\frac{r_A}{\sqrt{{r_A}^2 + {d_A}^2}} = \frac{n_2}{n_1} \frac{r_B - r_A}{\sqrt{(r_B - r_A)^2 + {d_B}^2}} \tag{4.14}$$

where r_A and r_B are the radial distances from an axis parallel to the Z-axis and passing through the camera center to the point of intersection of the chief ray with the front and back of the tank wall, respectively. The distance d_B is the thickness of the wall and is assumed to be known. The distance d_A is given as $d_A = Z_w - Z_C$, where Z_w is the Z-coordinate of the front of the wall and Z_C is the Z-coordinate of the camera center, all expressed in the world coordinate frame. The model assumes a set value for Z_w, which fixes the orientation of the X-Y plane. As will be discussed, the camera centers and point coordinates are free parameters, and thus fixing Z_w does not impact the accuracy of the model. Similar application of Snell's law at the tank-water interface yields the equation

$$\frac{r_B - r_A}{\sqrt{(r_B - r_A)^2 + {d_B}^2}} = \frac{n_3}{n_2} \frac{r_P - r_B}{\sqrt{(r_P - r_B)^2 + {d_P}^2}} \tag{4.15}$$

where the distance d_P is given as $d_P = Z_P - (Z_w + d_B)$, where Z_P is the Z-coordinate of the point.

If we assume for the moment that the world point and camera center coordinates are known, then equations 4.14 and 4.15 provide two equations for two unknowns (r_B and r_A). Each segment of the ray lies in the same plane, as shown in the end view in Figure 4-8. Therefore, the $X - Y$ coordinates of the point of intersection with the front of the tank wall can be defined in terms of the plane angle, ϕ, which is fixed by the camera and world point coordinates,

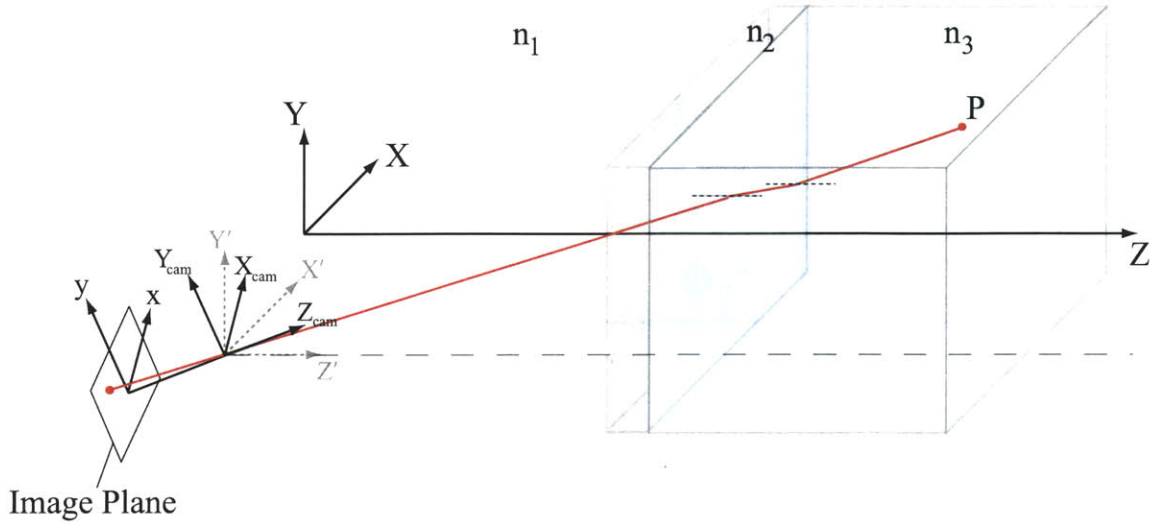
$$tan\phi = \frac{Y_P - Y_C}{X_P - X_C} \tag{4.16}$$

$$X_A = |r_A| cos\phi + X_C \tag{4.17}$$

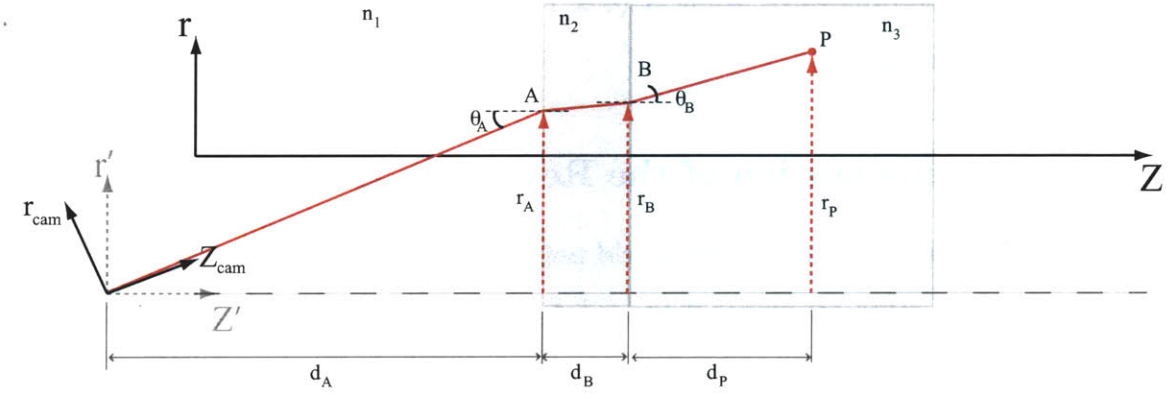
$$Y_A = |r_A| sin\phi + Y_C \tag{4.18}$$

Now that the coordinates of the point of ray intersection with the front of the tank wall are known (recall that $Z_A = Z_w$ is fixed in the model), this point can be projected into the cameras using a standard pinhole model, because there are no more changes in index of refraction in the path of the ray. Mathematically, the final step in projecting the world point into the camera is given by

$$\mathbf{x} = \underline{\underline{P}} \mathbf{X_A} \tag{4.19}$$



(a)



(b)

Figure 4-7: Three-dimensional view of the refractive imaging system model with planar wall (a), and two-dimensional view in the plane of the ray (b).

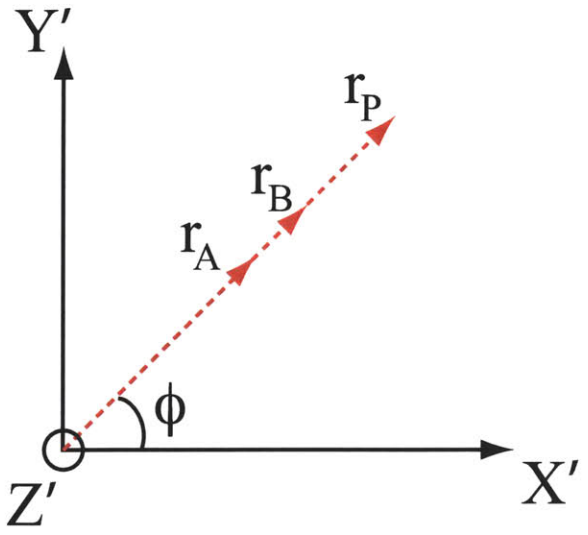


Figure 4-8: End view of refractive imaging model showing each segment of the ray lying in the same plane.

where $\mathbf{x}$ are the homogeneous image coordinates (3 x 1 vector), $\mathbf{X_A}$ are the homogeneous world coordinates of point A (4 x 1 vector), and $\underline{\underline{P}}$ is the pinhole camera matrix (3 x 4 matrix).

4.4.1 Implementation of the Refractive Imaging Model

Assuming, for the moment, that all world points and camera parameters are known, we need only to solve equations 4.14 and 4.15 for the two unknowns, r_B and r_A, followed by application of equations 4.16 - 4.18 for the coordinates of point A in the world frame. Equations 4.14 and 4.15 are non-linear and therefore must be solved numerically. The equations can be posed as a simultaneous root finding problem which can be solved using a mutli-equation Newton-Raphson scheme [17]. First,

equations 4.14 and 4.15 are rewritten as

$$f = \frac{r_A}{\sqrt{{r_A}^2 + {d_A}^2}} - \frac{n_2}{n_1}\frac{r_B - r_A}{\sqrt{(r_B - r_A)^2 + {d_B}^2}} \tag{4.20}$$

$$g = \frac{r_B - r_A}{\sqrt{(r_B - r_A)^2 + {d_B}^2}} - \frac{n_3}{n_2}\frac{r_P - r_B}{\sqrt{(r_P - r_B)^2 + {d_P}^2}} \tag{4.21}$$

The true solution simultaneously makes f and g zero. At each iteration, the unknown values are updated using

$${r_A}^{i+1} = {r_A}^i - \left[\frac{f\frac{\partial g}{\partial r_B} - g\frac{\partial f}{\partial r_B}}{\frac{\partial f}{\partial r_A}\frac{\partial g}{\partial r_B} - \frac{\partial f}{\partial r_B}\frac{\partial g}{\partial r_A}}\right]^i \tag{4.22}$$

$${r_B}^{i+1} = {r_B}^i - \left[\frac{g\frac{\partial f}{\partial r_A} - f\frac{\partial g}{\partial r_A}}{\frac{\partial f}{\partial r_A}\frac{\partial g}{\partial r_B} - \frac{\partial f}{\partial r_B}\frac{\partial g}{\partial r_A}}\right]^i \tag{4.23}$$

where i denotes values from the previous iteration. The partial derivatives in equations 4.22-4.23 are given in Appendix B. The convergence rate of the multi-equation Newton-Raphson scheme is very fast. Figure 4-9 shows a sample convergence plot for the solution to equations 4.20 & 4.21 for one point randomly seeded in a simulated volume. With an error tolerance of $O(10^{-8})$, the Newton-Raphson scheme typically converged in less than 10 iterations.

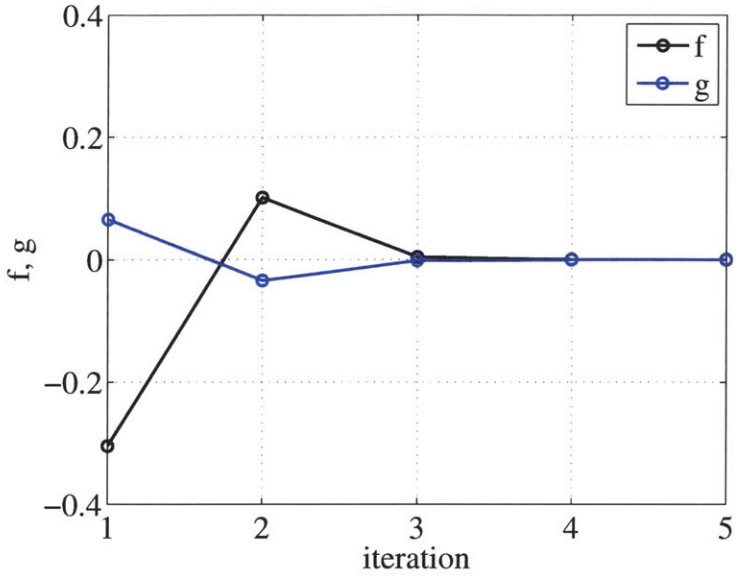


Figure 4-9: Convergence plot for solution to equations 4.20 & 4.21 using simultaneously Newton-Raphson scheme.

4.4.2 Multi-Camera Auto-Calibration of Refractive Imaging Systems

Attention is now turned to the auto-calibration of a multi-camera system in a refractive imaging environment, where the model discussed in the previous section applies to each camera. To reiterate, the problem we are faced with is solving for the world point coordinates and camera parameters ($\mathbf{X}_i$, p^k), given only image point correspondences ($\mathbf{u}_i^k$) across several cameras. In this section, an efficient method for generating calibration point correspondences is first discussed. Then, the simulated camera environment from Chapter 2 is used to assess the performance of the calibration procedure in a controlled, quantifiable manner. The accuracy and convergence rate of the calibration procedure as a function of the number of cameras, error in the image point measurements and error in the initial conditions is discussed. Finally,

application of the method to real data demonstrates the accuracy of the method.

The refractive imaging model is actually described by the same parameters as used in the pinhole model. However, explicit knowledge of the coordinates of the camera centers are required to solve equations 4.20 & 4.21. Therefore, rather than using the entries of the camera matrix (P) as the free parameters in the minimization, the extrinsic and intrinsic parameters are the free parameters. Insertion of the parameters into equation 4.8 allows for construction of the camera matrix, which can subsequently be used in equation 4.19 to project points into the cameras using matrix multiplication. Either a linear auto-calibration method or simply estimations of the parameters from measurements of the physical setup can be used to initialize the cameras. Any reasonable initialization provides good convergence rate, and does not affect the accuracy of the final model.

In finding the best-fit location of the world points, the three coordinates of each point could be left as free parameters and a loop run for N points to find the best fit coordinates. However, the overall convergence of the algorithm can be somewhat slow if a large number of calibration points are used. A more efficient method is to place a planar grid at arbitrary locations within the volume, which provides several benefits. First, planar grids can often be generated with relatively high accuracy for low cost (for example, by printing a checkerboard pattern). Second, instead of adjusting three coordinates for N calibration points, the 6 parameters (3 rotation angles, 3 translation distances) needed to define a plane in the world frame are used as the free parameters for each plane. Because each grid point is assumed to be known with high accuracy relative to a reference origin point on the grid, N world points are defined by $6K$ parameters for K planes, rather than $3N$ parameters. A typical number of plane locations sufficient for calibration is $K = 5$, giving 30 parameters to be adjusted; this number of parameters corresponds to 10 world point

coordinates, which is far fewer than normally used for calibration [9]. The overall convergence rate is therefore faster, because an extra constraint (that points lie on a plane) is physically imposed. Finally, this method is very amenable to practical implementation. Many open source codes exist for accurate corner detection on checkerboard grid images [18], and the grids are easy to produce on a high accuracy printer.

The grid should be placed at several plane locations to fill up the volume of interest with calibration points. Although knowledge of all grid locations is not required, some reference geometry with accurately measured coordinates in the world frame should be imaged by all cameras. All linear, pinhole model-based auto-calibration methods require alignment with known geometry in the world frame [9], but it is expected that the requirements for the refractive algorithm are less stringent. Nonetheless, some known points are required, and the reason will be discussed with the results of the calibration of the simulated refractive multi-camera system.

The coordinate system definitions for the simulated camera array correspond to those shown in Figure 4-7(a) & 4-7(b). A wall thickness of 5 mm is used for all simulations presented herein, and $Z = 0$ corresponds to the front of the wall. Several numerical experiments are performed to test various scenarios. For the first set of numerical experiments, 5 unchanging calibration planes with an 11×11 grid of points are used to establish point correspondences; the grid locations are shown in figure 4-10.

The variables in the first set of experiments are summarized in Table 4.1; each combination of the three variables is used. The variables $\sigma(e_u), \sigma(e_v)$ refer to the amount of noise imposed on the image point measurements. For each point in each camera, zero mean Gaussian white noise is added to the true u and v components of each point, where the standard deviation in e_u and e_v (the noise) corresponds

Table 4.1: Variables in the first set of numerical experiments for the simulated refractive camera calibration.

No. of Cameras	$\sigma(e_u), \sigma(e_v)$ [pixels]	$[\sigma(e_X), \sigma(e_Y), \sigma(e_Z)]/D$
3	0	0.001
	0.1	0.005
7	0.5	0.01
	1	0.1
13	2	0.2

to the values shown in Table 4.1. The image point error variable simulates the fact that in reality, image points are not measured perfectly, although corner finder algorithms can often measure checkerboard grid patterns to accuracies on the order of 0.1 pixels [18].

The other variable in the simulations is the amount of error imposed on the initial values of the world points. The third column in Table 4.1 expresses the standard deviation of the error added to the world point coordinates as a fraction of the maximum volume dimension, D. A zero mean Gaussian distribution again defines the noise in each coordinate. This variable is used to perturb the initial estimates of the camera parameters (as well as the world points) from the true values by finding the camera parameters using the errant world points. For the first set of numerical experiments the volume size is 50 x 50 x 50 mm^3, therefore the raw value of the standard deviation of the noise ranges from 0.05 mm to 10 mm. For each initial world point error level, the range of image point error levels is tested, with the same cameras being used. Different cameras are used for each world point error level to not introduce any bias error.

For the simulations, the camera parameters varied in the calibration procedure are: the scale (b) two rotation angles (ϕ,α) and the camera center coordinates

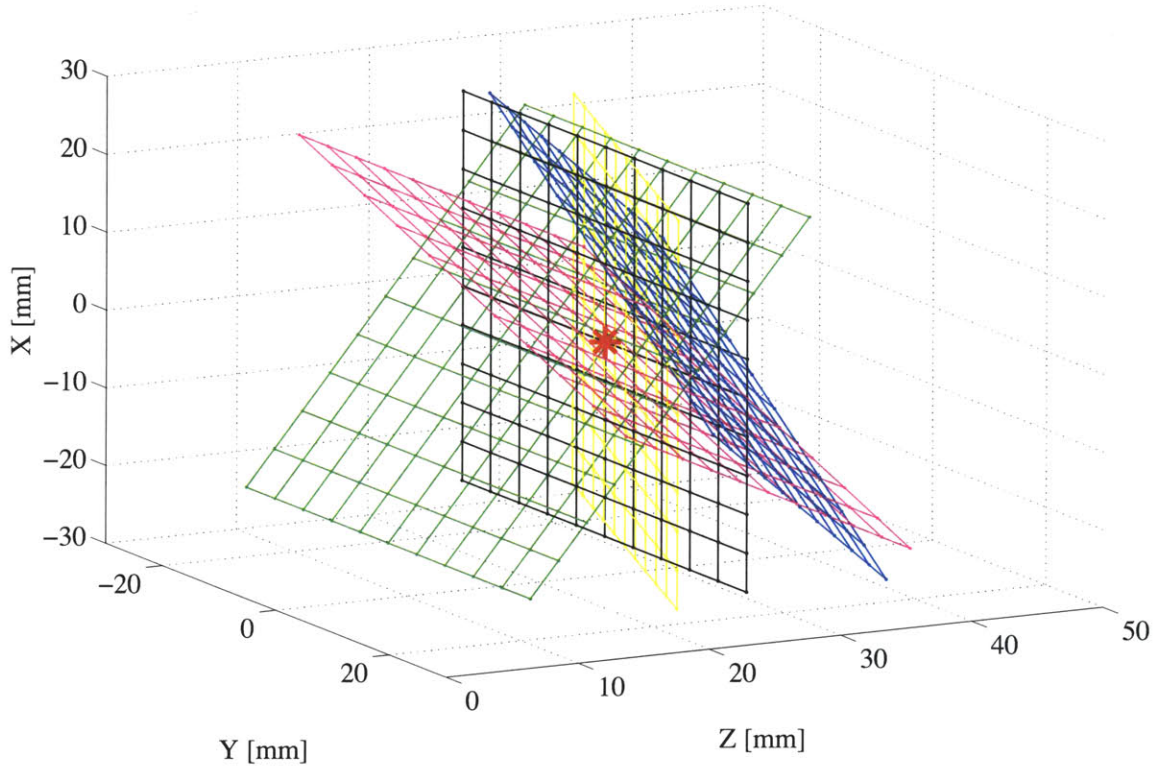


Figure 4-10: Location of calibration grid planes for the simulated camera calibration tests.

(X_C,Y_C,Z_C). Only two rotation angles are actually imposed in constructing the simulates array, and thus only two are left as free parameters; in the application to real data, all three rotation angles are left as free parameters. The image coordinate offsets (u_o,v_o) are assumed known based on knowledge of the image sensor size. The two rotation angles are initialized to zero and the scale is initialized to the true value; however, the scale will deviate from the true value in the initial iterations using errant world and image points, and thus initializing to the true value does not bias the final result. Matrix decomposition of the camera matrices found from linear calibration modeling methods (direct linear transform, [4]) yields initial estimates of the camera center coordinates. These initial estimates deviate significantly from the true values because the linear calibration method assumes a pinhole model.

Next, a Levenberg-Marquardt least-sqaures fitting method adjusts the camera parameters [4]; this corresponds to the k for loop in Figure 4-1. The sum of squared reprojection errors (i.e., Euclidian distance between measured and reprojected points) for all points within each camera serves as the objective function minimized by the Levenberg-Marquardt algorithm. This objective function is suitable when the errors are assumed follow a Gaussian distribution. Then, the i for loop in Figure 4-1 is carried out using a non-linear least-sqaures fitting method to adjust either the three coordinates of each world point or the six parameters of each plane to minimize the summed reprojection error across all M cameras. The auto-calibration procedure is repeated according to Figure 4-1 until some stop criterion is reached. Herein, when the largest value of mean reprojection error from all cameras is changing by less than 0.001 pixels, the procedure is stopped.

As a final step of the auto-calibration procedure, the world points are aligned with a known reference coordinate system. As alluded to earlier, some known reference geometry is required for the alignment. In the case of a linear, pinhole model-

based auto-calibration, the resulting world points are in a Euclidean frame that is within a similarity transform (scaling, translation and rotation) of the reference world coordinate system. This is due to a non-uniqueness of the solution to the linear auto-calibration problem. Because of the nature of the refractive model-based auto-calibration described herein, it is expected that the solution will be unique up to a translation of all points and cameras in X and Y and rotation about Z, because the wall is of infinite extent. Therefore, far less alignment with reference geometry should be expected compared to the linear auto-calibration methods. If a calibration grid provides the point correspondences, scaling of the world points is not necessary since it is included in the definition of the grid. From prior simulations, it was also observed that the world points resulting from the first refractive auto-calibration routine were only translated and not rotated from the true points. Therefore, alignment with the world reference frame for the simulated data requires only translation, which can be accomplished using only one accurately measured reference point.

The red star in Figure 4-10 is the single "known" reference point for the simulations. The translation is calculated by subtracting the known reference point coordinates from the calculated reference point coordinates resulting from the first refractive auto-calibration routine. The world points are then translated to the reference coordinate system, and the auto-calibration routine is repeated; the procedure typically converges within two iterations. In the application to real data described in Section 4.4.3, more reference points are required, and the reasons are expounded upon in that section.

The results of the simulations are presented in a variety of plots pertaining to the desired results of the auto-calibration procedure. The convergence rate of the entire procedure, final errors in reprojected image points and accuracy of the calculated world points constitute the metrics of interest in this study. Figure 4-11 shows the

convergence plots for the auto-calibration algorithm applied to 7 simulated cameras. For all plots, the standard deviation of error in the imposed image point measurements is 0.5 pixels. The initial error in world point coordinates increases from plot 4-11(a) to 4-11(e). The base 10 logarithm of mean reprojection error is plotted on the y-axis with iteration number plotted on the x-axis. For large initial world error levels, the initial mean reprojection error is quite large ($O(100)$ pixels for $\sigma(e_W)/D = 0.2$), but reduces rapidly to $O(1)$ pixels. For all initial world error levels, the solution converges in relatively few iterations, except for the case of $\sigma(e_W)/D = 0.1$. Although the large initial errors are reduced very rapidly, the convergence to the low error levels is slightly slow. This is likely due to the nonlinear fitting algorithms reaching areas of low gradient with respect to the parameters. Convergence plots in Appendix C for all other simulation tests show similar results.

Figure 4-12 shows the image point error summary for all seven camera simulations and the definition of each error is depicted in Figure 4-13 to facilitate explanation of the plots. The blue curves in Figure 4-12 refer to the mean initial image measurement error, which is imposed in the simulations. The red curves refer to the final mean reprojection error measure between the reprojected points and noisy "measured" points. The final true mean error - between the reprojected and true image points - is shown by the black curve, and could not be measured for real applications, of course. For each plot, y-axis contains the mean error and the x-axis represents the imposed standard deviation of the image point errors. The symbols on each curve correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2. As previously mentioned, different cameras are used for each initial value of world point error. Therefore, each of the seven plots (Figures (a)-(f)) shows the results for different cameras, but the cameras are not the same across all world error levels.

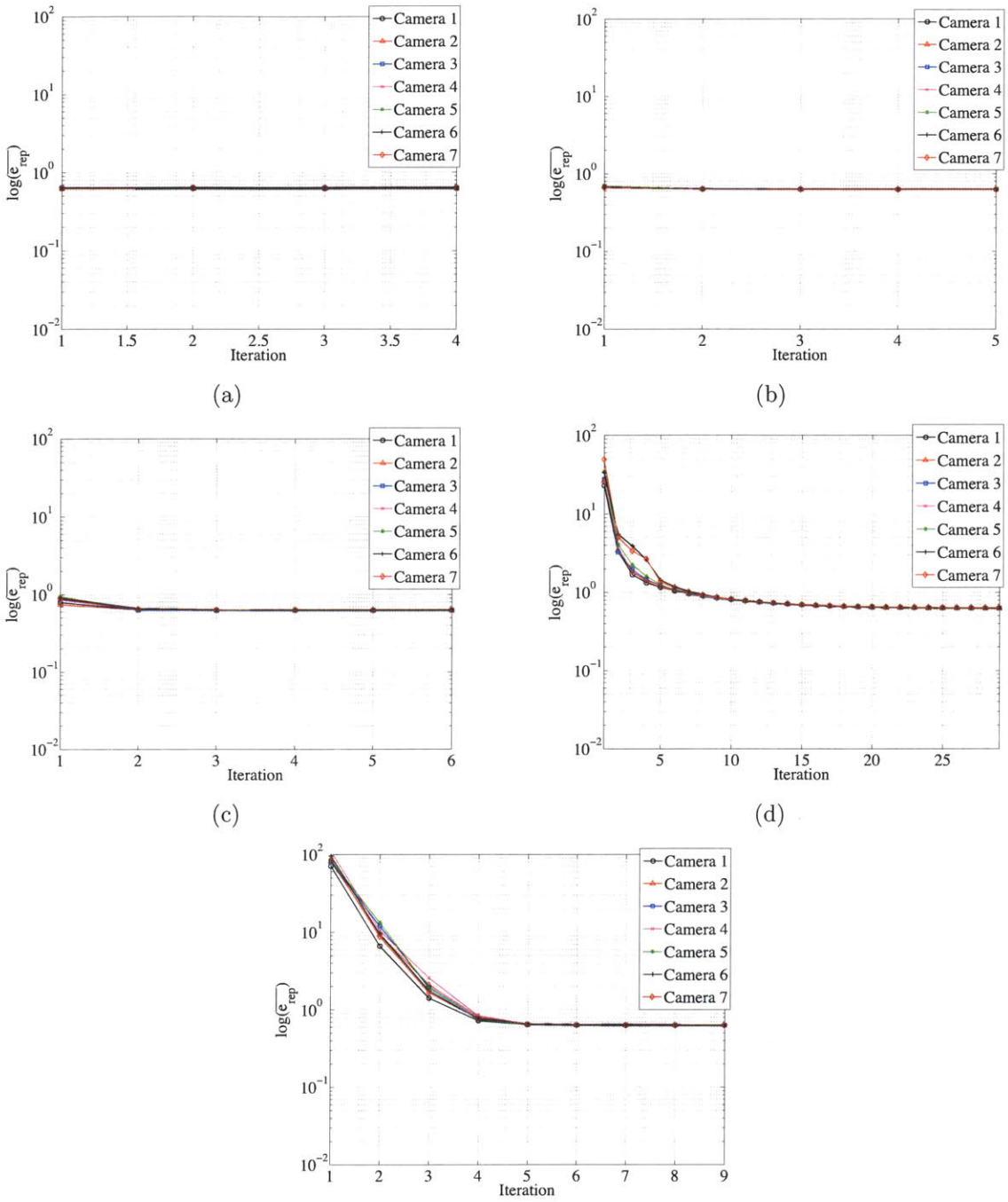


Figure 4-11: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras. For all plots, the standard deviation in the imposed image point measurements is 0.5 pixels. The initial error in world point coordinates increases from plot (a) to (e).

Appendix C contains the corresponding figures for the three and 13 camera studies (Figures C-16 - C-19). Immediately, we observe that the final mean reprojection error curves fall almost exactly on the mean initial image measurement curves. This indicates that the reprojected points do minimize the objective function, and therefore provide the best estimate of the true points in a least-squares sense. Furthermore, the black curves reveal that the true mean error between reprojected and true points is significantly smaller than the mean reprojection error ($\bar{e}_{true}/\bar{e}_{rep} \approx 10\%$), but increases with increasing imposed image point error. The initial value of world point error has very little effect on the final true mean error.

In conclusion, the most effective manner to reduce true error in reprojected image points is to accurately measure the image points in the first place. This statement is obvious. Less obvious is the fact that relatively large reprojection error does not necessarily indicate an inaccurate calibration. Even with mean reprojection error of 2.5 pixels, the true image point error is on the order of 0.25 pixels. This brings up a subtle point that appears to often be overlooked in the literature. The reprojection error is not itself a measure of the accuracy of the calibration, it is an objective function that we try to minimize. The accuracy of the estimated world point locations truly defines the accuracy of the calibration, which is discussed next.

Since the true location of world points is known in the simulations, we can determine the accuracy of the calibration by comparing the calculated and known world points. Figure 4-14 and 4-15 show the mean (4-14(a)-(c) and 4-15(d)-(e)) and standard deviation (4-14(f)-(h) and 4-15(i)-(j)) of the final error in each component of the world points divided by the volume dimension plotted as a function of the initial world point error level. The imposed image point error level corresponds to each row of plots. The final mean error level appears to be random and uncorrelated to the initial world point error. However, the standard deviation of the final world

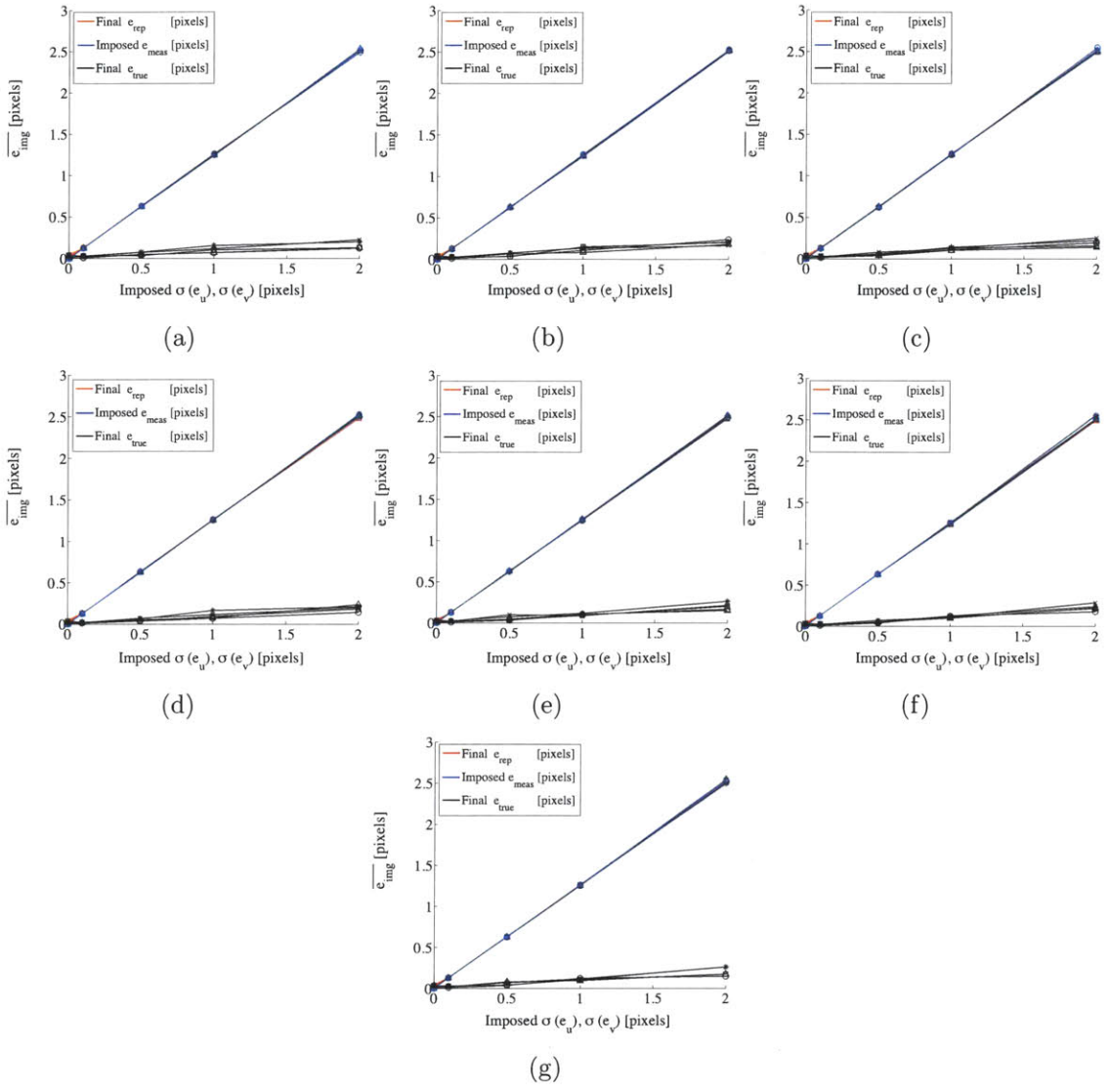


Figure 4-12: Final mean reprojection error after convergence of the auto-calibration procedure. The red and blue curves refer to the final mean reprojection error and the mean initial image measurement error, respectively; the final true mean error is shown in black. The symbols correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2.

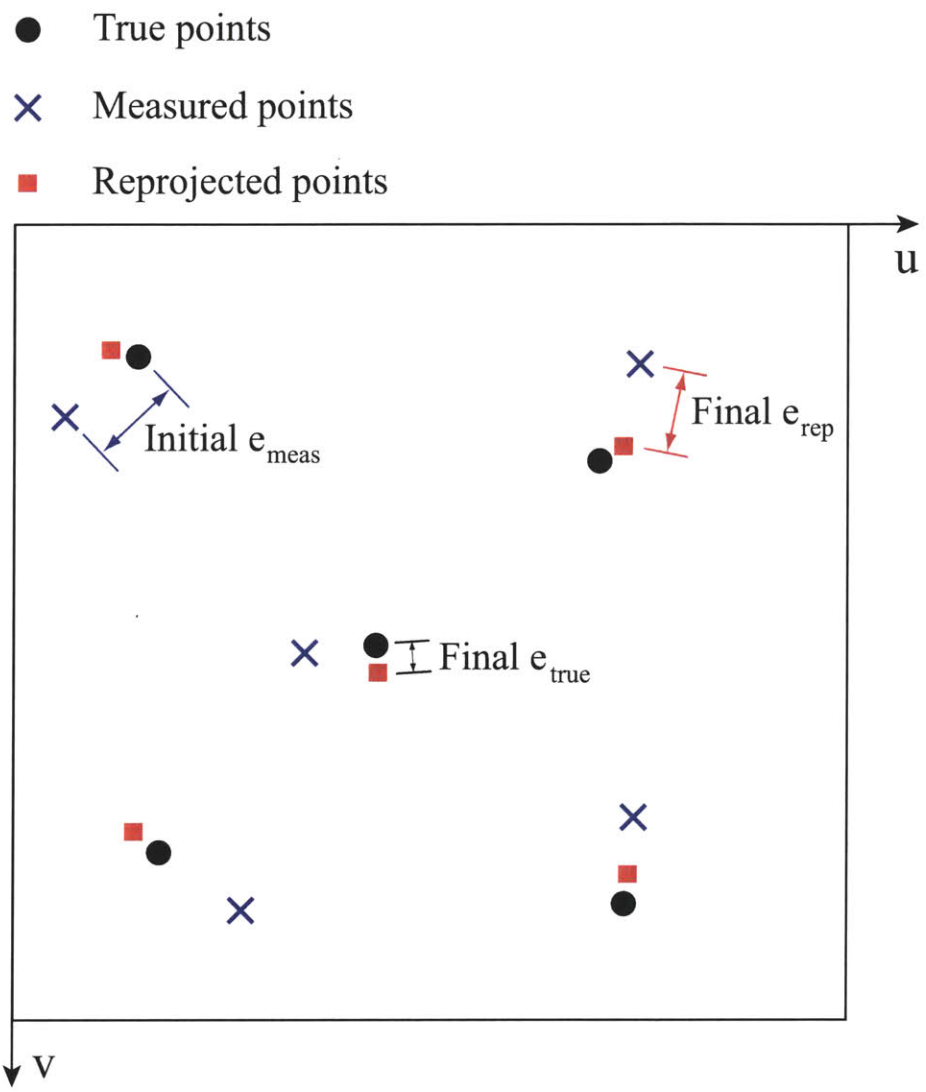


Figure 4-13: Definition of calibration errors summarized in Figure 4-12.

point error generally does increase with increasing initial error point level. To put the error in better context, consider the dimension of a pixel back projected from the center camera into the middle of the volume, d_{pix}; for the simulated camera array, $d_{pix} = 0.052$ mm. The magnitude of the errors are relatively small, with the maximum value of mean and standard deviation in the final error of $0.33d_{pix}$ and $1.17d_{pix}$, respectively, for the seven camera study with $\sigma(e_u) = \sigma(e_v) = 2$ pixels. In each case, the maximum error is in the Z component.

Nonetheless, it it hypothesized that the increase in final world point error with increasing initial world point error may be in part due to the algorithm settling into a local minimum. To attempt to mitigate this effect, it is proposed that the calibration algorithm be applied to randomly selected subsets of the cameras, similar to a random sample consensus (RANSAC) approach [19]. Each calibration yields estimates of the world point coordinates (or plane parameters) and if the difference between the world point estimates exceeds some threshold, the calibration is repeated using new subsets. The initial values of the plane parameters used for the next iteration are the average of the set of plane parameters estimated from the previous iteration. The net effect of this procedure should be to reduce any bias error that may occur when applying the calibration routine to a single set of cameras.

To test this approach, the simulated camera array with eight cameras is used, with the subset size equal to four cameras. The actual cameras composing each subset varies randomly for each iteration. When the standard deviation of the difference in all coordinates of the points estimated using the two camera subsets is less than 0.001 mm or the maximum standard deviation of the differences exceeds that of the previous iteration, the loop is stopped. The plane parameters are then taken as the average of the set of plane parameters estimated from the iteration providing the minimum value of standard deviation of differences between estimated world points,

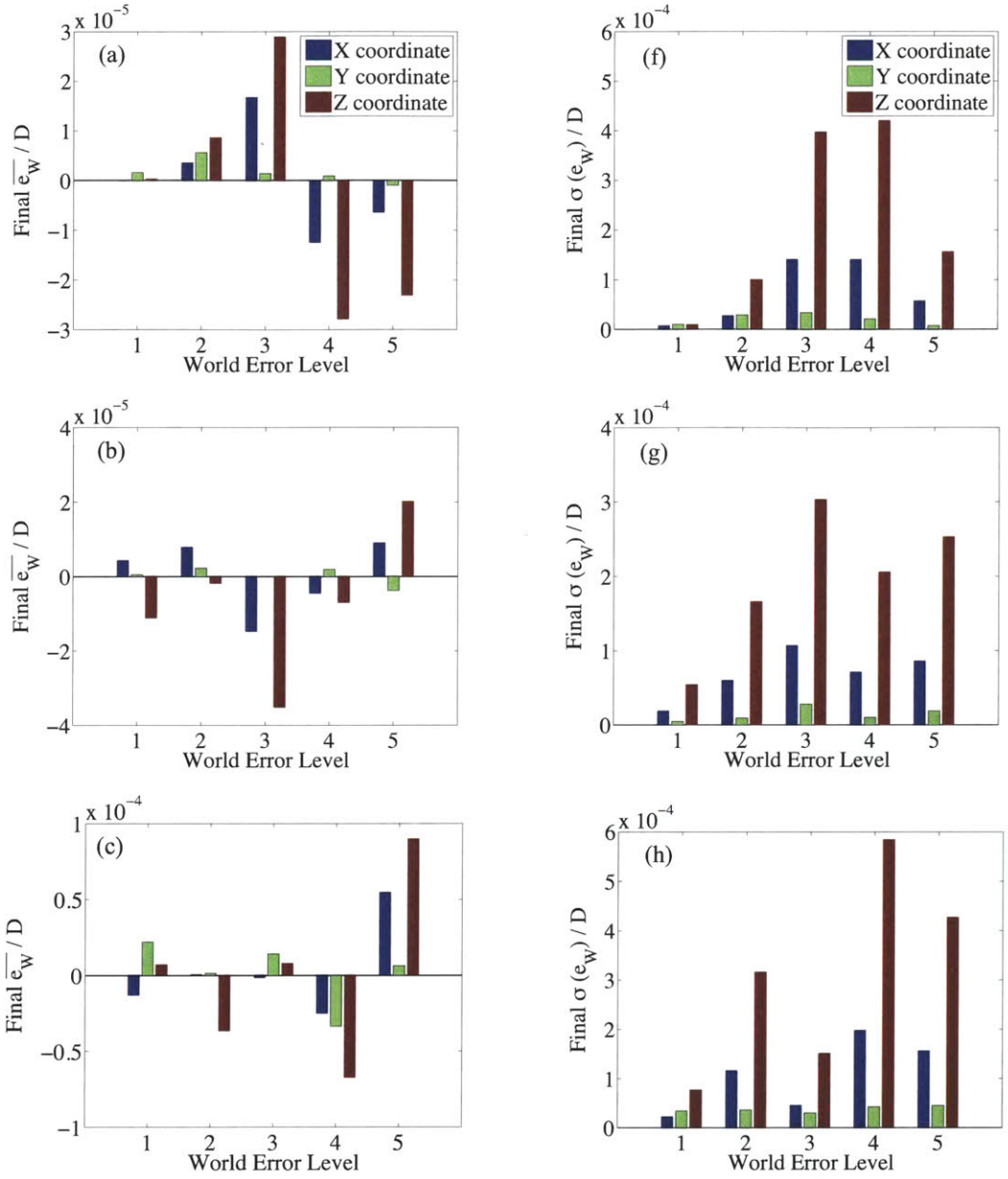


Figure 4-14: Mean (a)-(c) and standard deviation (f)-(h) of the error in world point locations after the final iteration of calibration procedure applied to 7 simulated cameras. Rows 1-3 correspond to imposed image points error levels of $\sigma(e_u) = \sigma(e_v) = 0$, 0.1 and 0.5 pixels, respectively.

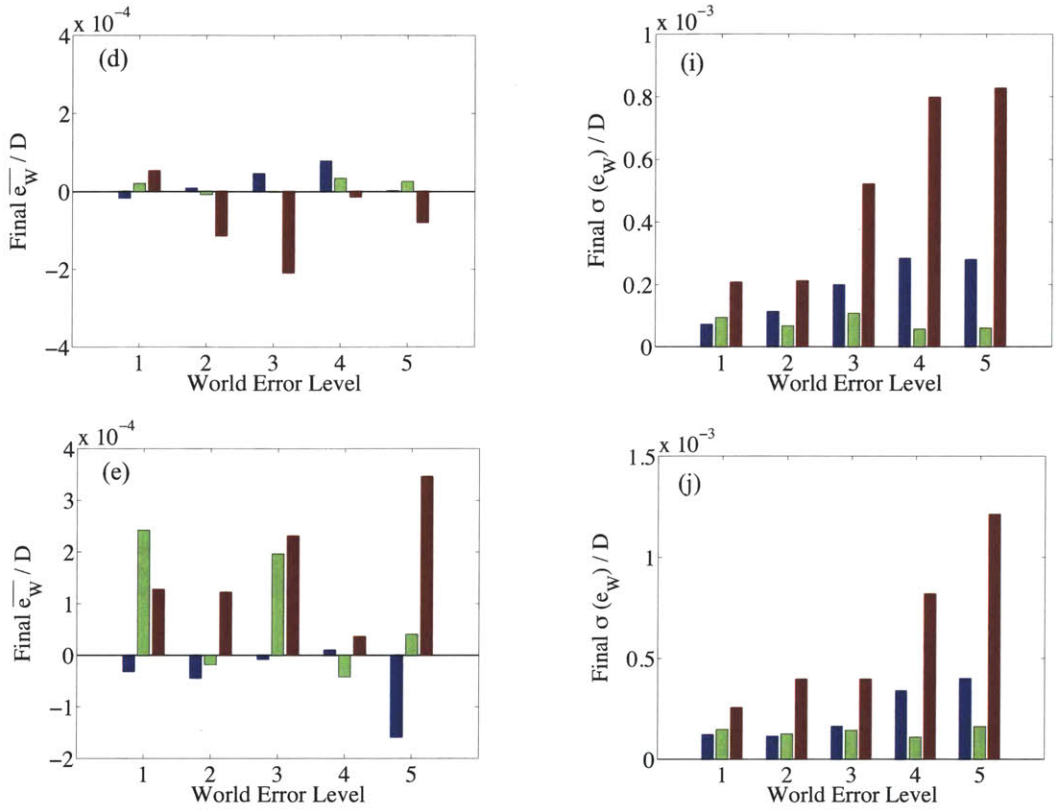


Figure 4-15: Mean (d)-(e) and standard deviation (i)-(j) of the error in world point locations after the final iteration of calibration procedure applied to 7 simulated cameras. Rows 1-2 correspond to imposed image points error levels of $\sigma(e_u) = \sigma(e_v) = 1$ and 2 pixels, respectively.

and the original calibration routine is then applied to all eight cameras. The results of this approach are summarized in Figures 4-16 and 4-17, which show histograms of the mean and standard deviation of world point error for results from the randomized as well as the original calibration method. As observed for the previous study, the mean world point error appears random and is typically an order of magnitude smaller than the standard deviation in world point error. In general, the randomized calibration procedure reduces the standard deviation in world point errors, and the effect is most pronounced for large imposed image point errors and initial world point errors. The increased accuracy comes at the cost of increased computational effort, however, and may not be necessary depending on the required accuracy for a given experiment.

Finally, to characterize the accuracy of the calibration throughout the entire volume, 500 random points are distributed within the volume and projected to image coordinates using the exact cameras. Gaussian noise is added to the image points in the same five amounts as used for the previous simulations. Five tests are carried out whereby the world point coordinates are estimated using the noisy image points and both the exact cameras and the cameras estimated from the corresponding auto-calibration. This offers a direct comparison of the accuracy of the refractive auto-calibration procedure in estimating world point locations since the most accurate estimation will come from using the exact cameras. Figure 4-18 shows the histograms of error in the Z coordinates for the estimated world points with initial world error level of $\sigma(e_W)/D = 0.2$ for all five image point error levels. The blue bins correspond to the difference between the known points and the points found using the estimated cameras and the red bins correspond to the difference between the known points and the points found using the exact cameras. The error distributions for the X and Y coordinates resemble that for Z, but have a smaller standard deviation. With the exception of the first histogram (corresponding to zero image point

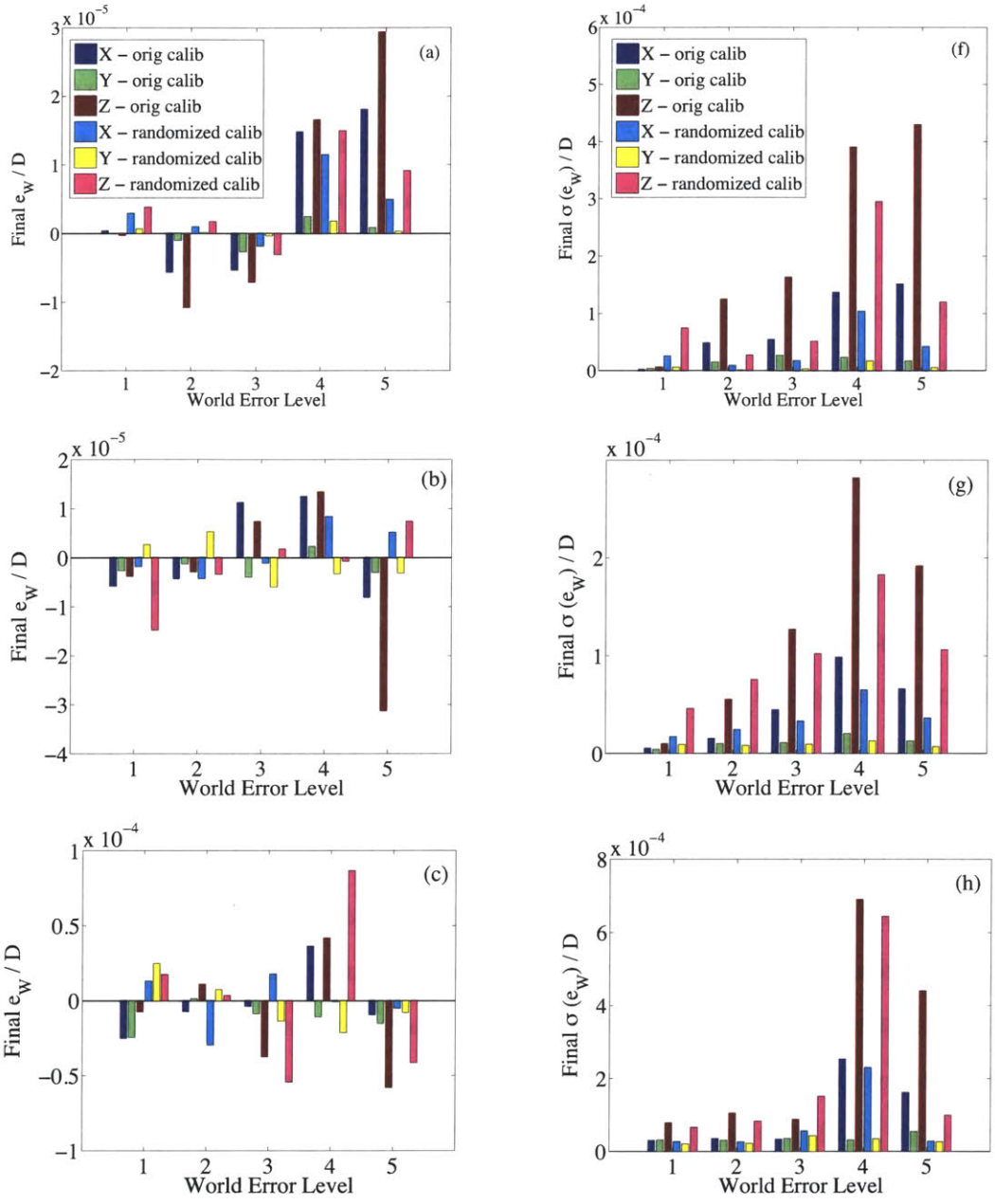


Figure 4-16: Mean (a)-(c) and standard deviation (f)-(h) of the error in world point locations after the final iteration of calibration procedure applied to 8 simulated cameras with and without randomization of cameras. Rows 1-3 correspond to imposed image points error levels of $\sigma(e_u) = \sigma(e_v) = 0$, 0.1 and 0.5 pixels, respectively.

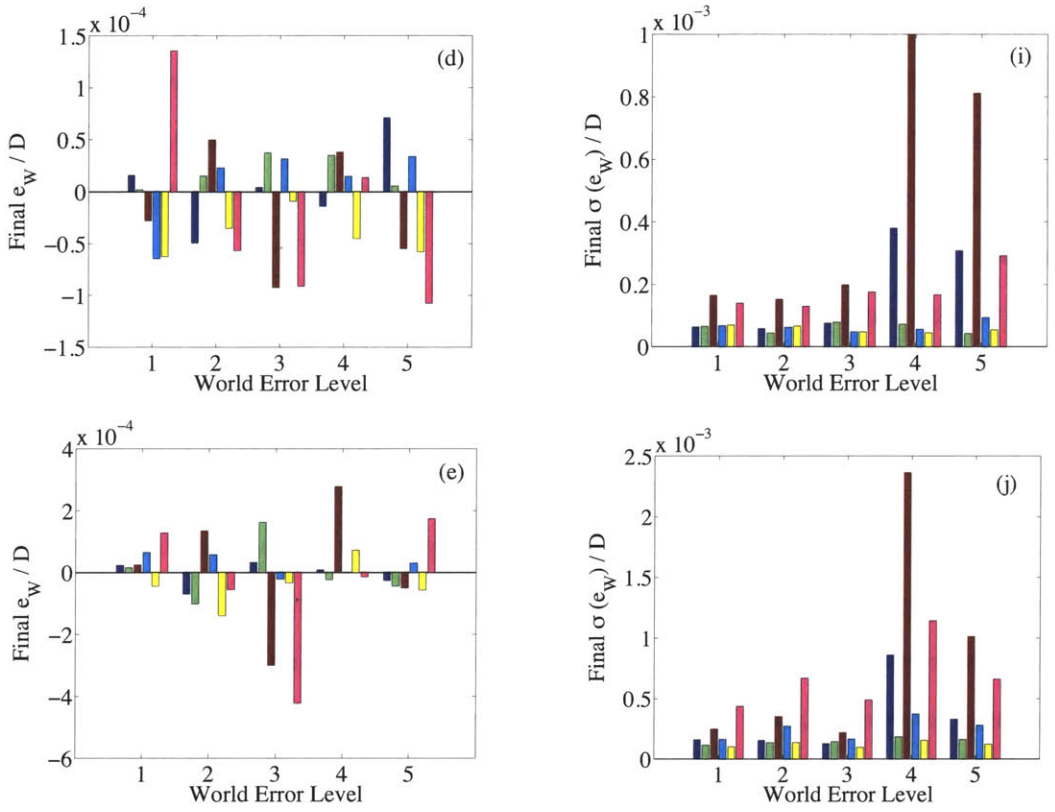
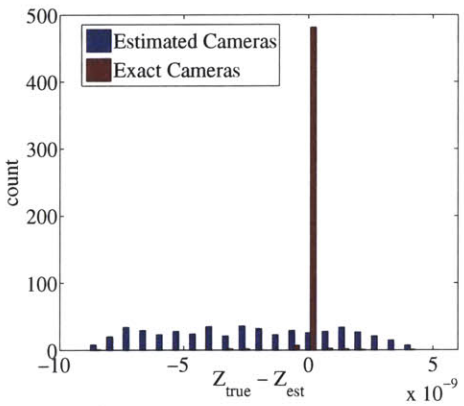


Figure 4-17: Mean (d)-(e) and standard deviation (i)-(j) of the error in world point locations after the final iteration of calibration procedure applied to 8 simulated cameras with and without randomization of cameras. Rows 1-2 correspond to imposed image points error levels of $\sigma(e_u) = \sigma(e_v) = 1$ and 2 pixels, respectively.

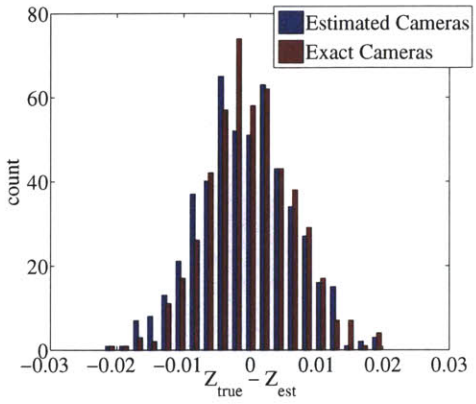
error), each displays the same general distribution with standard deviation increasing with increasing imposed image point error. Furthermore, the similarity between histograms coming from the exact and estimated cameras attests to the accuracy of the auto-calibration procedure. Figure 4-19 shows the distributions of error in the Z coordinates for the case of image point error equal to $\sigma(e_u) = \sigma(e_v) = 0.5$ pixels and all five initial world error levels. Interestingly, the distributions are approximately the same for each case. Therefore, the accuracy of the calibration is a function of the accuracy in image point measurements, but not of the error in the initial guesses of world point locations, as we would hope.

As a final measure of accuracy in world point estimation, the same world grid points are projected into the same simulated cameras used above, except there is no index of refraction change between the points and the cameras. This corresponds to projection of points by pinhole cameras. The linear auto-calibration procedure described in [9] is then applied using the image point correspondences across the cameras. The resulting world points and cameras from the linear auto-calibration procedure require alignment with the true world reference frame. The similarity transform was calculated between the reference grid (at $Z = 0$) as well as four off-plane control points and the corresponding world points estimated by the calibration algorithm. Alignment using simply the reference grid still resulted in large error in the world points, hence the use of additional control points.

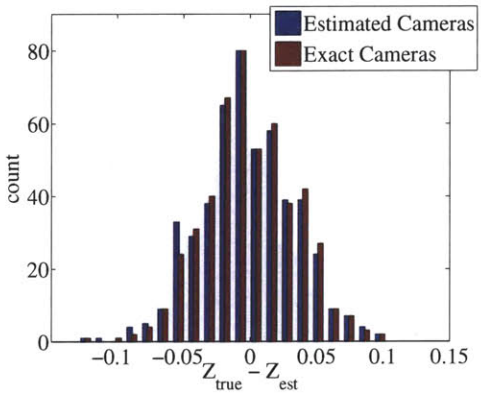
Figure 4-20 shows the distributions of world point Z-coordinate error resulting from the linear auto-calibration applied to the same points and cameras as used in the refractive simulations that generated Figure 4-18 (but now without refraction). The five plots correspond to increasing image point error levels, where the same imposed image point error levels used in the refractive auto-calibration simulations are used here. The shape of the distributions resembles those shown in Figure 4-



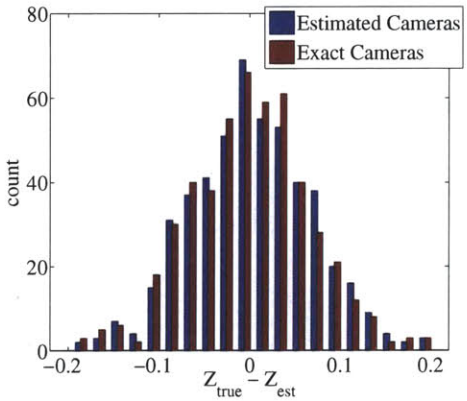
(a) $\sigma(e_u) = \sigma(e_v) = 0$



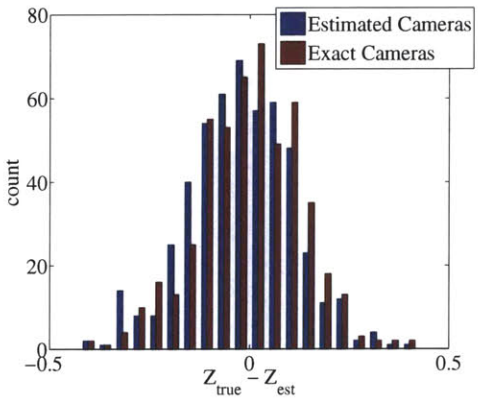
(b) $\sigma(e_u) = \sigma(e_v) = 0.1$



(c) $\sigma(e_u) = \sigma(e_v) = 0.5$

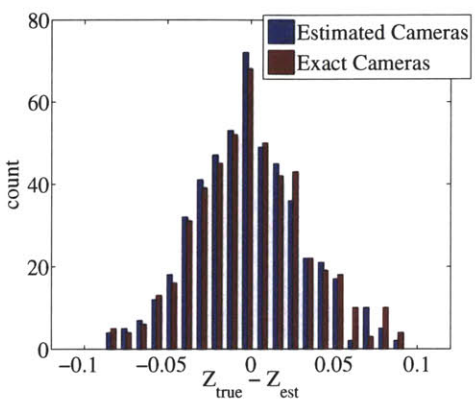


(d) $\sigma(e_u) = \sigma(e_v) = 1$

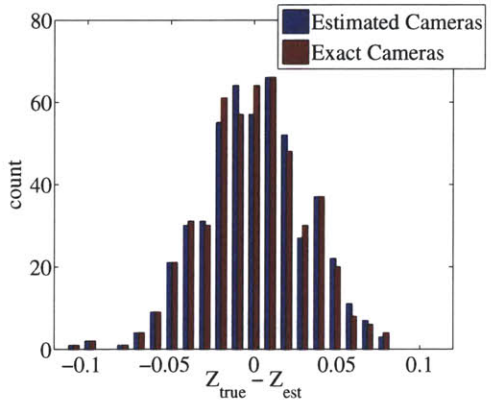


(e) $\sigma(e_u) = \sigma(e_v) = 2$

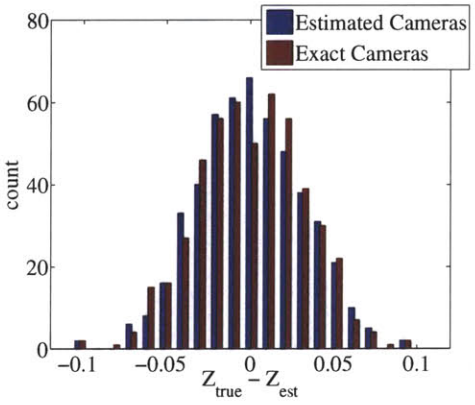
Figure 4-18: Distributions of world point errors for initial world error level of $\sigma(e_W)/D = 0.2$. Each plot (a)-(e) corresponds to increasing image point error level.).



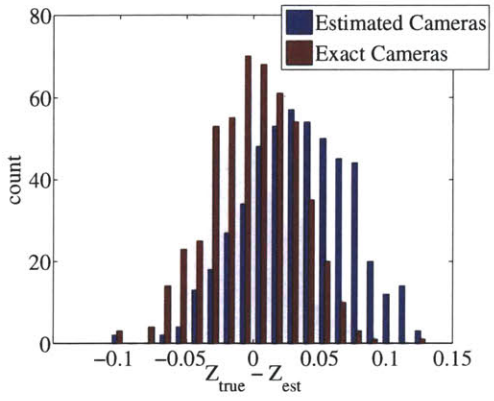
(a) $\sigma(e_W)/D = 0.001$



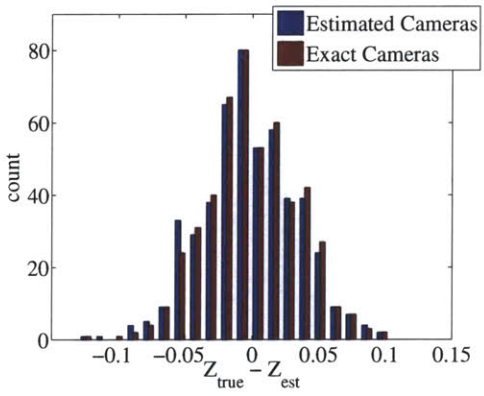
(b) $\sigma(e_W)/D = 0.005$



(c) $\sigma(e_W)/D = 0.01$



(d) $\sigma(e_W)/D = 0.1$



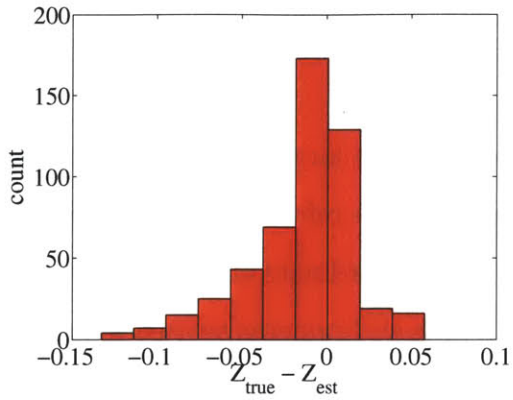
(e) $\sigma(e_W)/D = 0.2$

Figure 4-19: Distributions of world points errors for image point error equal to $\sigma(e_u) = \sigma(e_v) = 0.5$ pixels and the five initial world error levels.

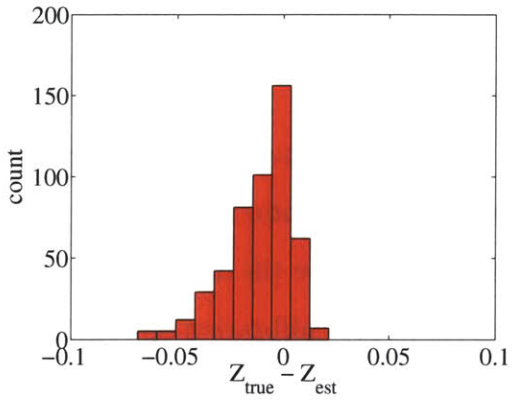
18, with the magnitude of the errors being at least as large, if not larger for the linear auto-calibration procedure as compared with the refractive auto-calibration. Figure 4-21 shows the distributions of world point Z-coordinate errors resulting from the linear auto-calibration applied to the same points and cameras as used to generate Figure 4-19. Because this calibration procedure does not require initial world point estimates, each plot in Figure 4-21 simply corresponds to a different set of cameras. The magnitude of the errors are, in general, larger than the error resulting from the refractive auto-calibration procedure, even though the standard deviation of imposed image point error is the same. Overall, this indicates that the refractive auto-calibration algorithm presented in this chapter is as or more accurate than the linear auto-calibration presented in [9]. Furthermore, more reference geometry is required for accurate alignment of the linear auto-calibration results than for the refractive results.

4.4.3 Application of Refractive Auto-Calibration to Real Data

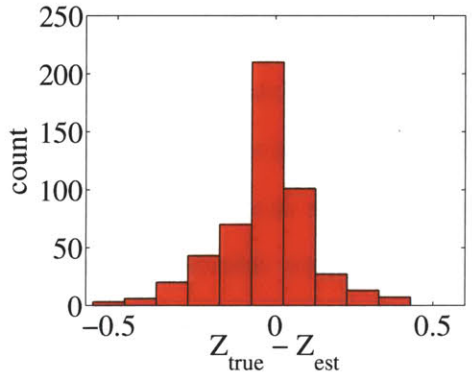
The refractive auto-calibration procedure is now applied to the calibration images from the 3D PIV vortex ring experiment presented in Chapter 2. In that experiment a precision calibration grid traversed through 11 planes in the volume in 4 mm increments generated the calibration points. With the grid located at the middle calibration plane (reference plane, see Figure 4-22), the distance between the front of the plane and inside of the front tank wall was measured at each edge of the grid to attempt to align the grid plane parallel to the front tank wall. From there, the precision traverse moved the grid 20 mm towards the front tank wall, then 40 mm toward the back tank wall, with 4 mm increments between planes. The accuracy of the traverse is expected to be be on the order of ± 0.01 mm [20]. The auto-calibration



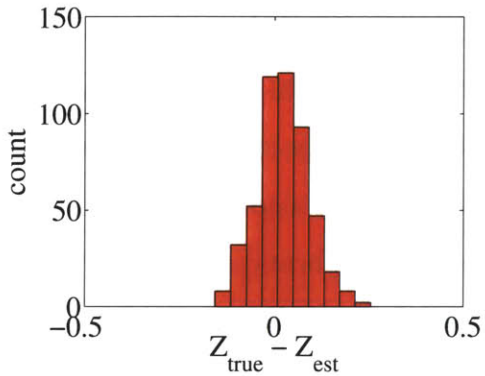
(a) $\sigma(e_u) = \sigma(e_v) = 0$



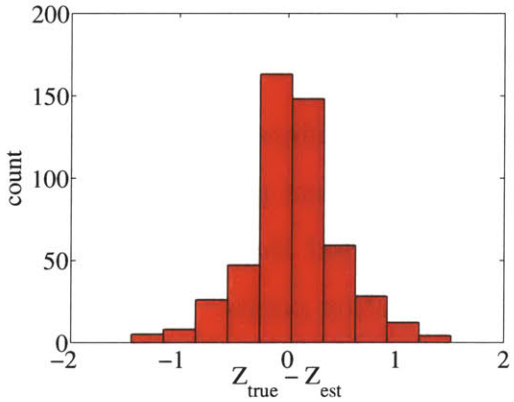
(b) $\sigma(e_u) = \sigma(e_v) = 0.1$



(c) $\sigma(e_u) = \sigma(e_v) = 0.5$



(d) $\sigma(e_u) = \sigma(e_v) = 1$



(e) $\sigma(e_u) = \sigma(e_v) = 2$

Figure 4-20: Distributions of world points errors resulting from the linear auto-calibration applied to the same points and cameras as used to generate Figure 4-18. Each plot corresponds to one of the five image point error levels.

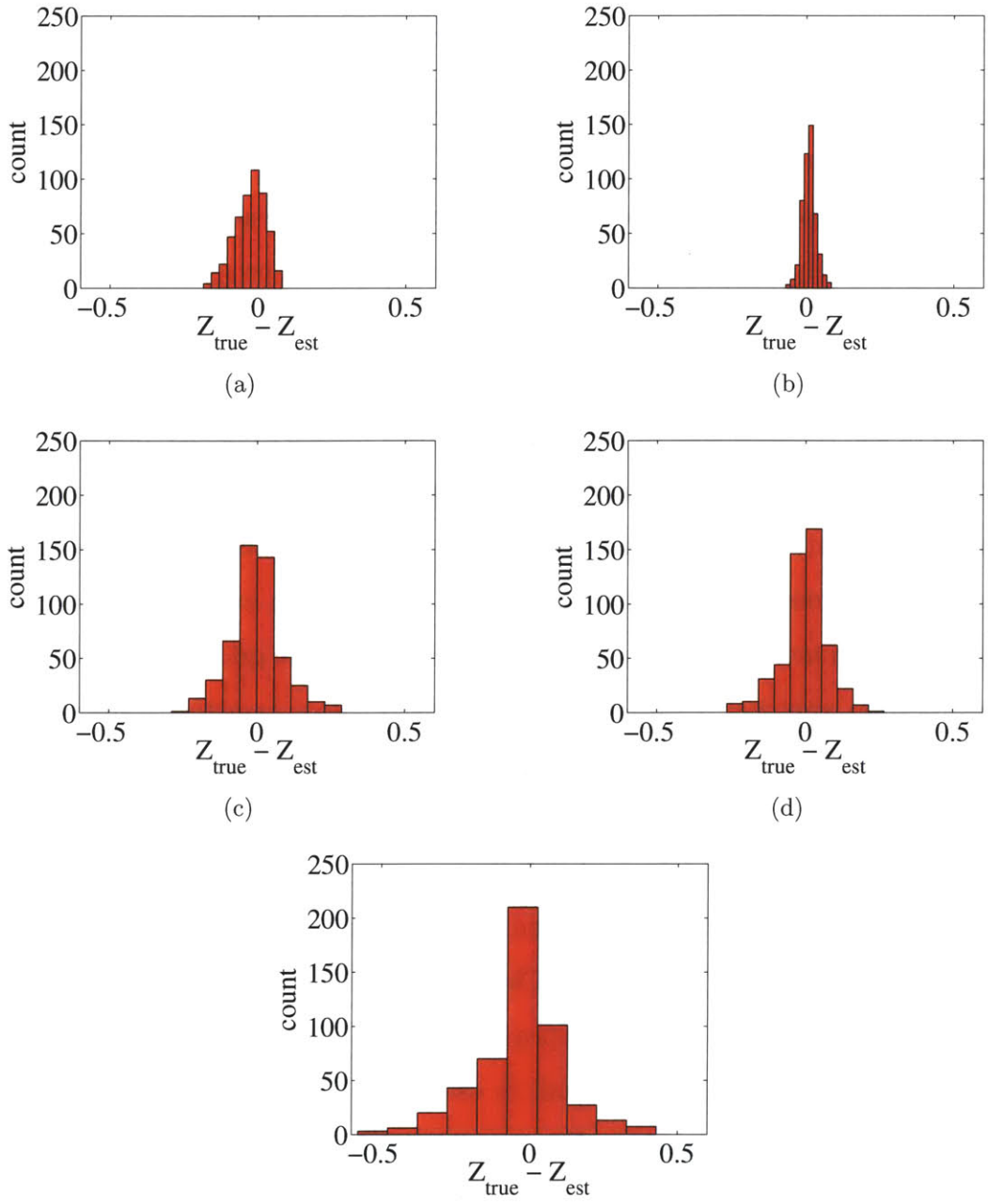


Figure 4-21: Distributions of world points errors resulting from the linear auto-calibration applied to the same points and cameras as used to generate Figure 4-19. Each plot corresponds to a different set of cameras.

procedure described in this chapter aims to eliminate such tedious alignment, which may itself be prone to errors.

To test the auto-calibration procedure with real data, the accuracy of the initial estimates of the plane parameters given to the algorithm is progressively decreased over three tests by varying the initial estimates of the Z location of each plane. All planes are assumed to be parallel to the front tank wall. For the first test, the initial estimates of the Z coordinate of the plane locations ranges from -20 mm to 20 mm in 4 mm increments; these correspond to the assumed true locations based on the precision alignment. For the other two tests, the initial estimates of the Z coordinate of the plane locations ranges from -10 mm to 10 mm in 2 mm increments and -5 mm to 5 mm in 1 mm increments, respectively. For each test, the grid-based randomized auto-calibration approach with four camera subsets yields camera parameters and world point estimates. A single grid point from the central plane location is taken as the reference point for all tests, similar to the simulations. Point correspondences are found using a circle detector with subpixel accuracy [21].

Figure 4-23 shows the mean and standard deviation of the reprojection error for each camera and each calibration test. Clearly, the initial estimates of the plane locations has little impact on the final reprojection error. Furthermore, the magnitude of reprojection errors is relatively small in each camera, indicating good accuracy in the image point measurements. Figure 4-24 shows two calibration images with measured image points and reprojected world points for each test plotted on the image. Figure 4-24(a) shows an image from camera 1 with the grid at the location farthest from the front wall and Figure 4-24(b) shows an image from camera 3 with the grid middle (reference) location. For each image, zoomed views of a grid point are shown for clarity.

The accuracy of estimated world points is harder to characterize than the accu-

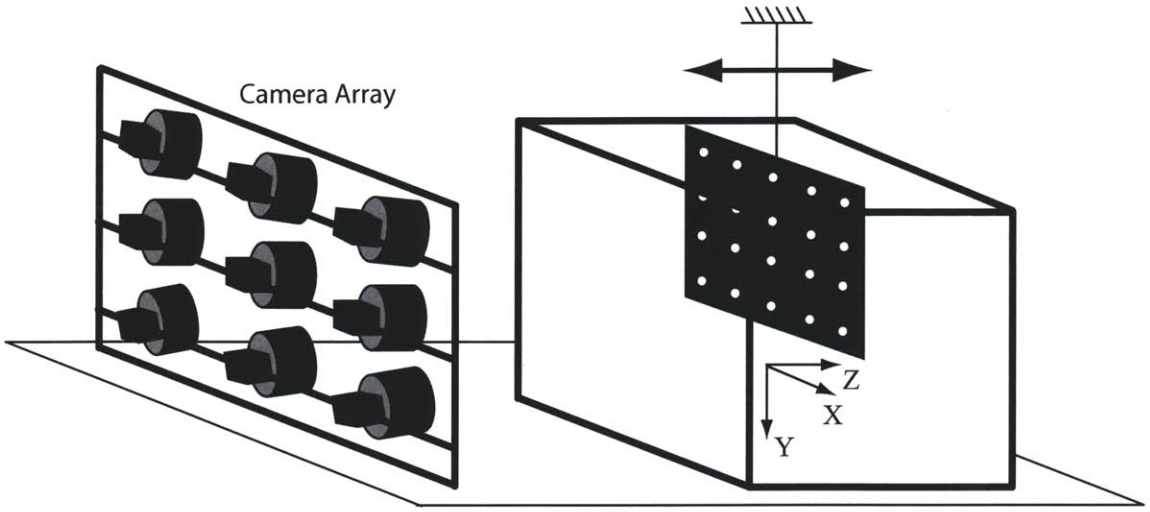
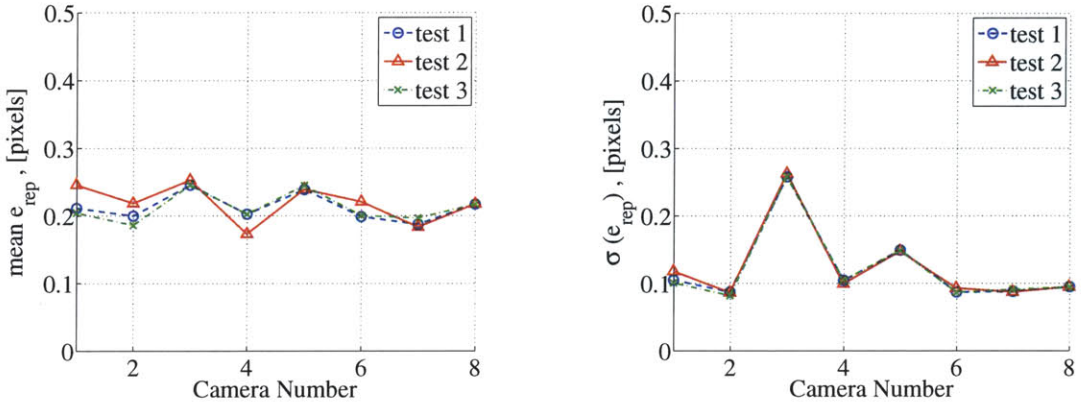
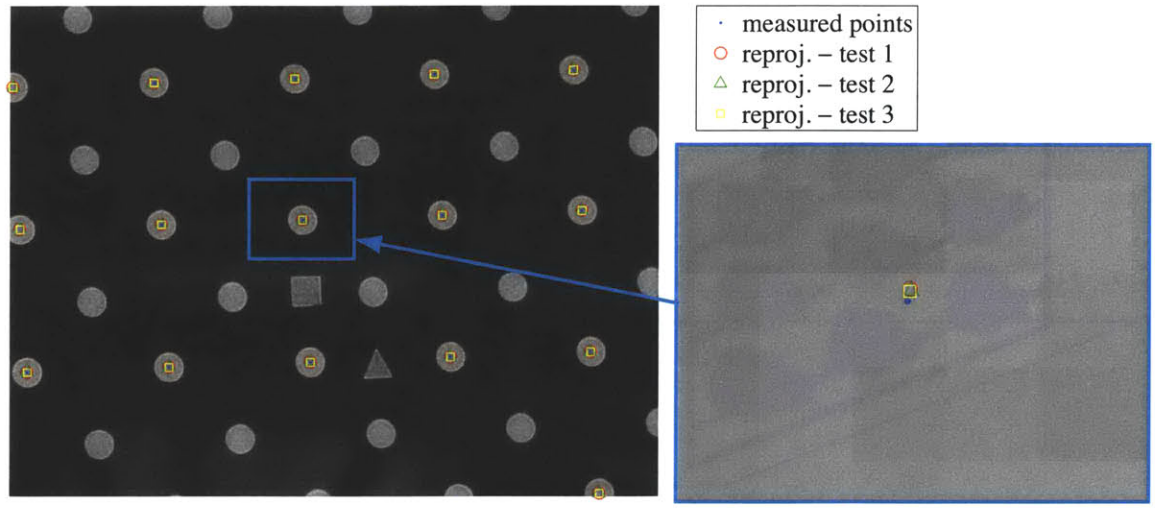


Figure 4-22: Schematic of the calibration setup for the 3D SAPIV vortex ring experiment.

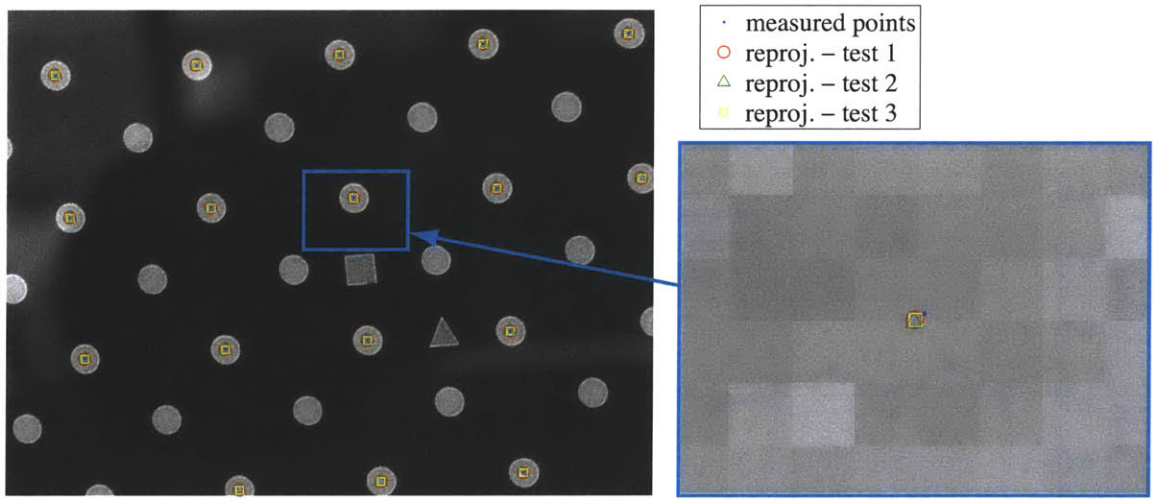


(a) (b)

Figure 4-23: Mean (a) and standard deviation (b) of the reprojection error for each camera and each calibration test for the 3D PIV vortex ring experiment.



(a)



(b)

Figure 4-24: Measured and reprojected points for all calibration tests for the 3D PIV vortex ring experiment: (a) shows an image from camera 1 with the grid at the location farthest from the front wall and (b) shows an image from camera 3 with the grid middle (reference) location.

racy in the image point measurements, given that these points are unknown. The most accurate and precise portion of the physical calibration procedure is the translation of the grid along the Z axis by the precision traverse. Therefore, as a measure of accuracy in the world points, the difference between Z-coordinates of corresponding grid points from adjacent plane locations is calculated. The nominal spacing between planes (according to the translation requested of the traverse) is 4 mm. Figure 4-25 shows the mean value of Z distance between estimated world point locations on adjacent planes; the value k refers to the distance between planes p_k and p_{k+1}. The error bars represent one standard deviation from the mean. For all tests, the estimated mean plane spacing is larger than 4 mm, which may be due to a bias in the traverse itself. However, the accuracy of the calibration method is evident from the data. The plane spacing fluctuates by less than ± 0.02 mm for all tests, which is again likely due to the accuracy of the traverse itself. This is supported by the fact that each calibration test predicts the fluctuations similarly, and therefore the fluctuations are not due to the calibration algorithm. Therefore, by this metric, the calibration accuracy approaches the accuracy of the traverse in locating world points, even when the initial estimates of plane locations are poor.

Attention is now turned to a possible bias error between the different calibration tests. Figure 4-26(a) shows the estimated world point locations resulting from the calibration algorithm for the three test cases. The two tests with poorer initial conditions appear to have a bias error whereby the entire set of points is rotated (and possibly translated) from the test 1 data set even though all estimted points are aligned with the same reference point. Keep in mind that the test 1 data set is not the ground truth (which is unknown), but is used here as the reference data for discussing the bias error.

Figure 4-23 shows that the reprojection error for each camera in each test is

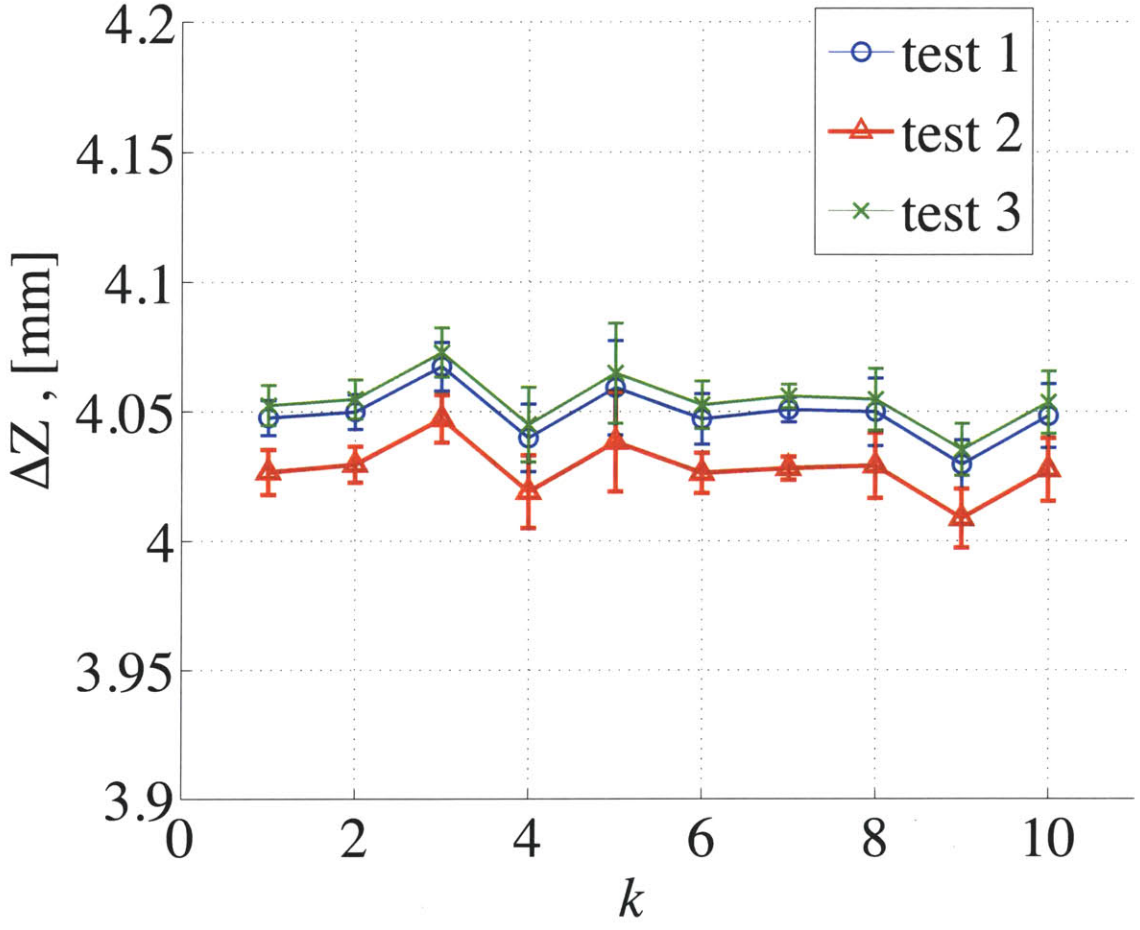


Figure 4-25: Mean value of Z distance between estimated world point locations on adjacent planes. Error bars represent one standard deviation from the mean.

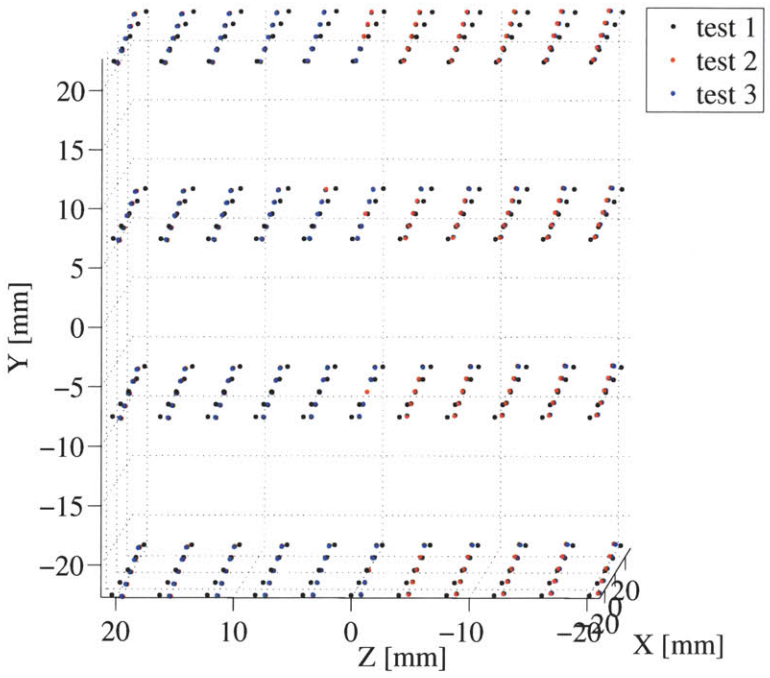
reduced to roughly the same level, which begs the question: are there multiple combinations of world points and camera parameters that yield a solution to the problem (i.e., that minimize the sum of squared reprojection errors)? Because the wall is assumed to be of infinite extent, translations of the world points and camera centers in X and Y will or course give the same result, but this is mitigated by alignment with the reference point. Also, the entire set of world points and camera centers could be rotated around the Z-axis and still yield the same result, but this does not explain the apparent rotation of world points about the X and Y axes in Figure 4-26(a).

Therefore, to investigate this error further, the 9 innermost cameras (3×3 arrangement) from the simulated camera array described in Chapter 2 are used. First, the same grid points as shown in Figure 4-10 are projected into each camera using the refractive model to generate true image points. Then, the calibration for each camera is found using these true image points and a series of perturbed world points. The calibration is calculated using the Levenberg-Marquardt method to adjust the camera parameters with the sum of squared errors between reprojected and true image points serving as the objective function. The idea being that, if the calibration found using the perturbed world points and true image points contains no error (i.e., projects the perturbed world points to the original true image points), then there are multiple combinations of world points and camera parameters that yield a solution. Note that *this statement is true for linear auto-calibration methods* (e.g. [9]), but is not expected to be true for the refractive auto-calibration method given the imposition of more physical constraints. The perturbed world point locations are generated by imposing a rigid rotation ranging from -4° to 4° in 0.5° increments about the vertical axis passing through the centroid of the world points and separately by shifting the world points along the Z-axis by an amount ranging from 0 mm to 50 mm in 5 mm increments. The resulting mean error between the reprojected and true image

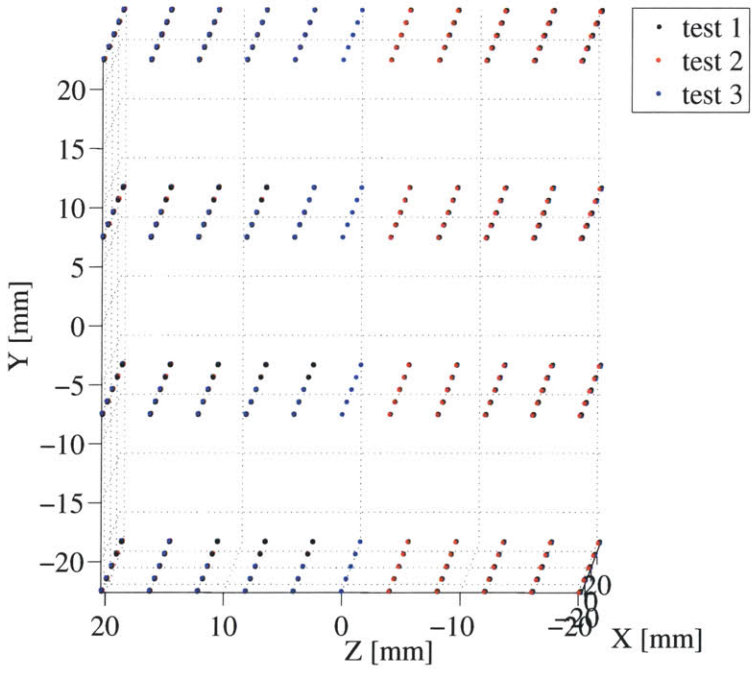
points for each case is shown in Figure 4-27. Note that these plots do not represent the sensitivity of the cameras to changes in point locations, but rather are indicative of the sensitivity of the auto-calibration method to changes in points locations because the algorithm finds the best fit camera parameters based on the world point locations and true image points.

The mean error never reaches zero except for when the world points reside at their true location. For the central camera (camera 5) the error remains small independent of translations of points along the Z-axis; however, the auto-calibration algorithm attempts to minimize the summed reprojection error across all cameras when estimating world point coordinates, which would be non-zero. The results of this simulation indicate that, under perturbations of the world points via a rigid rotation about a vertical axis or translation along the Z-axis, there are no camera parameters that yield a minimum solution. In other words, a unique solution exists for the world point locations and camera parameters with respect to these motions.

Still the question remains of why an apparent rotation and translation of world points for the real data was observed. Some insight can be shed by noting the caveat that, for the simulation test above, the image points are assumed to be measured with perfect accuracy; in practice, this is never true. A similar analysis is now applied to the real data. Assume, for now, that the camera parameters and world points estimated from test 1 represent the true solution. Projecting the test 1 estimated world points into each camera then gives "true" image points. A similar analysis as applied to the simulated data is used here. For tests 2 and 3, the best fit camera parameters are found using the estimated world points from each respective test - considered to be the "perturbed" world points - along with the "true" image points resulting from test 1. The mean error between the "true" image points and the projections of the perturbed points using the new camera parameters is shown

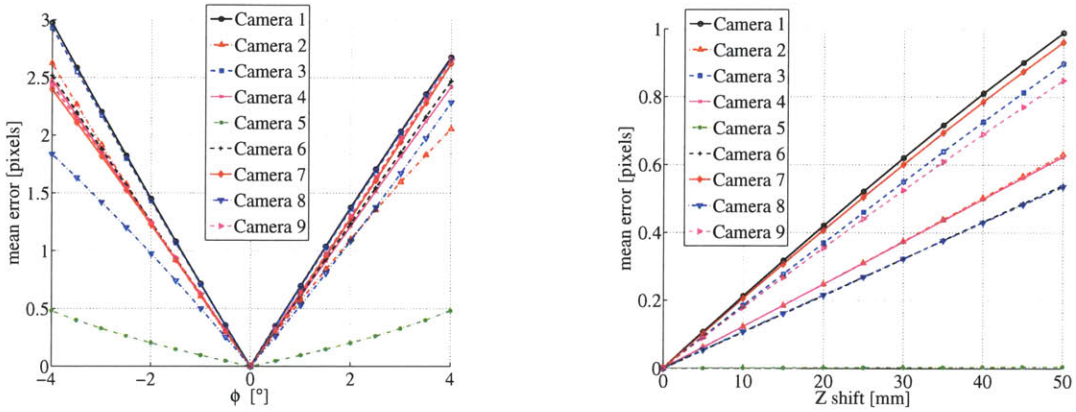


(a)



(b)

Figure 4-26: Estimated world point locations resulting the calibration algorithm for the three tests before (a) and after (b) reference plane alignment.

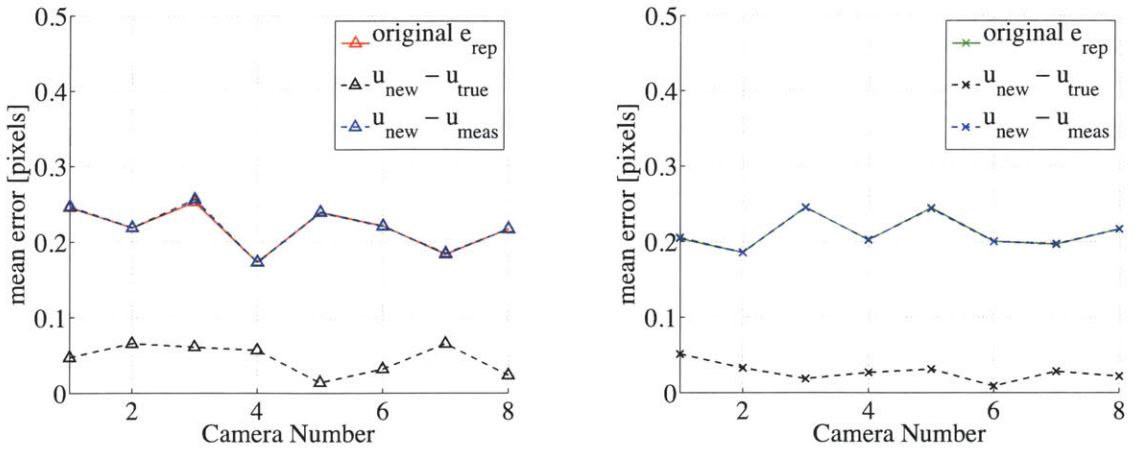


(a) (b)

Figure 4-27: Mean error between the reprojected and true image points for rigid rotation ranging from -4° to 4° in 0.5° increments about the vertical axis passing through the centroid of the world points (a) and shifting the world points along the Z-axis by an amount ranging from 0 mm to 50 mm in 5 mm increments (b).

by the black lines in Figures 4-28(a) & 4-28(b). The magnitude of these errors is relatively small compared to the original mean reprojection error calculated for each test (also shown in Figures 4-28(a) & 4-28(b)). Furthermore, the error between the new reprojected points and the original measured image points equals the original reprojection error with very small discrepancy. These results show that, in practice, multiple combinations of world points and camera parameters that yield a minimal solution exist. With perfectly measured image points, we would not expect this to be the case. However, because there is error in the image point measurements, multiple combinations of world points and camera parameters that minimize the reprojection error can be found even if the reprojected points differ from the true image points.

To ensure that the error indeed corresponds to a rigid rotation and translation of the world points and not something more problematic like skewing, the similarity transformation is calculated between the estimated points on the reference plane



(a) test 2

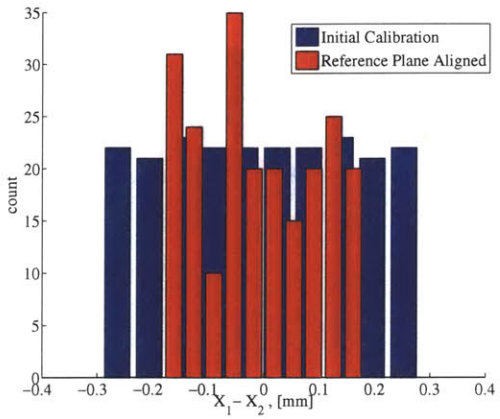
(b) test 3

Figure 4-28: Mean error between the "true" image points and the projections of the perturbed points, original mean reprojection error and error between the new reprojected points and the original measured image points for test 2 (a) and 3 (b) with test 1 assumed to be the "true" result.

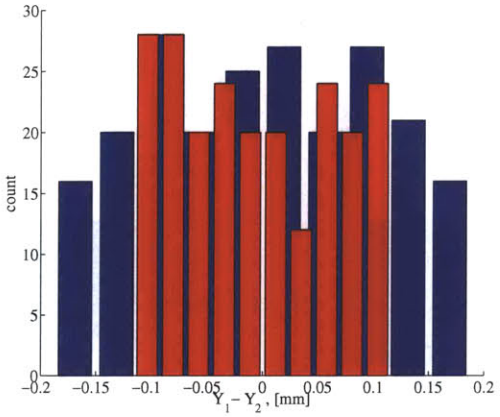
(middle plane) and the assumed location of points on the reference plane for each test case. The transformation is calculated using the method described in [22]. The hypothesis is that if most of the error between data sets is reduced simply through application of a similarity transformation, then the error is effectively due to the estimated points being placed in a coordinate system which is rotated and translated from the true coordinate system. Figures 4-29, 4-30 & 4-31 show the histograms of the point-by-point differences in each coordinate between estimated points for all test cases; the blue bins and red bins correspond to differences calculated before and after application of the similarity transformations, respectively. It should be noted that scale was also left as a parameter in the similarity transformation, and was calculated to be one for each test, which is consistent with the notion that scale is fixed by the grid. In Figures 4-29 & 4-30, which show histograms of the error

between test 1 and test 2 points and test 1 and test 3 points, respectively, the error is significantly reduced in all coordinates, particularly in Z, after alignment with the reference plane. The error reduction for the difference between test 2 and test 3 points - shown in Figure 4-31 - is less dramatic since these points were estimated near each other in the first place. Therefore, the conclusion is drawn that the world points estimated by the calibration procedure are located in a coordinate system that is within a rotation and translation of the true coordinate system due to the auto-calibration procedure settling into a local minimum.

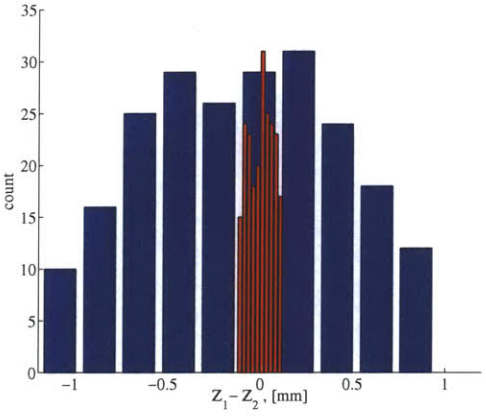
To attempt to remove this bias error, it is proposed that some accurately known points be imaged during the calibration, and a similarity transformation calculated as discussed here. As described in [22], at least three points are required in theory, but more points provide better noise immunity. Also, the point correspondences can be coplanar, but not collinear. Therefore, a calibration grid placed at one well-controlled location could provide the reference points, or the reference geometry could even be provided by an object in the volume of interest (e.g. shaft, foil, etc). Linear auto-calibration methods like that of Svoboda [9] all *require* reference geometry to align the resulting world points with a reference coordinate system because the output of these techniques are within a similarity transformation of a world frame (including scale). However, what should be mentioned with respect to the refractive auto-calibration discussed in this chapter is that the relative location of world points resulting from the calibration procedure is very accurate as evidenced by the calculated plane spacing in Z for all tests. The points are simply estimated in a world frame that is rotated and translated from the true frame.



(a)

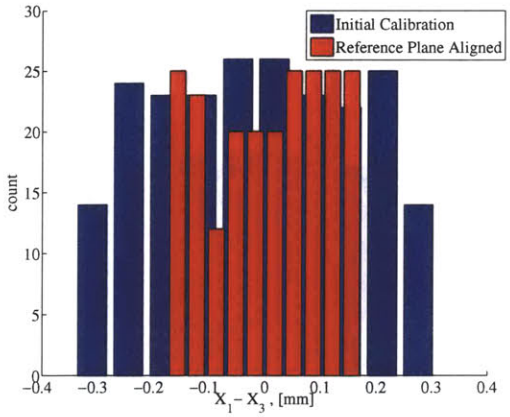


(b)

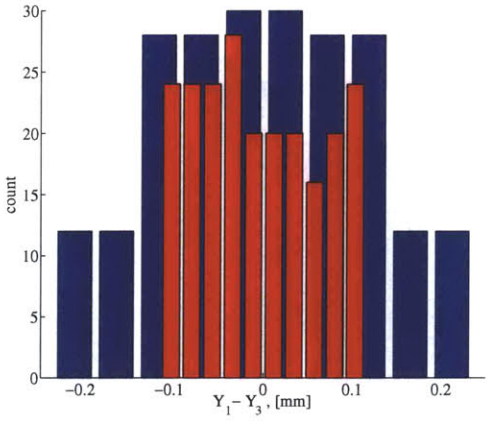


(c)

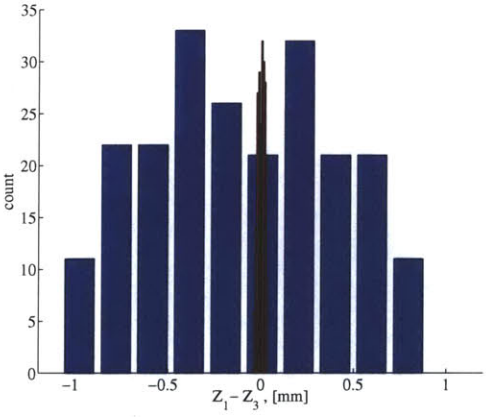
Figure 4-29: Histograms of the point coordinate differences between test 1 and test 2 before (blue bins) and after (red bins) reference plane alignment.



(a)

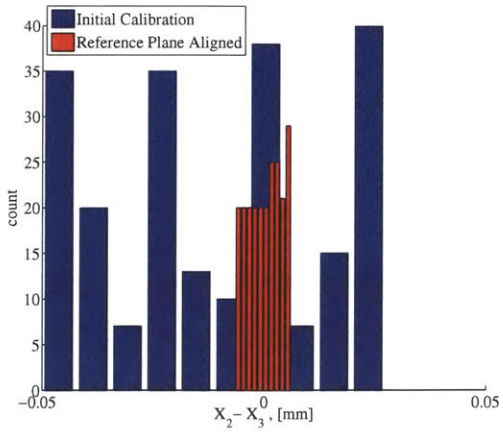


(b)

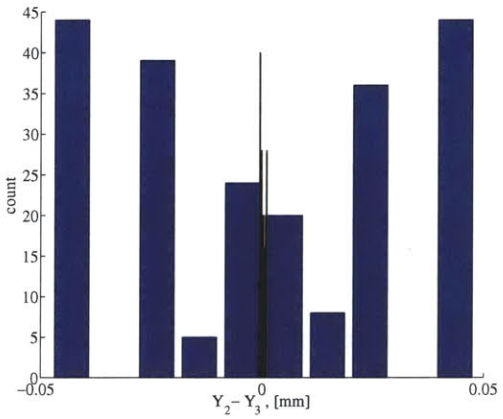


(c)

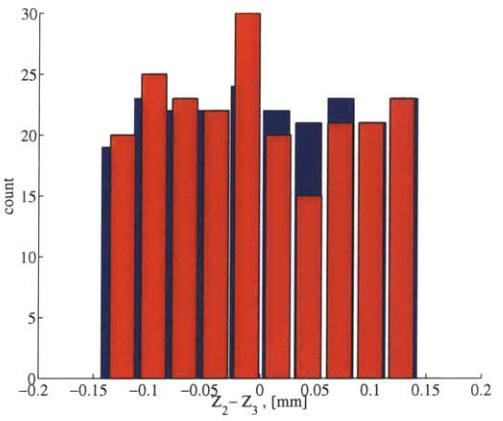
Figure 4-30: Histograms of the point coordinate differences between test 1 and test 3 before (blue bins) and after (red bins) reference plane alignment.



(a)



(b)



(c)

Figure 4-31: Histograms of the point coordinate differences between test 2 and test 3 before (blue bins) and after (red bins) reference plane alignment.

4.5 Conclusions

A novel method for performing auto-calibration of a multi-camera setup viewing through an air-glass-water transition was presented. The method contributes to multi-camera fluids experiments, where it is desirable to avoid tedious alignment of calibration grids in multiple locations. Traditional pinhole camera model based auto-calibration algorithms do not apply when an air-glass-water transition exists between the cameras and the scene, and therefore the refraction was accounted for in the present model resulting in improved accuracy. Cameras can be in general location and orientation with respect to the wall. The accuracy of the method was demonstrated by applying the technique to real calibration data where explicit knowledge of grid locations was removed. World points were shown to be reconstructed in a frame that was within a rotation and translation of the true reference frame; alignment with the true reference frame can be achieved through knowledge of at least three well-controlled reference points.

Bibliography

[1] F. Pereira, M. Gharib, D. Dabiri, and D. Modarress. Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. Application to bubbly flows. *Experiments in Fluids*, 29(7):S078–S084, 2000.

[2] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.

[3] O. Faugeras, Q. Luong, and S. Maybank. Camera self-calibration: Theory and experiments. In G. Sandini, editor, *Computer Vision, ECCV 92*, volume 588 of *Lecture Notes in Computer Science*, pages 321–334. Springer Berlin / Heidelberg, 1992.

[4] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[5] Guanghui Wang, Q. M. Jonathan Wu, and Wei Zhang. Kruppa equation based camera calibration from homography induced by remote plane. *Pattern Recognition Letters*, 29(16):2137–2144, 12 2008/12/1/.

[6] Manmohan Chandraker, Sameer Agarwal, David Kriegman, and Serge Belongie. Globally optimal algorithms for stratified autocalibration. *International Journal of Computer Vision*, 90:236–254, 2010. 10.1007/s11263-009-0305-2.

[7] Marc Pollefeys, Reinhard Koch, and Luc Van Gool. Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *International Journal of Computer Vision*, 32:7–25, 1999. 10.1023/A:1008109111715.

[8] Antonio Valdés, José Ronda, and Guillermo Gallego. The absolute line quadric and camera autocalibration. *International Journal of Computer Vision*, 66:283–303, 2006.

[9] Tomáš Svoboda, Daniel Martinec, and Tomáš Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4):407–422, August 2005.

[10] Tali Treibitz, Yoav Y. Schechner, and Hanumant Singh. Flat refractive geometry. In *In Proc. IEEE CVPR*, 2008.

[11] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334, 11 2000/11/01.

[12] Peter Sturm. Critical motion sequences for the self-calibration of cameras and stereo systems with variable focal length. *Image and Vision Computing*, 20(5-6):415–426, 4 2002/4/15/.

[13] Tomáš Svoboda, Daniel Martinec, Tomáš Pajdla, Jean-Yves Bouguet, Tomas Werner, and Ondrej Chum. Multi-camera self-calibration. http://cmp.felk.cvut.cz/ svoboda/SelfCal/.

[14] B. Wieneke. Volume self-calibration for 3d particle image velocimetry. *Experiments in Fluids*, 45(4):549–556, 2008-10-01.

[15] J. Heikkila. Geometric camera calibration using circular control points. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1066–1077, Oct 2000.

[16] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *Robotics and Automation, IEEE Journal of*, 3(4):323–344, August 1987.

[17] Steven C. Chapra and Raymond P. Canale. *Numerical Methods for Engineers.* McGraw-Hill, New York, NY, 5th edition, 2006.

[18] C. Harris and M. Stephens. A combined corner and edge detector. In *4th Alvey Vision Conference*, pages 147–151, 1988.

[19] Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, June 1981.

[20] I-Chien Lee and Ping-Hei Chen. An inclined jet through a forward expanded hole ejected into mainstream over a concave surface. *Journal of Fluid Science and Technology*, 2(2):311–321, 2007.

[21] Samuel Raben. Sub-pixel circle center detector. Matlab Code, 2010.

[22] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 9(5):698–700, 1987.

Chapter 5

Application to Multi-Phase Circular Plunging Jet

5.1 Introduction

Bubbly flows are ubiquitous in a variety of engineering problems ranging from air-sea interaction to ship surface signature to propellor cavitation to industrial processes. However, measurement of these flows presents several challenges. Often, interesting flows are optically dense due to large void fractions, which limits applicability of several techniques. Full temporal and three-dimensional spatial resolution of the bubble fields is a desirable outcome of measurement, but further limits the class of available instruments. Conductivity probes offer a proven method for measuring void fraction and bubble count rate, but only do so locally and are invasive [1]. Acoustic measurements using hydrophones can provide overall void fraction and bubble count, as well as bubble size spectra [1], but cannot provide detailed information about the location or track of bubbles within a flow. Other authors used planar shadow imaging

to resolve bubble size and location in two dimensions [2]. To obtain information about the location and track of bubbles in three-dimensions as well as size, three-dimensional imaging techniques are often the tool of choice. Pereira et al. [3] applied defocusing digital particle image velocimetry to measure bubble fields, although the bubble sizes were relatively small. Tian et al. [4] used digital holography to measure bubble size and location with a very efficient algorithm allowing for near real-time output. However, the digital holographic approach is typically limited to lower void fractions or flows that are not too optically dense.

In this chapter, a technique based on synthetic aperture (SA) imaging that enables instantaneous resolution of three-dimensional bubble fields is discussed. The fields can be quite dense, since SA imaging allows for reconstruction despite partial occlusions.

To generate a flow field with variable bubble density and size, we employed the classic experiment of a circular plunging jet impinging on a free surface. A circular plunging jet impinging on a free-surface tends to entrain bubbles that are carried downward within a conical volume under the influence of the jet momentum before ultimately rising to the surface outside of the conical volume under the influence of buoyancy [5, 6], as depicted in Figure 5-1 for an experimental case studied herein. A large body of literature exists on the multi-phase circular plunging jet problem. Extensive reviews on the subject are offered by Bin [5] and Chanson [6]. By far the most prominent experimental methods used to investigate the subject have been single point measurements (e.g., conductivity probes) and two-dimensional imaging; therefore data sets are inherently limited to two-dimensional, at most. Several studies described the different regimes of entrainment, which are dependent of the jet being laminar or turbulent [7, 8, 9, 10, 11, 12]. When the jet is laminar, the free-surface undergoes a large, cylindrical like deformation creating an air sheath and does not

entrain air until a critical Weber number where the jet radius and velocity exceeds a critical value [7, 8, 9, 10, 11, 12]. The entrainment occurs beneath the free-surface where the sheath begins to break up into bubbles. For turbulent jets, wave-like instabilities on the surface of the jet cause cavities to be periodically pinched off at the surface, and the resulting air pockets are entrained and broken up by the high speed jet [7, 8, 1, 12]. As described by Chanson & Manasseh [1], the entrainment caused by turbulent jets can be further broken down into three regimes corresponding to increasing impact speed (for a given jet height). In regime I, small bubbles are individually entrained directly from the free-surface. In regime II, larger air cavities for periodically at random locations around the periphery of the impact point, and air packets are sheared from the cavities and entrained. Finally, in regime III, an air cavity forms around the entire periphery of the impact point and packets are continually broken off and entrained into the flow. Relatively small changes in impact speed and/or jet height can have a large effect on the regime of air entrainment, and thus the amount of air entrained.

Figure 5-2 show instantaneous views from a high speed camera zoomed in on the near surface region for the experiments presented in this chapter. Each case appears to demonstrate the characteristics of regime II entrainment. For jet height $h = 7$ mm (referring to nozzle height above the free-surface), small bubbles are entrained after pinching off from the periodic air cavities. In the case of jet height $h = 21$ mm, the air cavities are larger, and air packets are sheared off and entrained, and subsequently break up into smaller bubbles. Finally, for jet height $h = 43$ mm, the periodic air cavities are larger still and more air packets are sheared off and entrained, increasing the overall volume of air in the flow.

Another set of studies is concerned with what Chanson terms the developing flow region [6], which contains the developing air diffusion layer. Below the initial entrain-

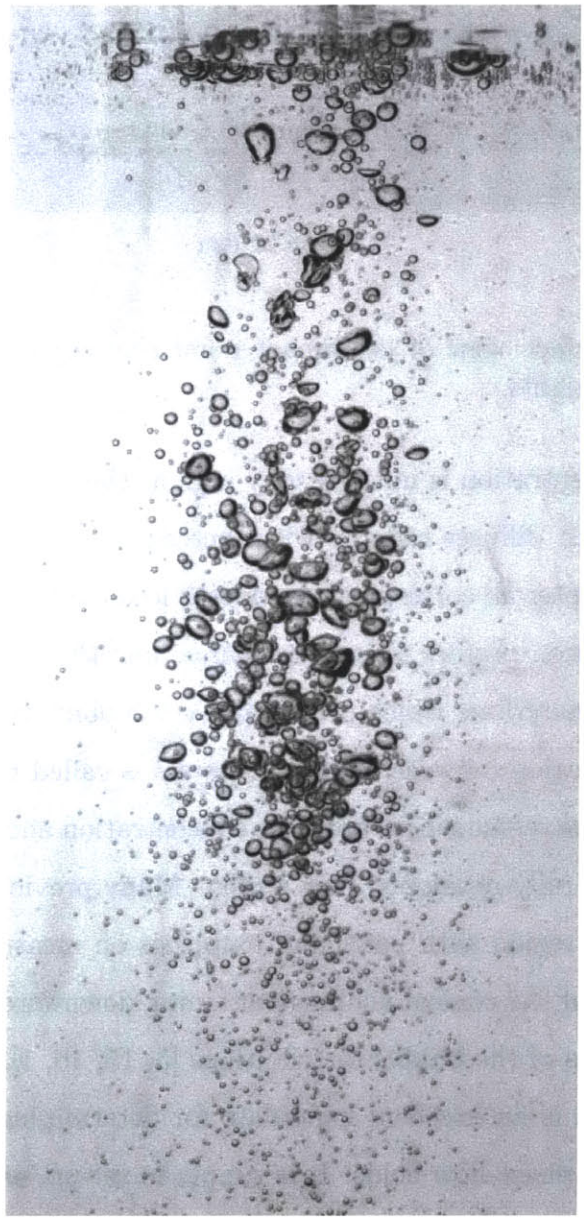
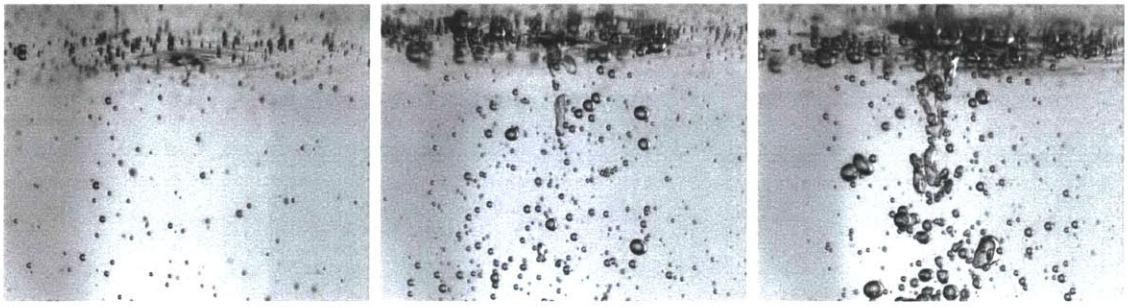


Figure 5-1: Sample instantaneous high-speed image of a bubble-entraining circular plunging jet.



(a) $h = 7$ mm (b) $h = 21$ mm (c) $h = 43$ mm

Figure 5-2: Near-surface view of jet impact point showing the entrainment regime for the various jet heights.

ment point, air concentration is initially very high at the entrainment point and low elsewhere, and the air diffuses radially with increasing depth [1, 13, 14]. The data presented in this chapter all corresponds to depths lower than the developing flow region. The concentration profiles ultimately merge, and the maximum concentration is found on the jet centerline, which occurs above the point where the fluid velocity profiles merge; the region between these two points is called the redistribution flow region [6]. Finally, the region where both the concentration and velocity profiles have merged is called the fully-developed flow region. Many previous studies discuss the fully-developed flow region with particular emphasis on measuring bubble penetration depth, the size of the conical volume containing downward moving bubbles and overall characteristics of the bubble distributions [5, 15, 16, 1].

The phenomenon is an excellent candidate for determining the ability of a system to resolve multi-phase flow fields. It is simple to set-up, easy to scale by simply changing the jet diameter or speed, and a quasi-steady state plume can be formed in just a few seconds. For these reasons we have utilized a turbulent air entrainment setup to perform measurements of a multi-phase flow field using the synthetic

aperture imaging technique. Furthermore, due to the experimental challenges, several aspects of the problem remain unresolved providing an opportunity to shed new insight based on the data yielded from the SA imaging technique.

This chapter first discusses the experimental setup and parameters of the plunging jet. Then, the details of the application of the SA imaging method to this problem are discussed. Next, two algorithms are presented for extracting bubble size and locations from the refocused SA images, including a novel metric for defining bubble features. Finally, the time-averaged bubble distributions and 3D bubble locations resulting from three test cases are presented. The tests are put in context of existing literature on the subject, and results are compared to several scaling laws showing excellent agreement. A quantitative measure for defining the penetration depth and width of the bubble cloud is also presented. Finally, a scaling law for total air concentration as a function depth below the free-surface is proposed, and is believed to be the first presentation of such a result.

5.2 Experimental Methods

To generate a steady-state jet, an outlet tube and nozzle was mounted below a tank with constant head height of water, as shown in Figure 5-3. The tube and nozzle inner diameter could be set at either 6.35, 9.53 or 12.7 mm. All data presented in this chapter comes from the 6.35 mm nozzle setup with the nozzle exit at three different heights above the free-surface. A bilge pump cycled water from the test tank to the head tank at a flow rate equal to the flow rate through the nozzle. As a fail-safe to ensure maintenance of a constant head pressure, an overflow outlet and tank was untilized. The head and overflow tanks, outlet tube and nozzle were all mounted to a vertically arranged precision traverse to allow the nozzle height (and

thus just impact velocity) to be varied without changing the overall pressure drop between the head tank and the nozzle outlet.

Table 5.1 summarizes the experimental parameters for the tests presented in this chapter; h is the jet height and V_i and D_i are the impact velocity and diameter, respectively. Although the nozzle diameter and outlet velocity remain constant, the impact velocity and jet diameter change as a function of jet height due to the conversion of potential to kinetic energy. Assuming a steady state jet and uniform velocity profile, the diameter and velocity at impact are estimated from application of Bernoulli's equation, according to [17]. The dimensionless parameters reported in Table 5.1 are the Reynolds number, given by,

$$Re = \frac{V_i D_i}{\nu} \tag{5.1}$$

where ν is the kinematic viscosity of water, the Froude number, defined as

$$Fr = \frac{{V_i}^2}{g D_i} \tag{5.2}$$

where g is the gravitational acceleration, the Weber number, given by

$$We = \frac{\rho {V_i}^2 D_i}{\sigma} \tag{5.3}$$

where σ is the surface tension between water and air (73 mN/m for clean water) and the Morton number defined as

$$Mo = \frac{g \nu^4 \rho^3}{\sigma^3} \tag{5.4}$$

Nine high speed Photron Fastcam cameras (two model SA1, four model SA3, three model SA5) arranged in a 3×3 array imaged the flow (Figure 5-4). The

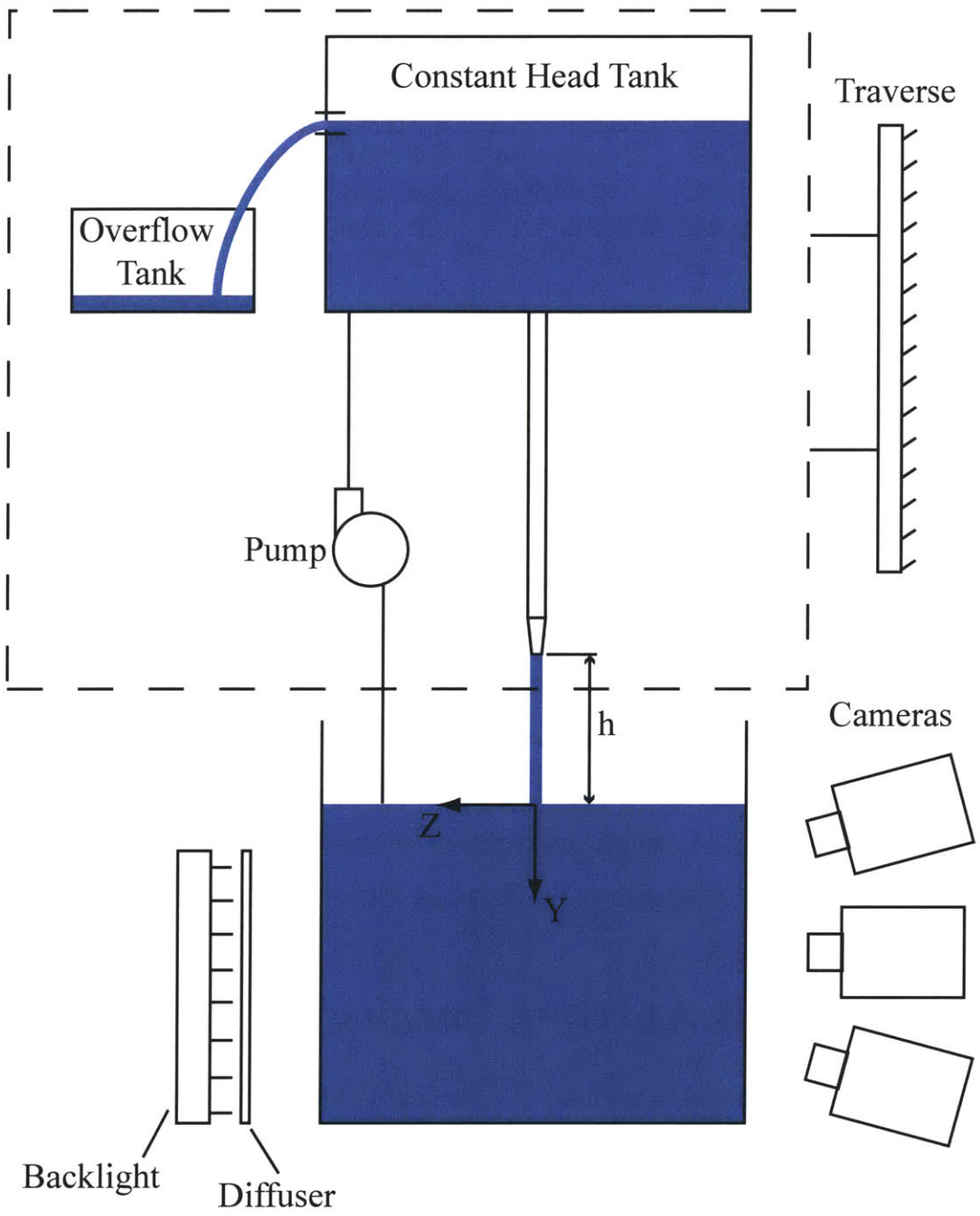


Figure 5-3: Experimental setup for bubble-entraining free-surface piercing jets. The dashed box denotes the hardware that is mounted to the traverse.

Table 5.1: Physical parameters for the plunging jet experiment.

h (mm)	V_i (m/s)	D_i (mm)	Re	Fr	We	Mo
7	6.29	3.45	2.17×10^4	112	514	2.52×10^{-11}
21	6.17	3.59	2.21×10^4	119	544	2.52×10^{-11}
43	6.01	3.78	2.27×10^4	128	589	2.52×10^{-11}

cameras were mounted on a custom 80/20® frame which allowed the cameras to be angled and thus the field of view (FOV) of each camera to be overlapped. All cameras were equipped with Nikon Nikkor 50 mm lenses set to f/8 in order to keep the entire volume of interest in focus; this is a pre-requisite for the synthetic aperture refocusing [18]. Cameras were synced to a master camera and each acquired 1024 pixel × 1024 pixel, 12-bit images (later saved as 8-bit tiff images) at a rate of 1000 frames per second (fps). A white backlight and diffuser provided illumination for the flow. Backlighting causes the edges of bubbles to appear dark against the white background [2, 19]. Setting the exposure time to 0.1 ms (10% of the total frame length) afforded enough light without causing bubbles to streak within the image (due to over exposure). A sample image from the central camera of the array (pre-processed for contrast enhancement) is shown in Figure 5-5.

5.3 Synthetic Aperture Imaging

Synthetic aperture imaging is a subset of light field imaging, which captures as many light rays emanating from a scene as possible [20, 21]. Synthetic aperture refocusing is a post-capture algorithm that reprojects in-focus images from multiple cameras onto planes in the imaged scene and recombines the images to create refocused images; objects that lie on these focal planes appear sharp while objects not on this plane

Figure 5-4: Arrangement of nine high speed Photron Cameras on a custom 80/20® frame.

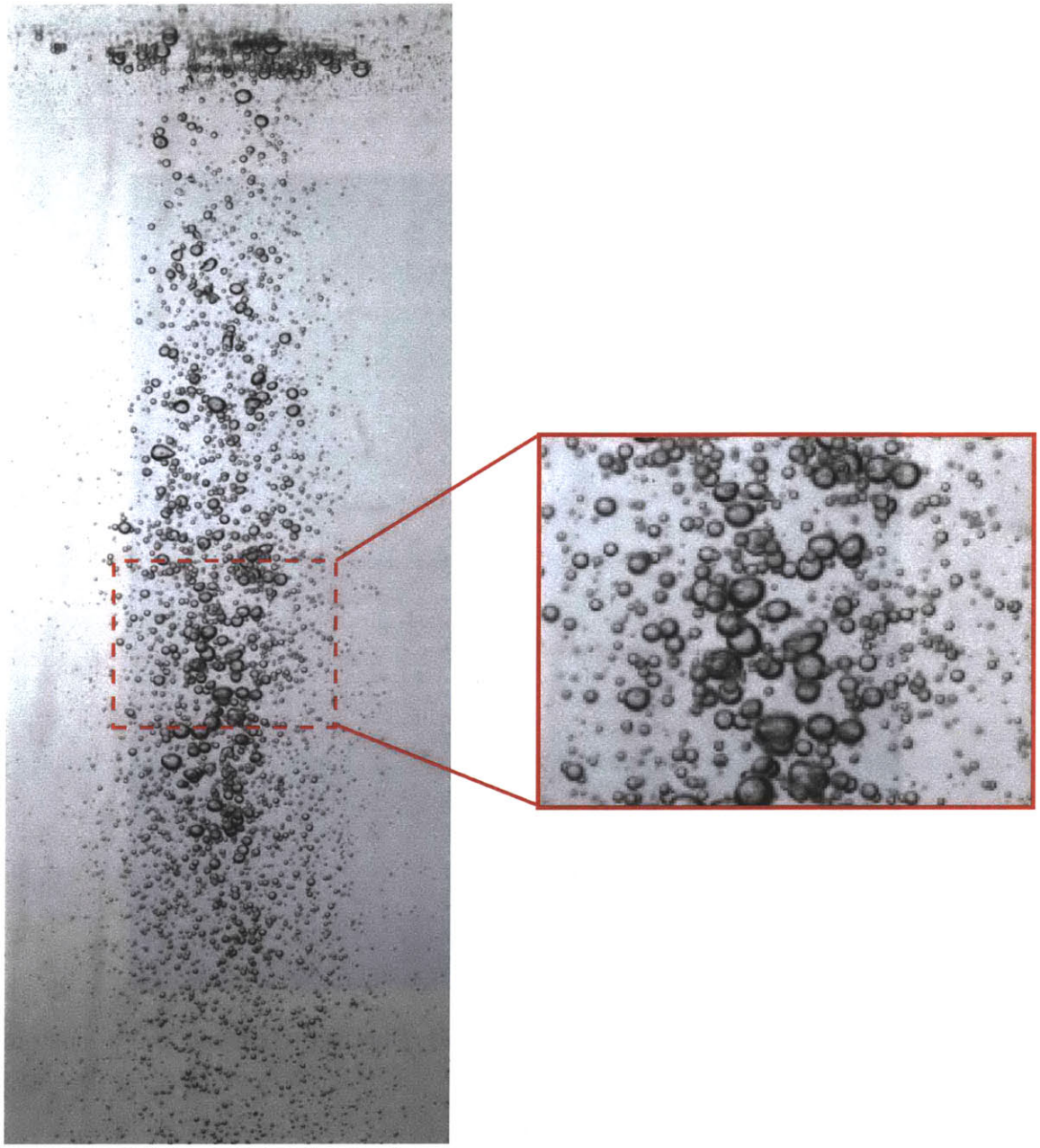


Figure 5-5: Sample pre-processed image from the central camera of the array from the 6.25 mm nozzle test at jet height $h = 21$ mm.

appear blurred [20, 21]. The application of synthetic aperture refocusing to three-dimensional particle image velocimetry (3D SAPIV) was described thoroughly in our previous work(Chapter 2 and [18]), and many of the same principles apply here. In general, the first step in SA refocusing involves establishing mapping functions between the camera image planes and world coordinates

$$\mathbf{u}_j^i = F\left(\mathbf{X}_j \; ; \; p^i\right) \tag{5.5}$$

where $\mathbf{u}_j^i$ is the 2x1 vector of the j^{th} image point coordinates in the i^{th} camera, $[u_j, v_j]^T$, $\mathbf{X}_j$ is the 3x1 vector of the j^{th} world point coordinates, $[X_j, Y_j, Z_j]^T$, p^i is a set of parameters defining the model of the i^{th} camera and F defines the form of the model. Using this model, each image from N cameras is reprojected onto k focal planes. The reprojected image from the i^{th} camera onto the k^{th} focal plane will be referred to as $I_{FP_{ki}}$. In [18], the refocused images were generated by taking the arithmetic mean

$$I_{SA_k} = \frac{1}{N} \sum_{i=1}^{N} I_{FP_{ki}} \; . \tag{5.6}$$

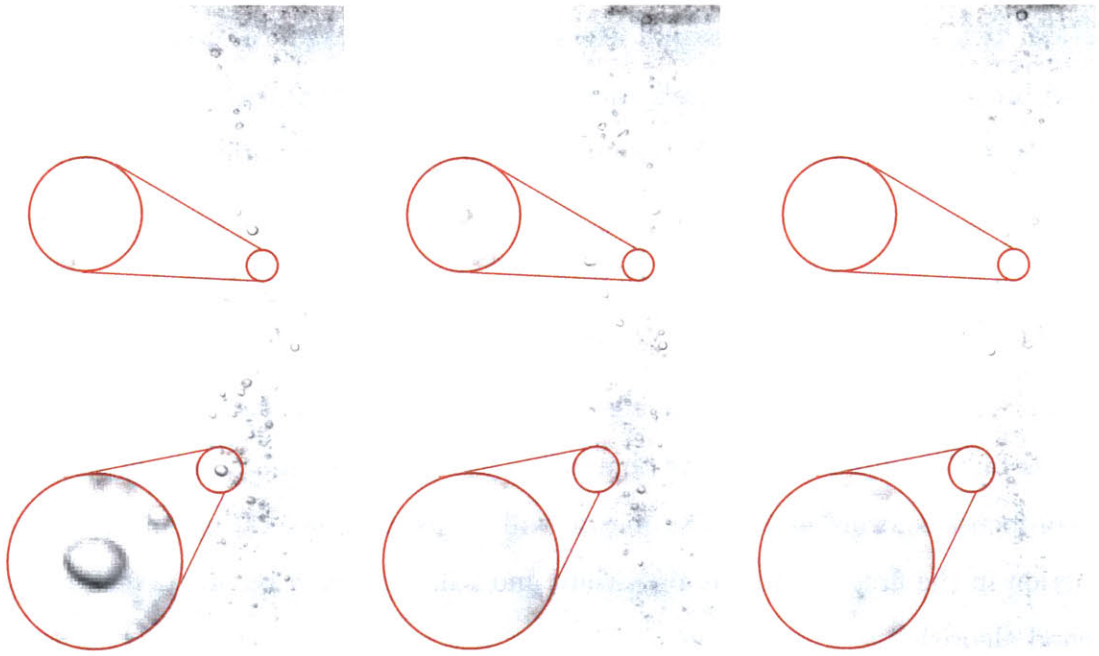
In Tomographic PIV [22], reprojected rays are multiplied together to enhance the signal-to-noise ratio. However, if any camera has an occluded view of an object (i.e., zero or a very small value for a pixel), multiplication brings the entire reprojected voxel to zero. Instead of using Equation 5.6, the signal-to-noise ratio can be improved by refocusing the images according to

$$I_{SA_k} = \prod_{i=1}^{N} \left(I_{FP_{ki}}\right)^n \tag{5.7}$$

as suggested by [23], where n is an exponent between 0 and 1. This allows for enhancement of the signal-to-noise ratio without letting any camera with an occluded view of an object to prevent that object from being refocused, because a small number raised to an exponent between 0 and 1 is non zero. Equation 5.7 was applied to all data in this chapter using $n = 1/3$, a value found through experimentation.

Many options exist for the specific form of F, and the parameters are normally found via a calibration procedure. In the present paper, a precision machined calibration plate traversed through the target volume with Z location increments of 5 mm provided calibration targets. The average calibration in pixels/mm from the center camera image of the plate at the Z location farthest from the cameras was used to convert the reference geometry of the calibration plate from mm to pixels (herein, 4.1 pixels/mm). Second-order polynomial fits served as the mapping functions between image coordinates and reference coordinates on each Z calibration plane, and linear interpolation was used to find the polynomial fits on Z planes between each calibration plane, as has been used in [24, 25, 18]. The spacing between focal planes was set to 0.2 mm, which was chosen to be large enough such that adjacent planes were not redundant. For more information on focal plane spacing, see [18].

Refocused images at depths -10, 0 and 10 mm are shown in Figure 5-6 for a single time step from the 6.25 mm nozzle test at height 21 mm. All images were preprocessed to enhance contrast prior to application of the refocusing algorithm. The red circles on each image highlight locations where a certain bubble is in focus on one depth plane, but blurred from view on other depth planes.



(a) Z = -10 mm (b) Z = 0 mm (c) Z = 10 mm

Figure 5-6: Synthetic aperture refocused images at (a) Z = -10 mm, (b) Z = 0 mm and (c) Z = 10 mm. The red circles on each image highlight locations where a certain bubble is in focus on one depth plane, but blurred from view on other depth planes.

5.4 Feature Extraction

It is evident that SA refocusing is a powerful technique for isolating the depth location and in-plane features of bubbles in a dense flow, but the features must be extracted from the focal stack in order to quantify certain parameters. In this study, we are interested in measuring the bubble sizes and three-dimensional locations. In this section, two algorithms for feature extraction are presented: a "minimum image" algorithm suitable for less optically dense flows, and a "local feature" algorithm for more dense flows. Each is based on a novel metric for defining bubble features defined next.

5.4.1 Bubble Features

Prior to detailing the steps of the algorithm, a metric is first developed that distinguishes bubbles from noise (note that noise refers to both measurement noise and out-of-focuse information)[1]. The metric will be used in generating an attenuation function in the first extraction algorithm, and will serve as a bubble detector in the second algorithm.

In observing the raw bubble images formed in backlighting experiments, the distinguishing feature of bubbles are clearly the dark edges. Therefore, one can define a "gradient feature", $f(g)$, to help define bubbles, which comes from a set of possible gradient features: $f \in \{ |g_x| , |g_y| , \sqrt{\mathbf{g}^t\mathbf{g}} \}$. Similarly, higher order derivatives are also possible, but we restrict ourselves to the first order in this work. In particular we consider a gradient feature in the x and y components defined as the absolute value of the components of the in-plane gradient

[1]The bubble feature metric was co-developed with Dr. Sai Ravela of MIT EAPS Department.

$$f(g_x(i,j,d)) = |g_x(i,j,d)| = \left|\frac{\partial I(i,j,d)}{\partial x}\right| \tag{5.8}$$

where the in-plane gradient is calculated on the d^{th} depth layer of the focal stack and $f(g_y(i,j,d))$ is defined similarly. In the present implementation of the algorithm, the gradients are calculated using centered finite differences, but future implementations will account for scale changes in bubbles sizes when calculating the gradient A region of the refocused image containing a bubble will contain large values of f due to the presence of bubbles edges. Therefore we aim to construct a summary metric that indicates the likelihood that a region contains a bubble by calculating the expectation of samples of f taken from the local region. However, to calculate the expectation, we require a model for the probability distribution of f. For the types of images in this study, large values of gradient are far less likely, and therefore we propose a probability distribution based on a gradient feature potential defined as

$$L(f) = \exp\left(\frac{-f}{v}\right) \tag{5.9}$$

where $v > 0$. The probability distribution of gradient features can then be defined as

$$\rho(f) = \frac{L(f)}{\sum_f L(f)} \; ; \; 0 < \rho(f) \leq 1 \; . \tag{5.10}$$

For discrete images, $L(f)$ can never be zero and therefore neither can $\rho(f)$. A distribution with large probability corresponding to likely bubble regions (i.e., larger values of f) will be more amenable to the construction of an attenuation function that suppresses non-bubble features, which is an ultimate goal of the derivation of this metric. Thus, the distribution defined in Equation 5.10 is used to define a new

distribution that gives higher probability in regions of stronger gradients, which are assumed to be associated with in-focus bubbles. The "bubble feature potential" is defined as

$$\gamma(f) = \frac{1}{\rho(f)} \; ; \; 1 \leq \gamma(f) < \infty \tag{5.11}$$

and the bubble feature probability distribution is then given as

$$\psi(f) = \frac{\gamma(f)}{\sum_f \gamma(f)} = \frac{\exp(f/v)}{\sum_f \exp(f/v)} \; ; \; 0 < \psi(f) \leq 1 \; . \tag{5.12}$$

Now, we can return to the original goal of determining the likelihood that a bubble exists in a local region by calculating the expectation under ψ using samples of f taken from the local region. First, the values of f in the local region are rewritten in vector form as $F = [f(i - n_x, j - n_y, d), \cdots f(i + n_x, j + n_y, d)]^T$, where $l_x \times l_y = (2n_x + 1) \times (2n_y + 1)$ is the size of the local region. Then the expectation of f under ψ for the region centered at (i, j) is given by

$$E(i, j, d) = E_\psi[F] \equiv \sum_{k=1}^{l_x \cdot l_y} F_k \cdot \psi(F_k) \tag{5.13}$$

Inserting the definition of $\psi(f)$ into Equation 5.13 gives

$$E(i, j, d) = \sum_{k=1}^{l_x \cdot l_y} F_k \left(\frac{\exp(F_k/v)}{\sum_k \exp(F_k/v)} \right) \propto \sum_{k=1}^{l_x \cdot l_y} F_k \exp(F_k/v) \tag{5.14}$$

where the constant $\sum_k \exp(F_k/v)$ in the denominator is dropped. Finally, inserting the gradient feature used herein gives

$$E_x(i, j, d) = \sum_{k=1}^{l_x \cdot l_y} |g_x|_k \exp(|g_x|_k/v) \tag{5.15}$$

where $E_y(i, j, d)$ is defined similarly and v is yet to be defined.

5.4.2 Minimum Image Bubble Feature Extraction Algorithm

The minimum images algorithm is applied to the data from jet heights $h = 7$ mm and 21 mm. The steps of the proposed algorithm for feature extraction are listed below, and each is explained in further detail thereafter:

1. Apply attenuation function to suppress noise and make refocused images good candidates for feature extraction.
2. Generate minimum images over depth windows and store corresponding depth maps.
3. Generate edge map of minimum images and label possible features.
4. Apply either Hough transform (for "large" bubbles) or grayscale threshold (for "small" bubbles).
5. Validate resulting bubbles.

The **first step** in the algorithm involves making the focal stack images better candidates for feature extraction; namely, by reducing the amount of out of focus noise and leaving the sharply refocused bubbles. Here, we aim to construct a noise attenuation function that suppresses noise and rewards regions with large gradients based on the bubble feature metric defined above. The metric gives a single value for each depth layer and is therefore calculated on a windowed version of the refocused volume to emphasize local features. For the data presented here, the window size is $l_x \times l_y \times n_Z$ voxels, where l_x and l_y are length scales chosen to be 7 voxels in the

present study and n_Z is the depth dimension of the entire refocused volume in voxel units.

First, the windowed refocused volume (I_o, assumed to be grayscale) is normalized

$$I(i,j,k) = \frac{I_o(i,j,k) - \bar{I}_o}{\sigma_{I_o}} \tag{5.16}$$

where σ_{I_o} is the standard deviation of intensity values in the windowed refocused volume. For each depth layer of the windowed refocused volume, Equation 5.15 is applied to the x and y components with $v = 2/\sigma_I^2$, where σ_I^2 is the variance of the normalized windowed volume, which is equal to one. A more appropriate definition is to find v from a fit to sample data; this was not done herein, but will be in future applications. Clearly, large values of gradient magnitude result in large values of E_x and E_y. The final metric is calculated as the maximum of the metric applied to x-direction data and y-direction data, $E(d) = \max(E_x(d), E_y(d))$, for each depth. The metric is calculated in windows covering the entire volume; no window overlap is used in the present study. Because the metric is a summation of functions of the local gradient, it rewards *regions* with large gradients; for example, the edges of bubbles. Therefore, the length scale, l, should be chosen such that portions of bubbles are within the window rather than a single edge pixel, for example. However, making the length scale too large will retain unwanted information, and thus the choice of l is somewhat experimental.

Figure 5-7 shows E as a function of depth for one window of the bubble volume. Slices from the original refocused volume at three depths are shown on the graph. The first image corresponds to the largest peak, and as can be seen a bubble is contained within the window. The second image contains a smaller bubble that has less well defined edges. This bubble does cause a spike in E, but the peak is not as

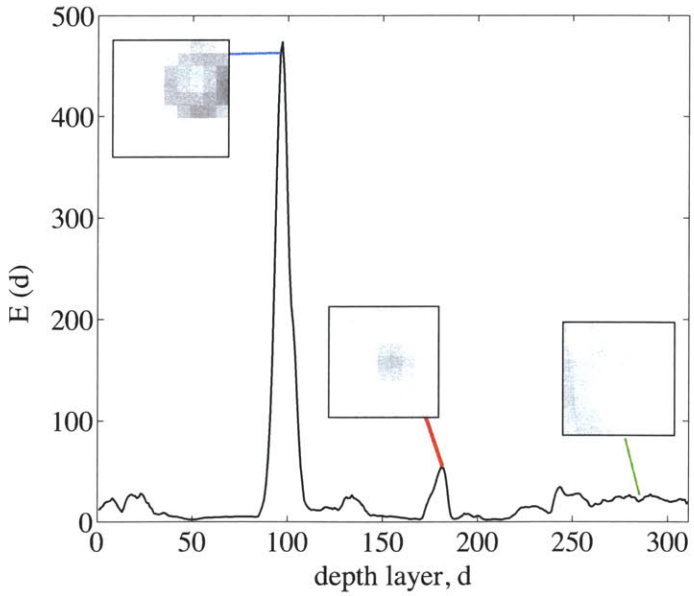


Figure 5-7: E as a function of depth for one window of the bubble volume. Slices corresponding to peaks are shown.

large because the gradients are not as strong. Finally, the third windowed image is more representative of the background noise, and does not increase E much over the background level.

The question still remains: what form should the attenuation function take? Here, we use the discrete cumulative distribution function (cdf) of E, which naturally assigns low levels a value near zero and high levels a value near one. The cdf values corresponding to each calculated E are assembled into a matrix; the nodes of the matrix are at the in-plane centers of the windows at each depth layer. Finally, the cdf matrix is interpolated onto a matrix of equal size to the refocused volume and the interpolated matrix is multiplied point-by-point by the refocused volume.

The **second step** of the algorithm involves isolating the focus plane of the bubbles. Rather than applying a focus metric to every refocused image, a "minimum" image is calculated over a depth window, similar to other previous works [4]. Many

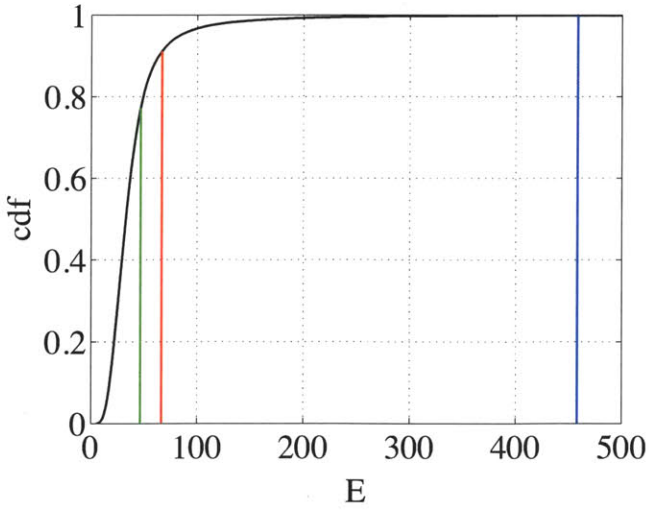


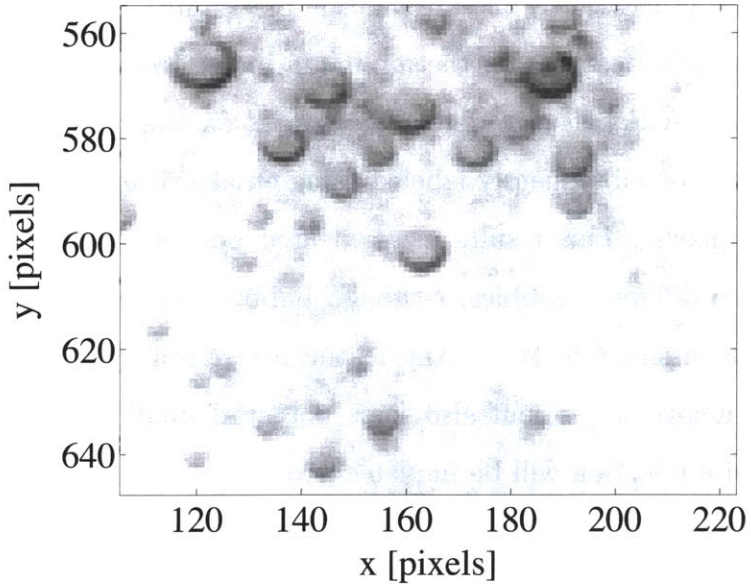
Figure 5-8: Discrete cumulative distribution function of E with the depths from Figure 5-7 marked.

metrics exist for defining the minimum image (e.g., gradients, entropy), but because the bubbles edges are defined by local minima in intensity, the intensity value itself serves as the metric in this study. At each pixel, the minimum intensity value is found over the depth of a stack of refocused images and the depth corresponding to each minima is recorded into a depth map. Other authors (e.g. [4]) have used this approach over the entire focal stack for lower density flows, but the larger bubble density in the present study would lead to many overlapping bubbles in the minimum image. Therefore, the focal stack is windowed to yield stacks of smaller depth, and a minimum image and depth map is generated for each window. Herein, a window size of 50 voxels yielded minimum images with relatively few instances of bubble overlap, but also was large enough to contain the depth of focus of even large bubbles. Sample portions of minimum images are shown in Figure 5-9; Figure 5-9(a) is a minimum image generated from the original refocused volume and Figure 5-9(b) is generated

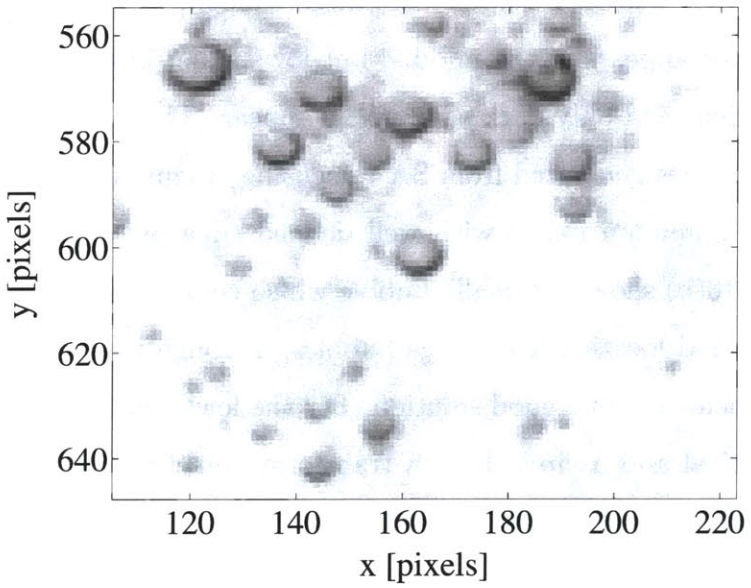
from focal planes processed by the attenuation procedure.

Labeling potential features on the minimum images constitutes the **third step** of the algorithm. A Sobel edge detector operates on the minimum images to define edges, which are subsequently labeled using an algorithm based on connectivity of neighboring pixels. This results in segmented portions of the minimum image corresponding to different bubbles, although bubbles very near to each other may get grouped into a single feature. Any feature described by three or less pixels is removed; this removes noise, but also some potential small bubbles. In the future, more robust noise rejection will be implemented.

In the **fourth step** of the algorithm, geometric properties of the bubbles are extracted from the labeled features. As discussed in [26] for backlighting of bubbles with an incoherent source, if bubbles are large enough, the edges will appear dark and a bright spot appears in the middle, but for smaller bubbles the middle bright spot is less evident and only dark edge pixels appear. This effect carries through to the minimum images generated from SA refocusing. Figure 5-10(a) shows a "large" bubble from a minimum image with well defined edge pixels and a bright center, while Figure 5-10(b) shows a "small" bubble which consists only of dark edge pixels. To extract size and location from large bubbles, a Hough transform or similar edge pixel voting scheme offers a good solution. For the lower jet heights the bubbles are primarily spherical and a circle Hough transform detection scheme similar to that described by [27] is applied. To determine ellipse parameters with higher accuracy, the vote weighted average of the parameter value with the highest votes and the parameter values on either side is taken for each parameter. This approach is not suitable for small bubbles that lack sufficient edge definition. For each small bubble, a local region is thresholded using the mean grayscale value (similar to [26]), to generate a binary image with zeros defining the bubble and ones defining the background. The

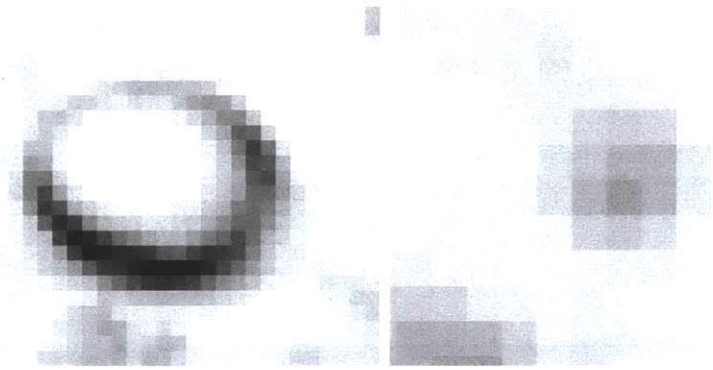


(a)



(b)

Figure 5-9: Minimum image using the original refocused volume (a), and minimum image after application of the attenuation function (b).



(a) (b)

Figure 5-10: Example of a large bubble with well defined edge (a), and small bubble with less edge definition (b).

centroid and equivalent diameter formed by the dark pixels are calculated from the binary image. To distinguish between large and small bubbles in this study, the equivalent diameter of each feature from the initial labeling is calculated and any feature with diameter less than four pixels is considered to be a small bubble. As a **fifth step** to the algorithm, any detected bubble entirely within a larger bubble is considered to be noise and is removed and overlapping bubbles are merged.

The contours of bubbles detected on the same portion of the minimum image as shown in Figure 5-9(b) are drawn in blue in Figure 5-11. Clearly, there is a high rate of success in bubble detection. In more dense and noisy areas of the minimum images, the instances of noise detected as bubbles and missed detection of overlapping bubbles increases, but overall the results are very good. Figure 5-12 shows results from various portions of different minimum images, again indicating the overall high success rate of the algorithm.

For each detected bubble, values from the depth map inside the bubble contour and within one pixel distance of the contour are used to define the depth location

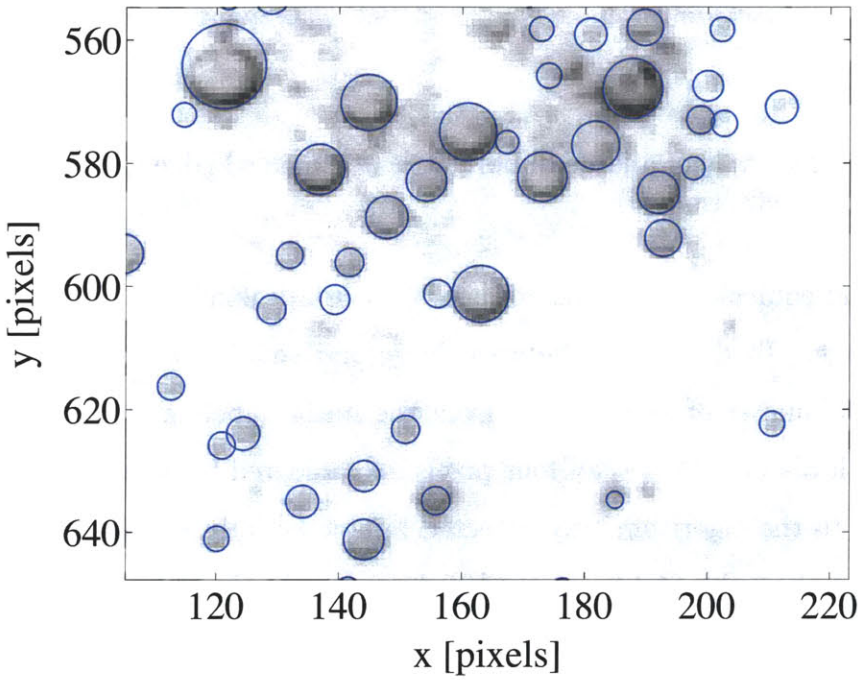


Figure 5-11: Result of the bubble feature extraction algorithm. Bubble contours are overplotted on the same portion of the minimum image shown in Figure 5-9(b).

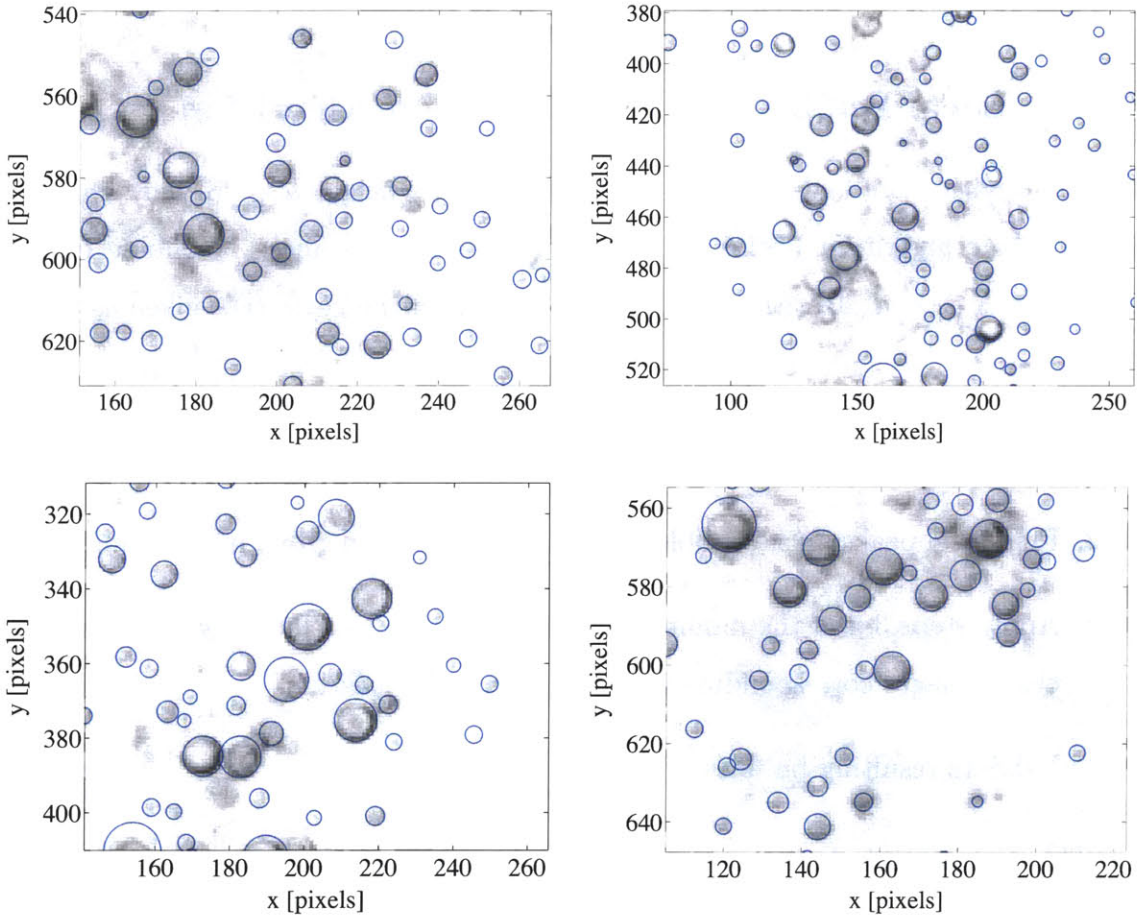


Figure 5-12: Result of the bubble feature extraction algorithm. The minimum image algorithm displays good success rate and accuracy in defining the bubbles.

of the bubble. The depth is calculated as the weighted mean of contributing pixels from the depth map. Conservatively, the resolution of the depth location is $\pm$ 0.1 mm (half the focal plane spacing), but the resolution is likely even better due to the weighted average.

5.4.3 Local Bubble Feature Extraction Algorithm

For cases with more and/or larger bubbles (as for jet height $h = 43$ mm), the minimum image algorithm begins to break down as the instances of bubble overlap increases. A second algorithm is described here, with the keys steps given as:

1. Apply two passes of the bubble feature metric & attenuation algorithm described in the previous sections.
2. Find local peaks in the bubble feature metric (E) and group neighboring peaks.
3. Apply steps 3-4 of the minimum image algorithm to a local region of the refocused images corresponding to each group of E peaks.
4. Validate resulting bubbles.

This algorithm is built around the fact that the bubble feature metric itself is an excellent bubble detector, as indicated by the signal-to-noise ratio for bubbles shown in Figure 5-7. In the **first step** of the algorithm, the attenuation function based on the bubble feature metric is applied to the refocused images, and then again to the attenuated images to further reduce noise. In this implementation of the algorithm, the windows for calculating E are overlapped by 50%. The **second step** involves identifying local peaks in the bubble feature metric that are above a noise threshold determined from investigation of the data. Peaks are then grouped

together by finding connected regions in which each peak is within one half window in X and Y and ± 10 voxels in Z of another peak; grouped peaks define potential local features.

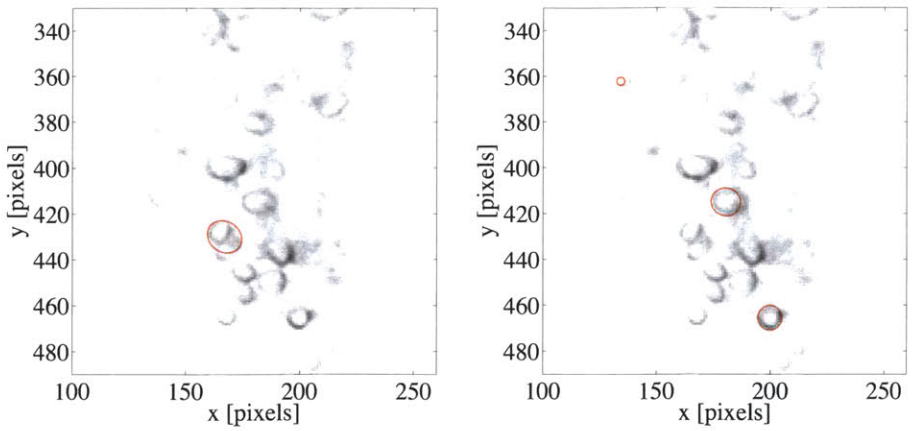
In the **third step**, steps 3-4 of the minimum image algorithm are applied to local regions of the refocused images defined by the grouped peaks. The refocused image from the average depth of the grouped peaks is used. A grouping may contain multiple bubbles, and therefore the Hough transform from the previous section is first applied to each labeled local region to split the labeled features if multiple bubbles exist (indicated by multiple high values in the Hough accumulator). The bubbles for the higher jet height tend to take on oblate spheroidal shapes with elliptical cross sections, and thus the circle Hough detector becomes unsuitable for defining the bubble parameters. Therefore, an elliptical variation of the Hough transform described by [28] is implemented. Finally, validation of the resulting bubbles is carried out in the **fourth step**. Any bubble with no E peak within the contour defined in the $X-Y$ plane, or with ± 5 voxels of the bubble depth in Z is removed. Also, bubbles with a minor to major axis ratio below a certain threshold were removed, as these typically constituted detection of partial edges of bubbles. Finally, bubbles entirely within other bubbles are removed, and partially overlapping bubbles are either merged or the smaller bubble is removed if the dimensions of the larger bubbles are more that 1.5 times bigger (typically, the smaller bubble is noise or a partial edge in this case).

Figure 5-13 shows bubble contours plotted on their associated attenuated refocused images for five adjacent depths. The results indicate that the algorithm is successful in isolating the plane of sharpest focus for the bubbles and estimating accurate contours. In some cases, the contours are influenced by part or all of another bubbles in the local region, which influences the estimation of the contour, but overall the results are very promising. In future work, the algorithm will be made more

robust to nearly overlapping bubbles. Figure 5-14 shows bubble contours plotted over various portions of minimum images calculated with a 50 voxel depth window. To reiterate, the minimum images are not used in this algorithm, they are simply used here as a tool for qualitatively evaluating the results. Overall, the algorithm performs very well, even in the higher density areas, but is subject to noise in very dense regions with nearly overlapping bubbles, as just discussed. The algorithm is also occasionally prone to "locking on" to bubbles that have very strong edges on one part of the bubble, but not all the way around the bubble perimeter. In some cases where it appears as though two different contours are describing the same bubble, the results are misleading. For example, in the lower right image of Figure 5-14, the two contours pointed out by the blue arrows appear to describe a single bubble, but review of the individual refocused planes clearly shows two distinct bubbles at different depths, with accurate contours. This underscores the need to use the local bubble feature extraction algorithm as opposed to the minimum image algorithm for more optically dense flows.

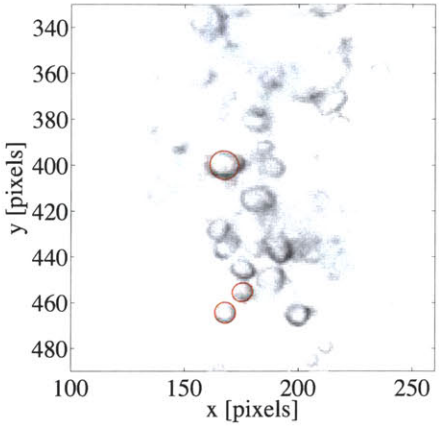
5.5 Results and Discussion

The ensuing sub-sections discuss calculation of global and local bubble distributions, bubble penetration depth and local features of the air concentration for the cases study in the present work. Comparisons are drawn to existing results and scaling laws where available, and a new scaling law is proposed for concentration as a function of depth. To facilitate explanation of the results, some definitions are first made. The total concentration at a given depth is defined as

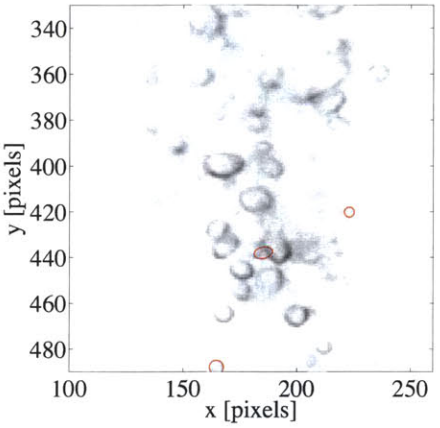


(a) $Z = 1.6$ mm

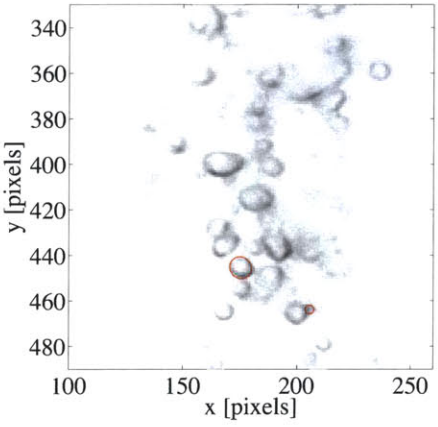
(b) $Z = 1.4$ mm



(c) $Z = 1.2$ mm



(d) $Z = 1.0$ mm



(e) $Z = 0.8$ mm

Figure 5-13: Results of the local bubble feature extraction algorithm on isolated refocused images.

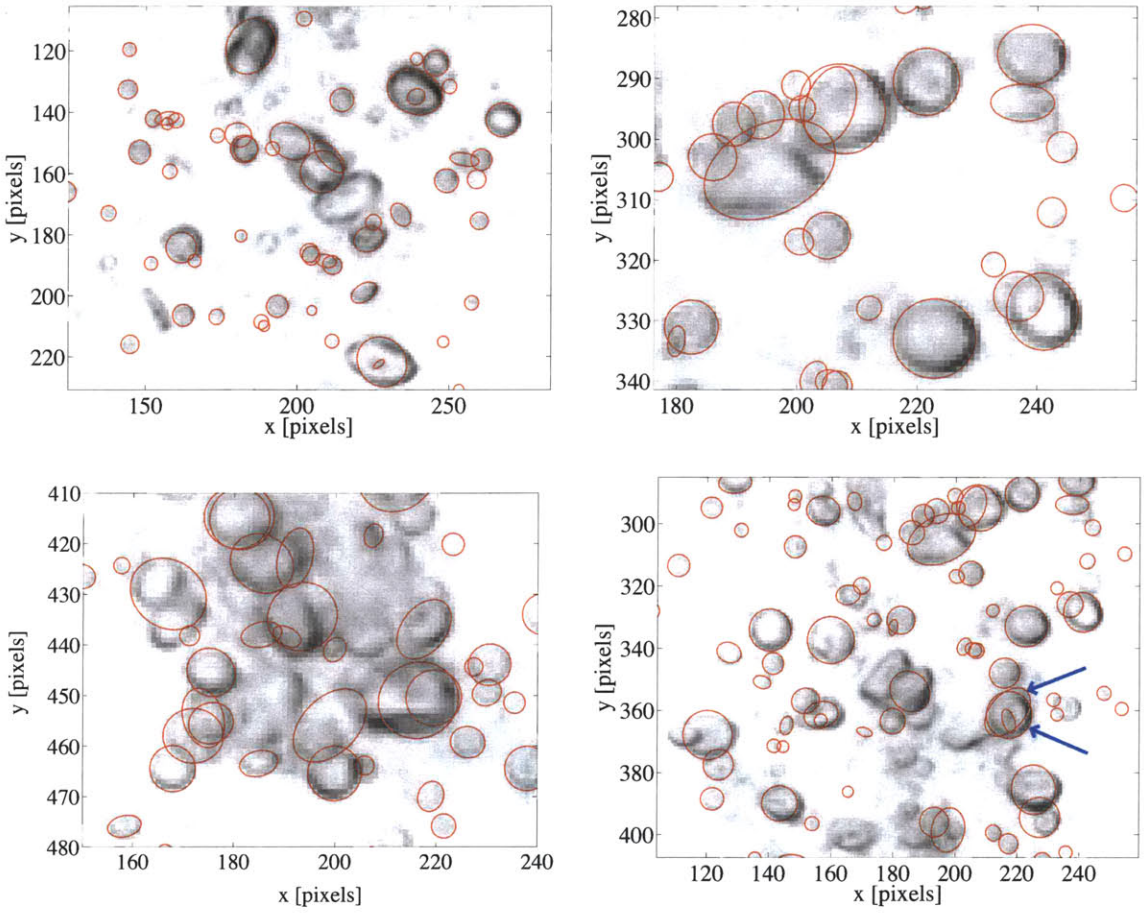


Figure 5-14: Result of the local bubble feature extraction algorithm plotted over minimum images with a 50 voxel depth window.

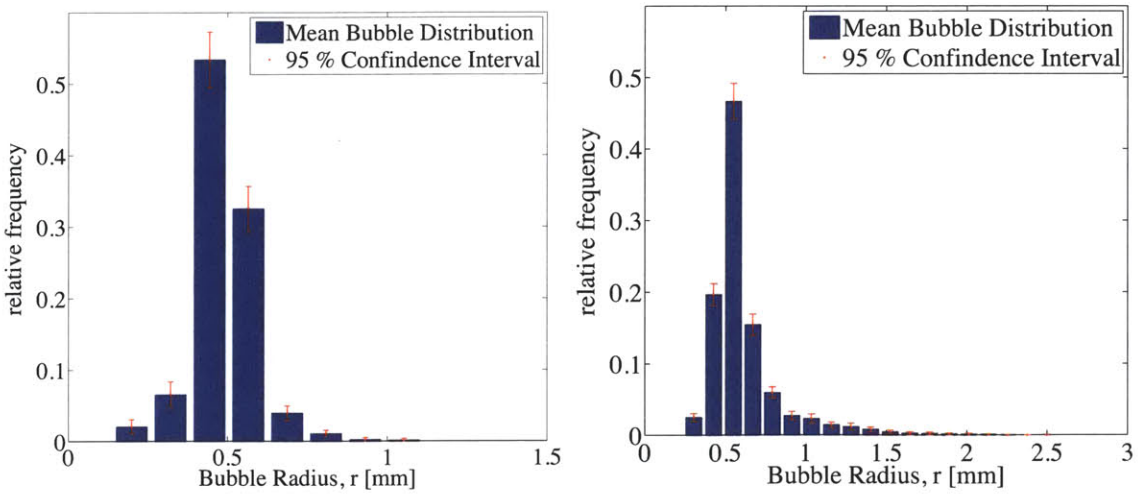
$$C_Y(Y) = \int_0^{\infty} \int_0^{2\pi} C(Y,R) d\theta dR = 2\pi \int_0^{\infty} C(Y,R) dR \tag{5.17}$$

where it is assumed that the bubble distributions are axisymmetric ($C = C(Y,R)$). The maximum value of total axial concentration is defined as $C_{Y_o} = \max(C_Y(Y))$ and occurs at depth Y_o. The maximum concentration at a given depth, for all values of R is defined as $C_{max}(Y) = \max\left(C(R)\Big|_Y\right)$

5.5.1 Global Bubble Distributions

The SA refocusing and bubble detection algorithms were applied to data from each of the jet heights. For jet heights $h = 7$ mm and $h = 21$ mm, the minimum image version of the extraction algorithm applied to 50 time steps yielded 3D bubble size distributions. Alternatively, the local peak detection algorithm was applied to 49 time steps of the $h = 43$ mm jet height data, due to the increased amount of occlusion.

Figures 5-15 and 5-16 displays the time-averaged distribution of the calculated bubble radii for $h = 7$ mm and $h = 21$ mm and ellipse dimensions for $h = 43$ mm, respectively, with error bars representing the 95% confidence intervals. The range of radii is 0.138 mm to 2.38 mm for the $h = 7$ mm and 0.244 mm to 2.44 mm for $h = 21$ mm data. The low end of the range for $h = 7$ mm data corresponds to just over half of the size of a pixel back-projected into physical space and thus represents the low end limit on resolution for this implementation of the algorithm. For the $h = 43$ mm case, the range of dimensions (major axis a or minor axis b) is 0.1946 mm to 4.32 mm. The distributions reflect the expectation that more small bubbles remain in the volume than large bubbles due to buoyancy. However, as discussed in Section 5.5.2, the distribution of "large" and "small" bubble is more subtle than one would expect. It should be pointed out that the global bubble distributions reported

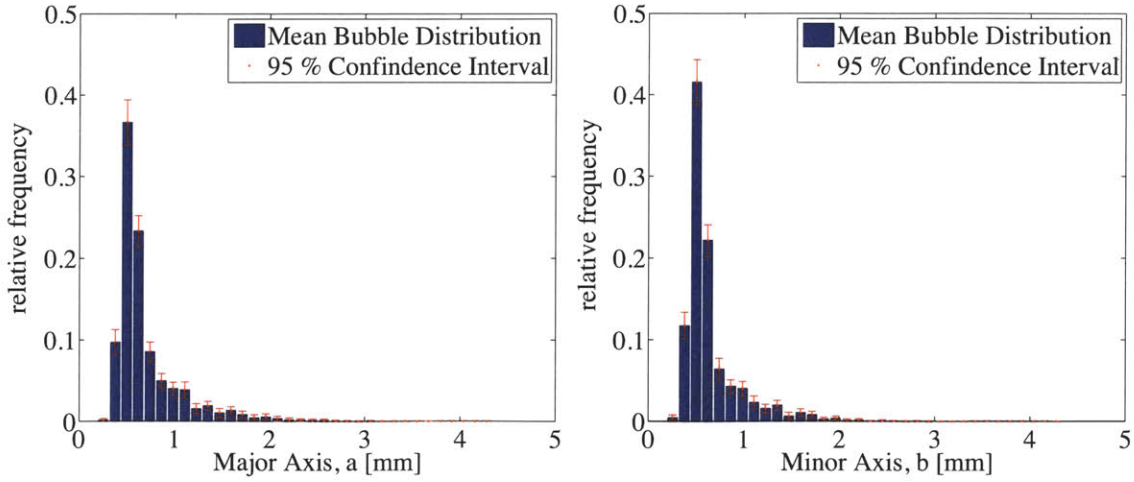


(a) $h = 7$ mm (b) $h = 21$ mm

Figure 5-15: Distribution of the calculated radii of the bubbles with 95% confidence intervals.

in prior work were measured using either a global acoustic measurement technique or the integrated results from single point sensors [1]. The bubble distributions reported herein represent the first instantaneous distributions in which the size and 3D location of each bubble is known, to the knowledge of the author. The total concentration in the downward flowing bubble cone, C_{tot}, are reported for each case in Table 5.2. While the total concentration is not that large (0.34% for the highest jet height), the concentration reaches nearly 8% locally in some regions of the flow for $h = 43$ mm.

Figure 5-17 shows the bubble diameter distributions on a single plot, as well as a zoomed view of the distributions (for elliptical bubbles, diameters are approximated as $d = 2a$). Figure 5-17(c) - reprinted from Chanson & Manasseh [1] - is a bubble distribution measured using an acoustic sensor with experimental parameters corresponding to $h = 5$ mm, $d_i = 25$ mm and $V_i = 3.9$ m/s. In general, the shape of



(a) $h = 43$ mm, major axis (a)

(b) $h = 43$ mm, minor axis (b)

Figure 5-16: Distribution of the calculated radii of the bubble with 95% confidence intervals.

Table 5.2: Characteristics of the entrained bubble region for various jet heights.

h (mm)	C_{tot} (%)	Y_p (mm)	D_{eq} (mm)	α
7	0.02	211.5	87	11.6
21	0.23	160	61.6	10.9
43	0.34	120	48.1	11.3

the distributions are in very good qualitative agreement, though the data from the present study contain much larger counts of small bubbles ($d <\approx 1.5$ mm).

The plots of three-dimensional spatial location and size of the bubbles in the flow field shown in Figures 5-18 - 5-20 yield further insight into the 3D distribution of bubbles. Marker sizes are scaled by the bubble diameter (or by $2a$ for $h = 43$ mm data). Figure 5-18 corresponds to jet height $h = 7$ mm; clearly the bubble entrainment rate is low and the entrained bubbles are small. More interesting is the higher jet height case of $h = 21$ mm shown in Figure 5-19. The bubble distribution

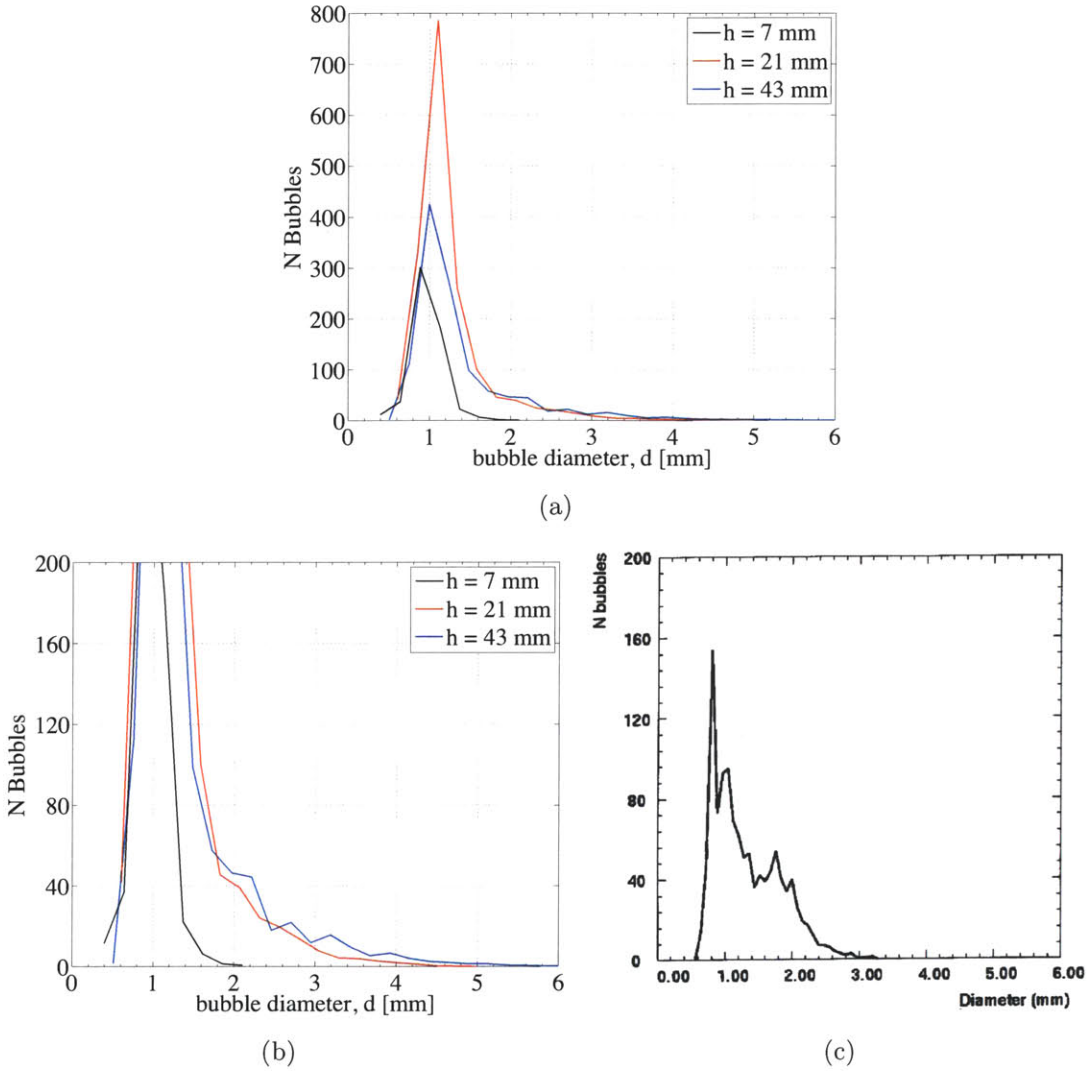


Figure 5-17: Bubble distributions for each jet height (a), zoomed view of the bubble distributions (b) and results from Chanson & Manasseh [1] corresponding to $h = 5$ mm, $d_i = 25$ mm and $V_i = 3.9$ m/s.

reflects the well-known conical volume [15, 16], which contains the downward moving bubbles and relatively stationary bubble cloud. As can be seen, the large bubbles present in the flow reside close to the axis of the nozzle where high momentum fluid entrains the bubbles. Smaller bubbles are more uniformly distributed throughout the flow and penetrate deeper. Ultimately, buoyancy overcomes drag and bubbles (large and small) rise toward the surface outside of the conical volume. Some bubbles rise all the way to the surface, while some are re-entrained in the high-momentum region of the jet. Figure 5-20 shows a similar picture, but the bubble distribution now contains significantly more large bubbles. Also, the bubbles do not appear to penetrate as deep as for the $h = 21$ mm jet height, which is potentially surprising given the increased jet momentum. However, this effect is well-documented in the literature [15, 16] and is discussed further in Section 5.5.2

5.5.2 Penetration Depth and Local Bubble Distributions

The penetration depth of bubbles entrained by a circular plunging jet has been discussed by several authors, notably Clanet & Lasheras [15] and Suciu & Smigelschi [16]. As described by Clanet & Lasheras, the entrained bubbles are transported along the axis of the jet as well as radially and larger bubbles appear deeper in the flow due to coalescence; this behavior is reflected in Figures 5-19 - 5-20. Ultimately, the bubbles become so large that buoyancy overcomes drag, and the bubbles rise to the surface. Visually, this is marked as a sharp transition beyond which large bubbles are not observed in the flow. However, smaller bubbles continue to descend beyond this transition. Clanet & Lasheras define the penetration depth as being one half of the maximum bubble cloud width above this transition point. Clearly, the definition of penetration depth is somewhat qualitative. The quantitative measures presented

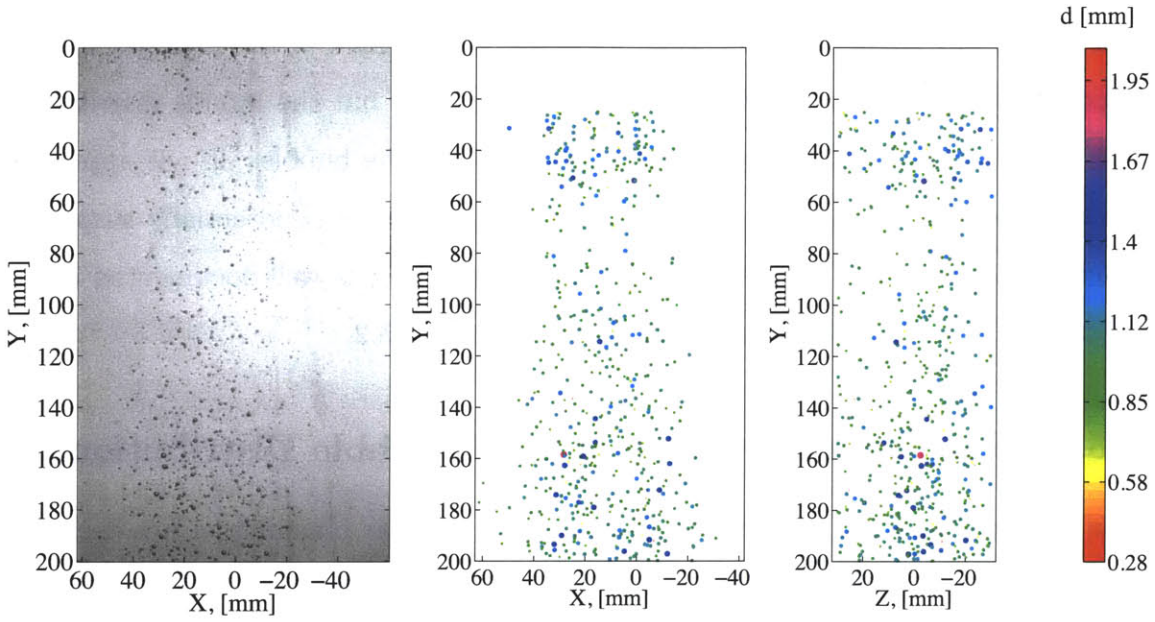


Figure 5-18: From left to right: raw image from center camera of the array re-projected onto the $Z = 0$ mm focal plane, $X - Y$ view of bubble size and spatial distribution, $Z - Y$ view of bubble size and spatial distribution for jet height $h = 7$ mm, and color bar corresponding to the two bubble distribution plots.

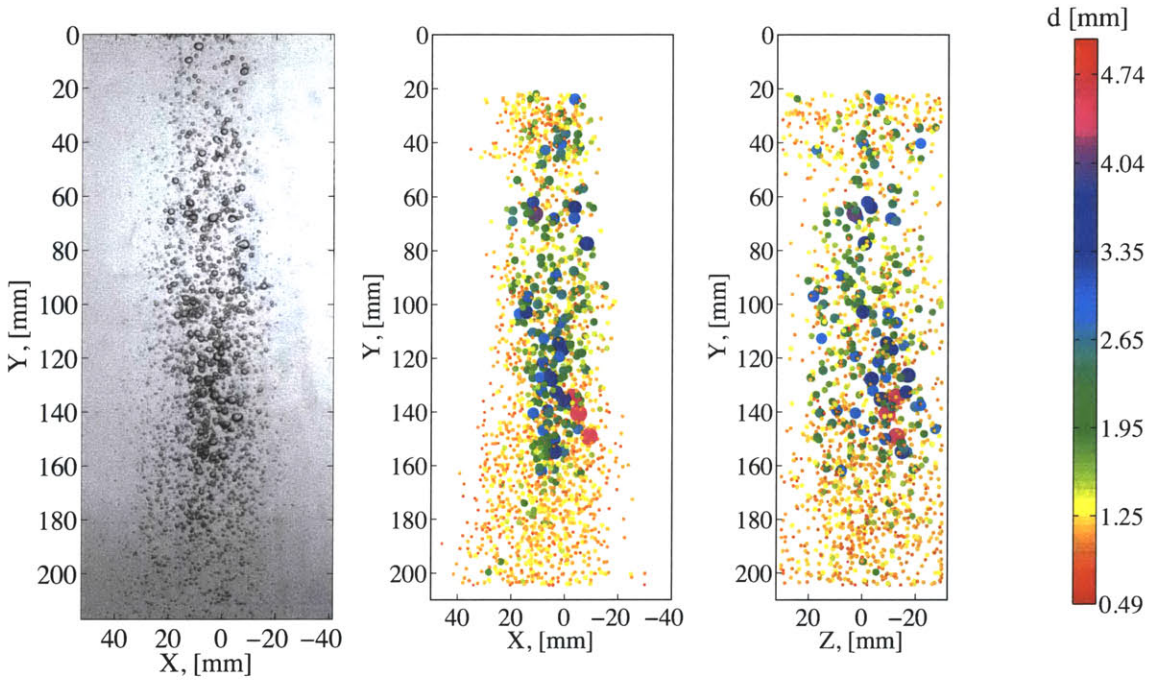


Figure 5-19: From left to right: raw image from center camera of the array re-projected onto the $Z = 0$ mm focal plane, $X - Y$ view of bubble size and spatial distribution, $Z - Y$ view of bubble size and spatial distribution for jet height $h = 21$ mm, and color bar corresponding to the two bubble distribution plots.

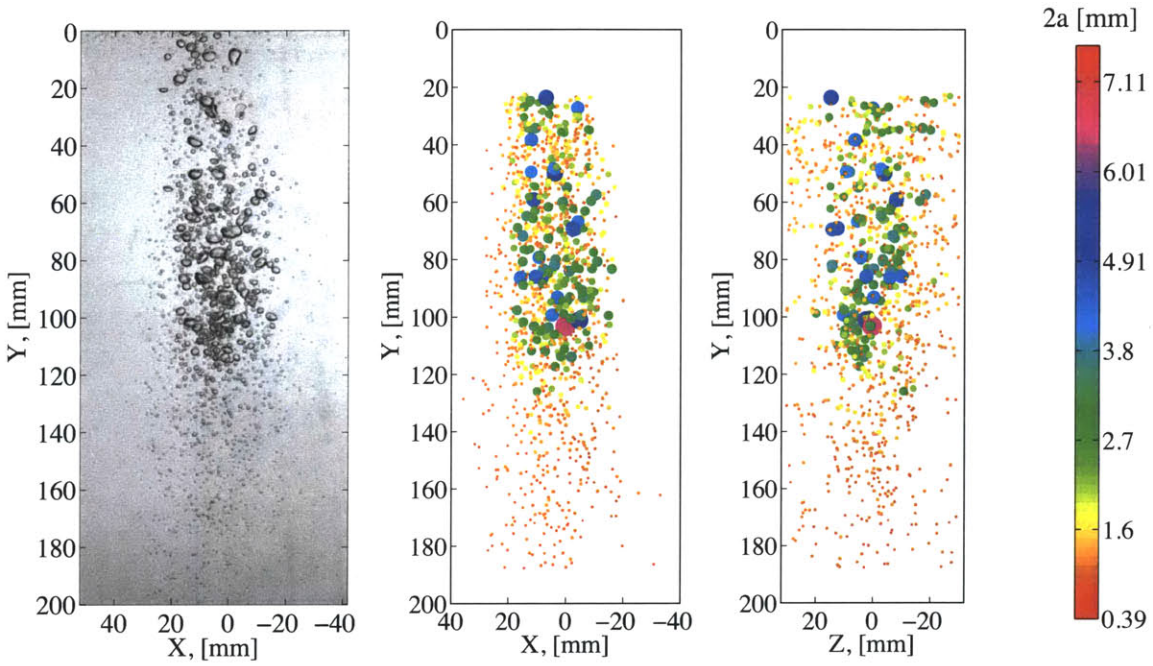
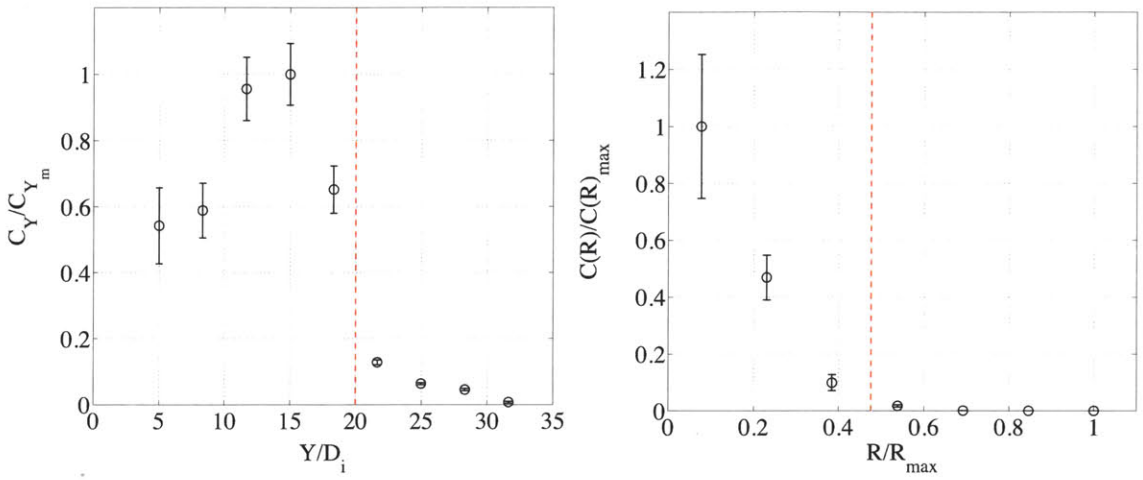


Figure 5-20: From left to right: raw image from center camera of the array re-projected onto the $Z = 0$ mm focal plane, $X - Y$ view of bubble size and spatial distribution, $Z - Y$ view of bubble size and spatial distribution for jet height $h = 43$ mm, and color bar corresponding to the two bubble distribution plots.

herein provide a more concrete definition of the penetration depth. In order to more quantitatively define the penetration depth, each volume is discretized into bins along the axial (Y) and radial dimensions. Bins are spaced by $\Delta Y = 20$ mm and $\Delta R = 8$ mm in the axial and radial dimensions, respectively. This facilitates calculation of the concentration locally; a sharp drop in concentration should occur at the transition point discussed in [15]. Figure 5-21(a) shows the concentration as a function of depth normalized by the impact diameter for jet height $h = 43$ mm data; C_o is the maximum concentration across all depths. The plot reveals a definitive drop in concentration at $Y/D_i = 21.6$. The penetration depth is calculated as the average of the depth corresponding to the sharp drop in concentration and the depth of the previous bin (one bin closer to the free-surface); this depth is marked by the dashed red line. The reason for calculating the depth as the average depth of these two bins is due to the fact that the bin corresponding to the dramatic decrease in concentration includes none of the prominent cloud of bubbles. Because we want the penetration depth to correspond to the bottom of the bubble cloud, the average of the two depths is taken. For the jet height $h = 7$ mm data, no drop in concentration is observed indicating that the penetration depth occurs outside of the field of view of the refocused volume. For this case, the penetration depth is estimated from raw image sequences, but for the other two cases the drop in concentration is observed and the penetration depth calculated accordingly.

Suciu & Smigelschi [16] investigated the penetration depth for a range of jet heights and observed the behavior seen in the present work in which increasing jet height leads to a reduction in penetration depth. Ultimately, the penetration depth reaches a limiting value, called Z_c in [16], which is constant even for increasing jet height. Figure 5-22 shows a reprint of Figure 3 from Suciu & Smigelschi of dimensionless penetration depth as a function of dimensionless jet height with data from



(a) (b)

Figure 5-21: (a) Concentration as a function of depth for the jet height $h = 43$ mm data. Concentration is calculated in bins centered around $X - Z$ planar cuts through the jet. (b) Concentration as a function of radial displacement from jet centerline for the jet height $h = 43$ mm data at $Y/D_i = 18.3$. Error bars show range of one standard deviation from the mean.

the present study plotted as well. For data from the present work, the red markers denote normalization of the jet height by impact diameter and the blue markers denote normalization by nozzle diameter (as was done in [16]). The penetration depth calculated for jet height $h = 43$ mm is considered the limiting penetration depth for the present study, although it is possible that the true value of Z_c is actually somewhat smaller; $h = 43$ mm was the highest jet height studied. Despite differences in the nozzle diameter, the data show excellent agreement with work of [16], indicating a consistent scaling of dimensionless penetration depth in the present work.

Calculation of the penetration depth enables measurement of the width of the bubble cloud (defined as equivalent diameter, D_{eq} in [15]) and subsequent calculation of the jet spreading angle. To define the equivalent diameter here, the concentration is plotted as a function of radial displacement from jet centerline for the Y bin above the penetration depth. Again, this is a more quantitative measure than proposed in prior literature [15] and the author recommends this quantitative definition of the width of the bubble cloud. Figure 5-21(b) shows this plot for the jet height $h = 43$ mm data at $Y/D_i = 18.3$. The equivalent radius is defined as the point at which the concentration decreases to 5% of the maximum at the given depth; linear interpolation is used to find the corresponding radius, which is marked by the red dashed line in Figure 5-21(b). Assuming the jet radius to be zero at the free-surface, the jet angle can be calculated from the equivalent diameter and the corresponding depth. Table 5.2 summarizes the penetration depth (called Y_p henceforth), equivalent diameter of the bubble cloud and angle of jet spreading (α) for each jet height. Previous studies have reported jet angles in the biphasic region in the range of 12-13^o [16, 15] . Although close to this value and the values reported in [15], the angles are slightly less for the cases presented herein. The most likely explanation for this is the variability in definitions of the equivalent diameter across measurement techniques.

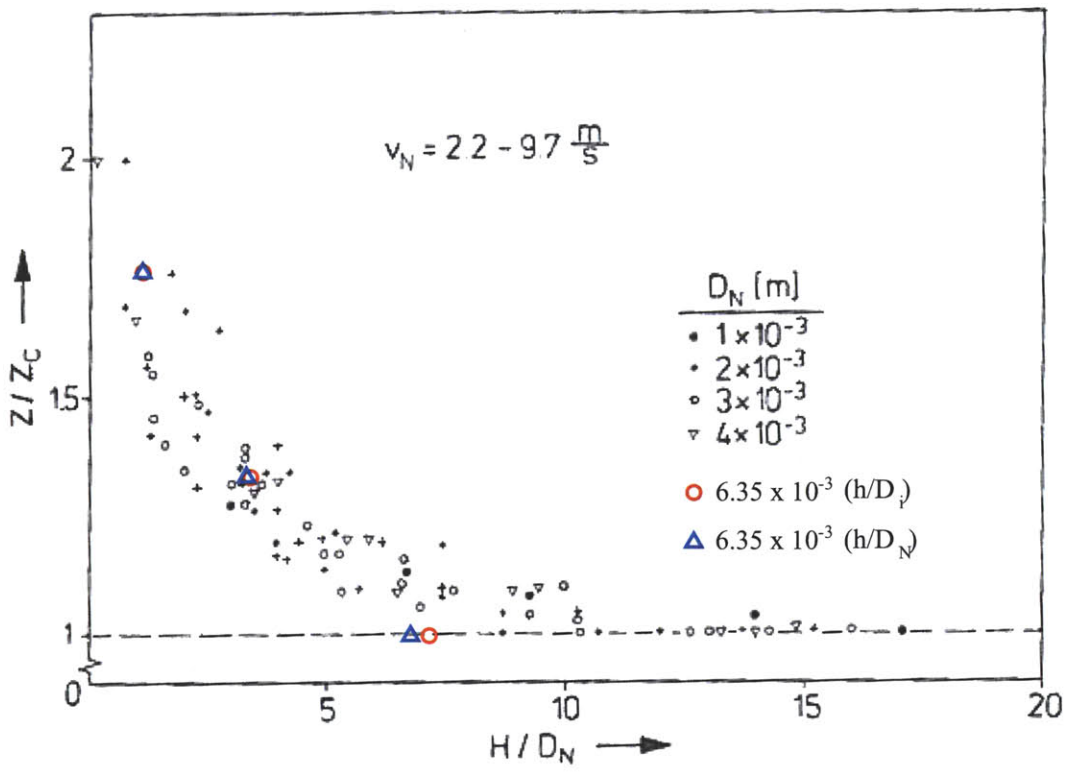


Figure 5-22: Dimensionless penetration depth as a function of dimensionless jet height. Original plot is reprinted from Suciu & Smigelschi [16]. Red and blue markers are data points added from the present work normalized by impact diameter and nozzle diameter, respectively. D_N is the nozzle diameter, and [16] defines the jet height as H (defined as h herein).

While the bulk features of the flow have been discussed and are in excellent agreement with existing scaling laws, the nature of the transition region from large to small bubbles has not been discussed. This is the main thesis of the paper by Clanet & Lasheras [15]. Bubbles rise to the surface when buoyancy becomes larger than the drag force on the bubble. The equilibrium point occurs at the terminal velocity, U_T. As discussed in [15], if the terminal velocity increased monotonically with bubble size, then bubbles of decreasing size would appear at monotonically decreasing depths in the flow. However, at the transition point, the bubble size changes dramatically. This is due to the non-monotonic dependence of terminal velocity on bubble size [15], which was discussed for single bubbles by Maxworthy et al. [29]. Figure 5-23 shows a reprint of Figure 4 from Maxworthy et al. [29], which shows the terminal velocity of single bubbles as a function of bubble diameter at various Morton numbers. The terminal velocity first increases monotonically with diameter, then suddenly decreases to a minimum. The dashed red and blue lines are generated from data from the $h = 21$ mm and $h = 43$ mm cases, respectively. First, the maximum diameter the terminal velocity corresponding to the maximum diameter is found for each case. For jet height $h = 43$ mm, the maximum value of $2a$ is greater than the diameter corresponding to the minimum in terminal velocity, and thus the minimum U_T is used. The point at which the rising portion of the curve (Mo $= 2.72 \times 10^{-11}$) reaches the terminal velocity is then found; this point gives the cutoff for the largest bubbles expected to remain in the flow. For the $h = 21$ mm case, the distribution should contain bubbles with diameter up to 4.88 mm before the transition point, then only diameters below 1.05 mm below the transition point. Similarly, for $h = 43$ mm case, the largest bubbles pre-transition should correspond to $2a = 8.64$ mm, while after transition the sizes should also be limited to $2a \leq 1.05$ mm. Figures 5-24(a) & 5-25(a) show the bubble distributions in various axial bins

for $h = 21$ mm and $h = 43$ mm, respectively. The red line marks the penetration depth. Clearly, below the penetration depth, a significant drop of in the number of large bubbles exists. Figures 5-25(b) & 5-25(b) show the bubble distributions in the two bins below the penetration depth, with the red dashed line marking the expected transition size cutoff value. In each case, the transition size cutoff is located within the size bin containing the most bubbles. Some bubbles of larger size exist below the transition, but this is partially due to the bin below the penetration depth containing part of the bubble cloud above the transition region. The data indicate a clear and sharp decrease in bubble size consistent with the physical arguments of Clanet & Lasheras [15] and Maxworthy et al. [29].

The local distributions also provides evidence for a region of coalescence above the penetration depth. The distributions show an increase in the instances of larger bubbles (for jet heights $h = 21$ mm and $h = 43$ mm), which is visually manifested as the bubble cloud. In this region, bubbles tend to have an increased residence time due to a temporary balance of buoyancy and drag, and thus the chances for bubble coalescence increase. This coalescence region was observed in [15].

5.5.3 Air Concentration

To this point, no discussion of the spatial dependence of concentration has been made. It is desirable to understand both the axial and radial dependence of concentration for various initial conditions for many engineering problems. Although the three cases of jet height presented in this work represent a limited data set, the bubble distributions span a wide range from deeply penetrating, sparsely distributed bubbles ($h = 7$ mm) to more densely distributed bubbles with penetration depth approaching the limiting depth ($h = 43$ mm). It should be noted that prior to the making the

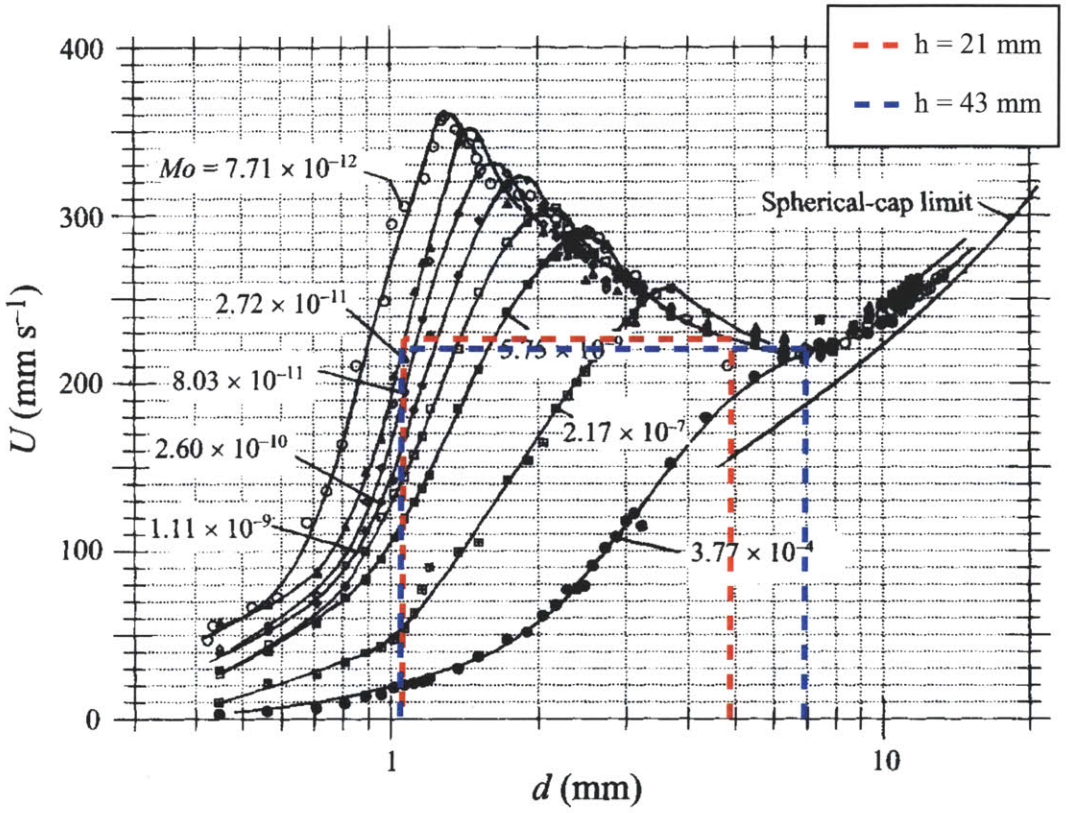
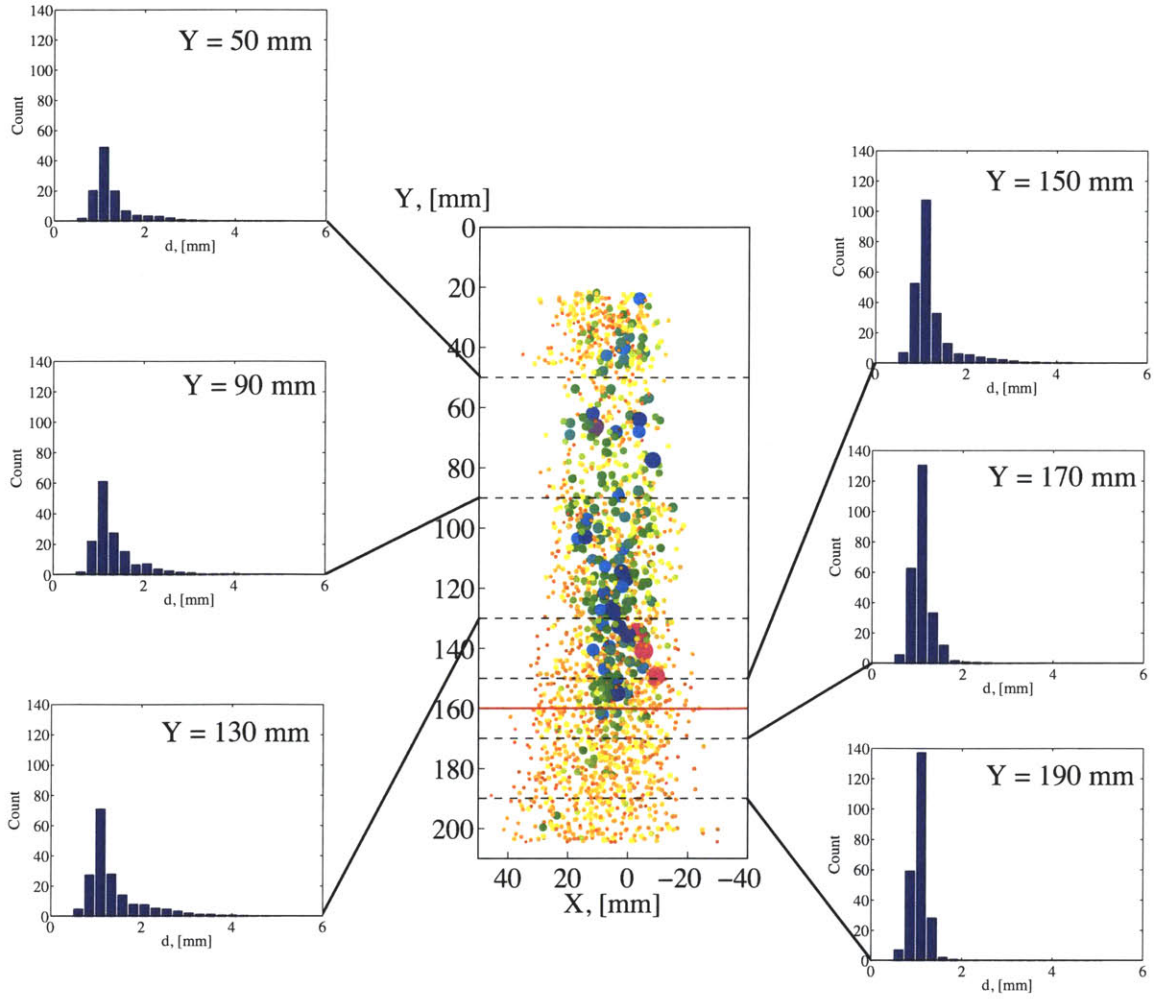
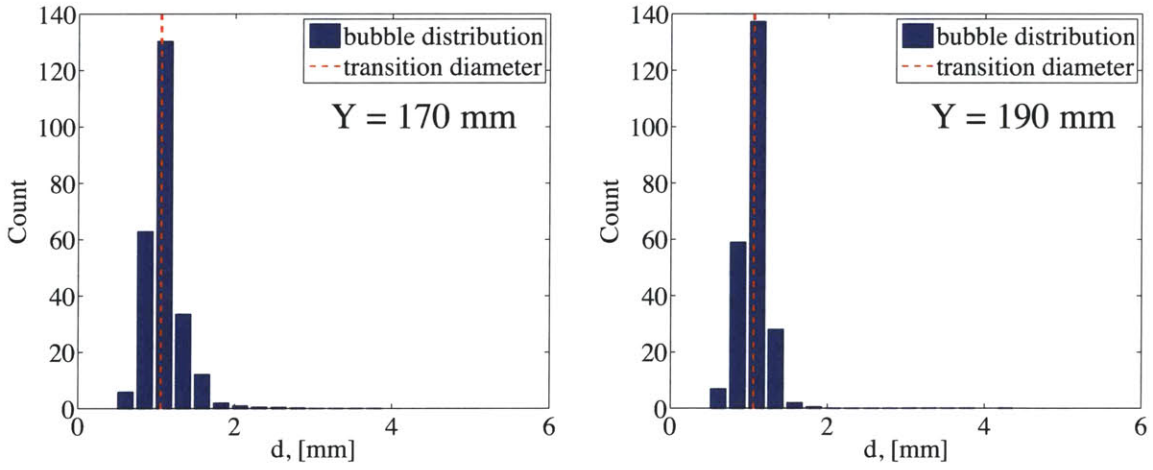


Figure 5-23: Original figure is reprinted from Maxworthy et al. [29], showing the terminal velocity of single bubbles as a function of bubble diameter at various Morton numbers. The red and blue lines represent traces from the largest bubble diameters in the $h = 21$ mm and $h = 43$ mm cases, respectively.

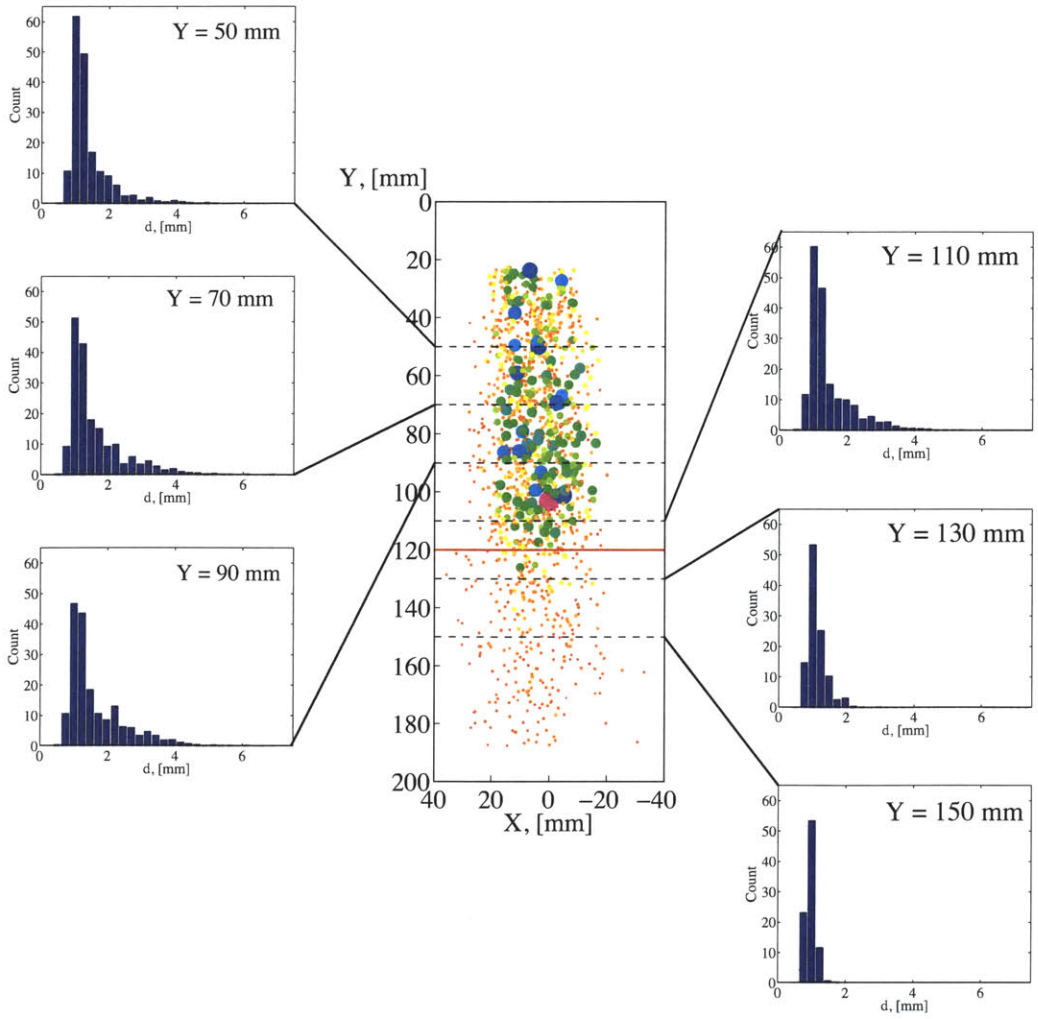


(a)

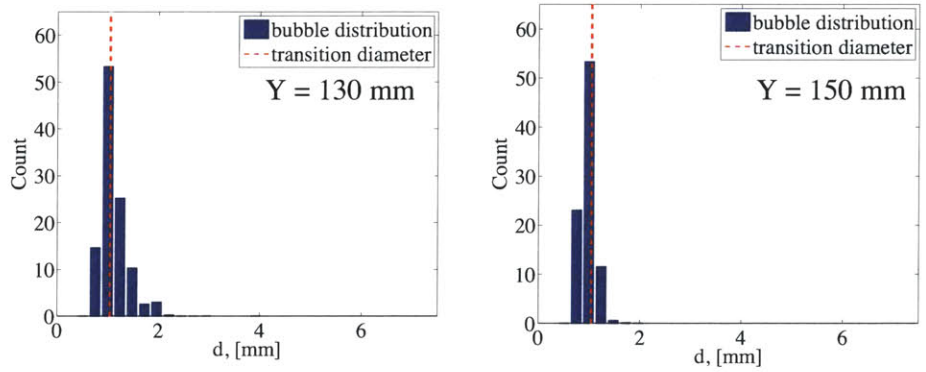


(b)

Figure 5-24: Bubble distributions in various axial bins for jet height $h = 21$ mm showing the transition in bubble size at the penetration depth.



(a)



(b)

Figure 5-25: Bubble distributions in various axial bins for jet height $h = 43$ mm showing the transition in bubble size at the penetration depth.

calculations presented in this sub-section, it was observed that the jet had a slight angle (1.5^o) with respect to the vertical. Therefore, all of the data were rotated about an axis parallel to Z and passing through the impact point to align the Y-axis of the data with the jet axis.

Figure 5-26(a) displays the total air concentration, C_Y, in each axial bin for the three jet heights. Only bubbles within the conical volume defined by the calculated cone angle (plus a 10% offset in radius at each depth) are considered, as these are expected to be the downward moving or stationary bubbles. The data appear similar only insofar as the concentration rises from near zero to a maximum value, then decreases rapidly below the penetration depth. We now propose the following scaling: the concentration is normalized by the maximum total axial concentration, C_{Y_m}, and the depth corresponding to the maximum concentration (Y_m) is subtracted from Y and the resulting values are normalized by the penetration depth (i.e., $Y^* = (Y - Y_m)/Y_p$). Figure 5-26(b) shows a scaled version of the data from the three cases, demonstrating the collapse of the data onto a single curve using the proposed scaling. Although more cases need to be studied to verify the scaling law across varying nozzle diameters and a wider range of velocities, the results are potentially impactful. Given the measurement of maximum concentration and corresponding depth, coupled with a measurement of the penetration depth, the entire total axial concentration profile, $C_Y(Y)$ can be predicted for any jet height, at least for the regime II entrainments types. Each of these measurements can be made with a single point conductivity probe, which is amenable to measurement *in situ* in engineering applications. Figure 5-27 shows the collapsed data fit with a Gaussian function plus an offset defined as

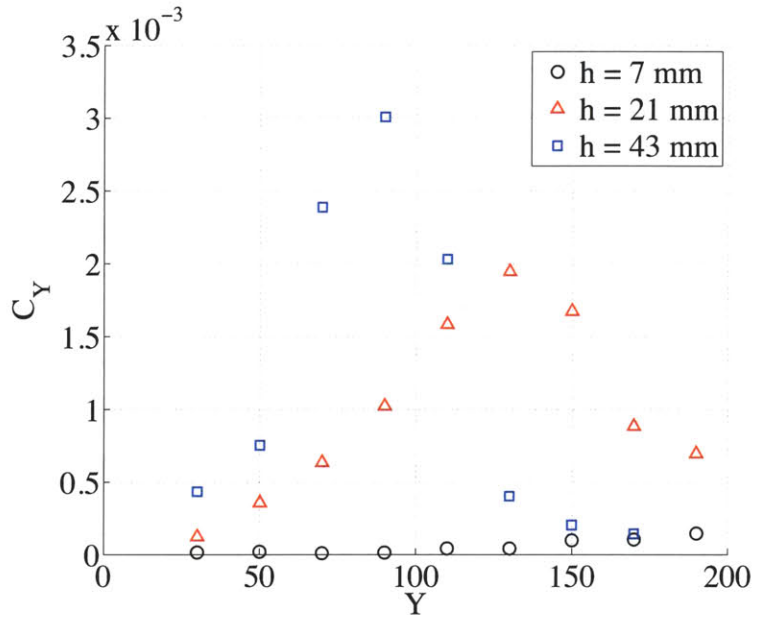
$$\frac{C_Y}{C_{Y_m}} = A \exp\left(\frac{-Y^{*2}}{2\sigma^2}\right) + (1 - A) \tag{5.18}$$

where the parameters of the best fit are found to be $A = 0.903$ and $\sigma = 0.1894$; the function captures the data with high accuracy.

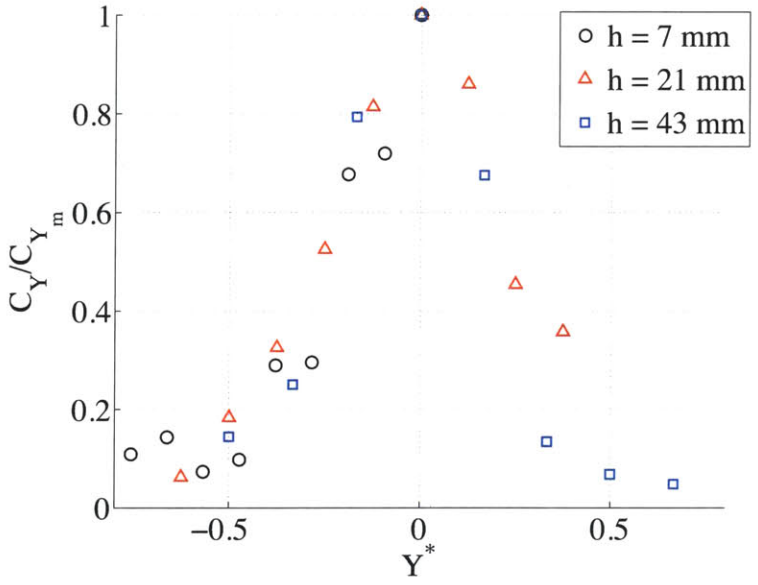
It is well-established that in the fully developed flow region, the concentration takes on a Gaussian dependence as a function of radial distance from the jet axis [30, 6, 5]. In fact, in a re-analysis of data from van de Donk [30], Bin [5] showed that all profiles collapse to a single curve given by

$$\frac{C(Y,R)}{C(Y,0)} = \exp\left(\frac{-0.69315\ R^2}{R_{0.5}^2}\right) \tag{5.19}$$

where $C(Y,0)$ is the concentration on the jet axis and $R_{0.5}$ is the radial coordinate corresponding to the point at which $C(Y,R)$ decreases to 50% of $C(Y,0)$. In the fully developed flow region, the maximum concentration $C_{max}(Y)$ should be the same as $C(Y,0)$. The raw concentration profiles for the present data are shown in Figure 5-28, and for some depths that the maximum value of concentration occurs not on the jet axis, but at one radial bin removed. This typically occurs at smaller Y locations, which indicates that the air diffusion layer could still be developing at these depths. More evidence is required to confirm this. Another possible explanation is that the shift in the maximum away from the jet centerline is caused by the slight angle of the jet, which itself would cause the concentration profile to be non-axisymmetric and/or may be causing the location of the jet centerline to be incorrectly calculated. Nonetheless, an alteration to equation 5.19 is proposed where $C(Y,0)$ is replaced with $C_{max}(Y)$ and $R_{0.5}$ becomes the radial coordinate corresponding to the point at



(a)



(b)

Figure 5-26: Total axial concentration as a function of depth for each jet height (a) and scaled total axial concentration as a function of scaled depth showing the collapse of all three cases onto a single curve for each jet height (b).

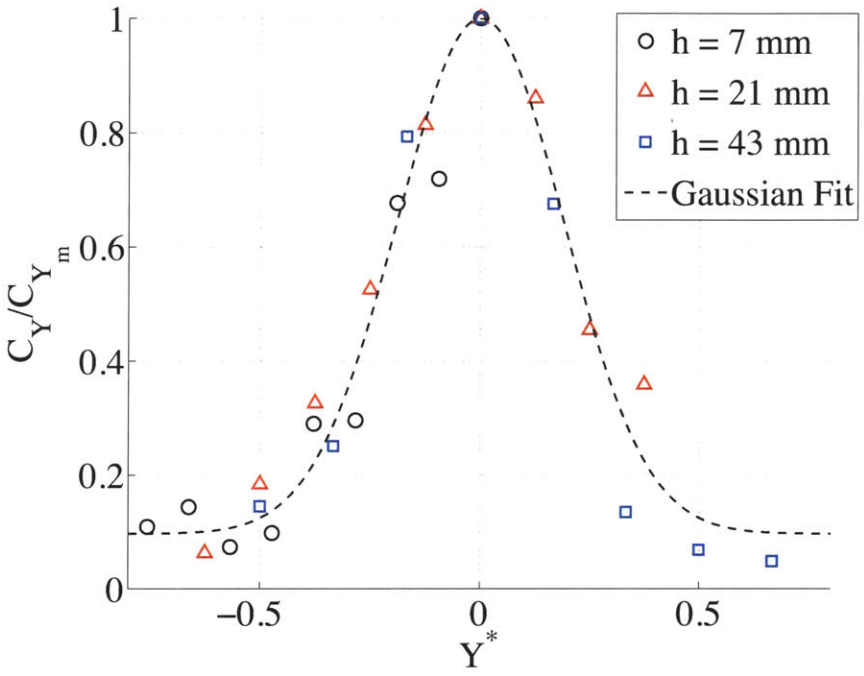
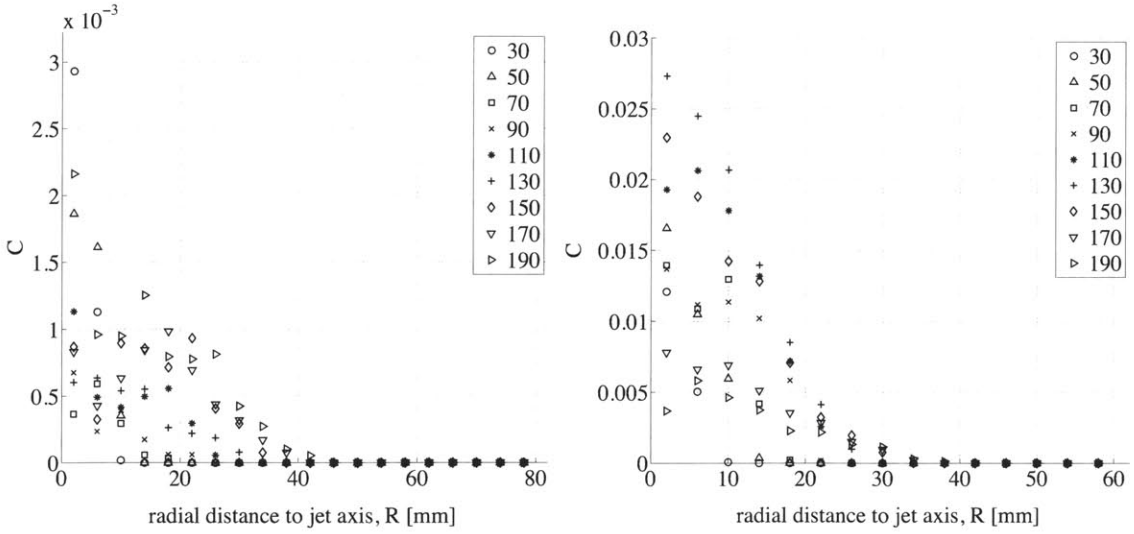


Figure 5-27: Gaussian fit to the scaled total axial concentration as a function of scaled depth.

which $C(Y,R)$ decreases to 50% of C_{max}) in an attempt to collapse the concentration profiles to a single curve. The results are shown in Figure 5-29; the data collapse very well onto the theoretical curve for the jet height of $h = 21$ mm, and reasonably well for the other two cases, although the slope predicted by the theory underestimates that shown by the data. In most cases, the data points that deviate far from the curve for $R/R_{0.5} > 2$ are caused by a small number being divided by a small number, and thus are likely due to measurement noise. Data points near $R/R_{0.5} = 0$ that deviate far from the curve correspond to depths where the maximum concentration does not occur on the centerline. It should also be pointed out that other authors [30, 6] presented observations that the maximum concentration decreases linearly as a function of depth, but the maximum value of concentration measured here increases at first with depth to a maximum, then subsequently decreases. More investigation is required to compare this with existing data in the appropriate regimes, but the present data clearly show a region where the maximum concentration increases with depth.

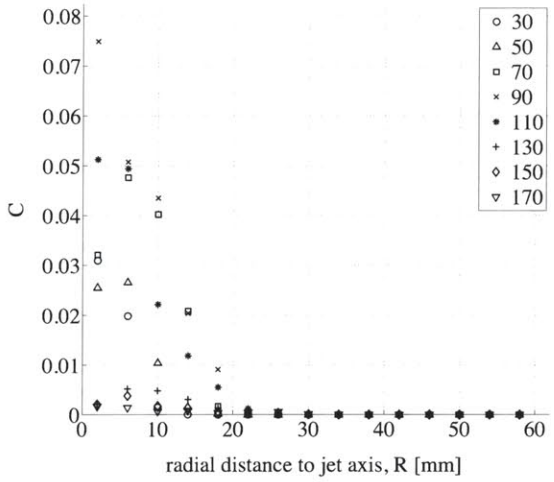
Combining the scaling proposed for the total concentration as a function of depth with this well-established scaling of concentration with radial distance provides a potentially powerful predictive tool, if the dependence of either C_{max} or $R_{0.5}$ on Y is known. First, consider writing the total concentration in terms of the radial concentration profiles as the integral

$$\begin{aligned} C_Y(Y) &= 2\pi \cdot \frac{1}{2} \int_{-\infty}^{\infty} C(Y,R) dR \\ C_Y(Y) &= \pi \int_{-\infty}^{\infty} C_{max}(Y) \exp\left(\frac{-0.69315\ R^2}{R_{0.5}^2}\right) dR \end{aligned} \tag{5.20}$$



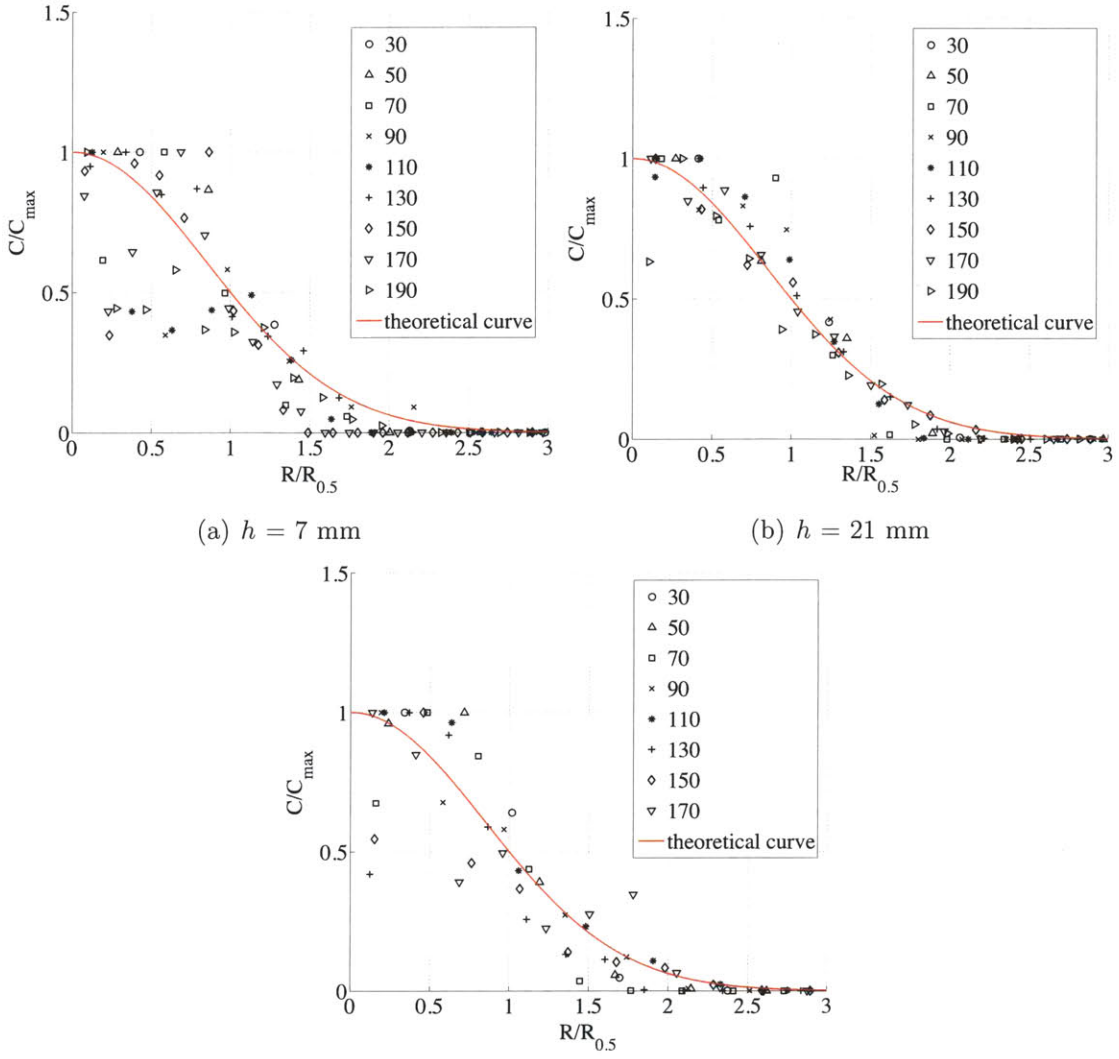
(a) $h = 7$ mm

(b) $h = 21$ mm



(c) $h = 43$ mm

Figure 5-28: Concentration as a function of radial distance from jet axis at various depths.



(a) $h = 7$ mm

(b) $h = 21$ mm

(c) $h = 43$ mm

Figure 5-29: Dimensionless concentration as a function of dimensionless radial distance from jet axis at various depths.

making a transformation of variables by letting $v = R/b$ where $b = R_{0.5}/\sqrt{0.69315}$, Equation 5.20 can be rewritten as

$$C_Y(Y) = \pi b\, C_{max}(Y) \int_{-\infty}^{\infty} \exp\left(-v^2\right) dv \tag{5.21}$$

where the integral is now the gaussian integral and thus Equation 5.21 becomes

$$C_Y(Y) = \pi^{3/2} b\, C_{max}(Y) = C_{max}(Y) R_{0.5}(Y) \frac{\pi^{3/2}}{\sqrt{0.69315}} \tag{5.22}$$

This expresses the total concentration at a given depth in terms of the maximum concentration and radial coordinate corresponding to $0.5C_{max}(Y)$ at that depth. The next task involves functionally described either $C_{max}(Y)$ or $R_{0.5}(Y)$.

Figure 5-30(a) shows the maximum concentration as a function of depth for each jet height. In Figure 5-30(b), the depth is scaled as before and the scaled maximum concentration is defined as

$$C^* = \frac{C_{max}(Y) - \min(C_{max}(Y))}{\max(C_{max}(Y)) - \min(C_{max}(Y))} \tag{5.23}$$

The scaled data show a consistent trend of relatively constant maximum concentration, followed by linear increase to depth Y_m, then linear decrease to the penetration depth. After the penetration depth, the maximum concentration appears to be relatively constant (or at least slowly decreasing), but more data is needed to confirm this. Note that for jet height $h = 7$ mm, no data exists beyond Y_m, which alters the scaling as the minimum concentration usually occurs beyond this depth. Considering only jet height $h = 21$ mm and $h = 43$ mm, it appears as though the data again collapse to a single curve as shown in Figure 5-30(c). Linear fits are shown for the increasing and decreasing regions of C^*. The end of the sharply decreasing

region coincides with the penetration depth, but the physical reason for location of the start the sharply increasing is not yet clear. Each linear fit is forced to have an intercept of 1 and the slopes of the increasing and decreasing regions are 2.21 and -2.87, respectively. Chanson [6] proposed that $C_{max}(Y)$ could be described by a line with negative slope, but the data presented herein show a region of linear increase, and the decreasing region is clearly not defined by a single line.

The above analysis indicates that the entire concentration distribution for any jet height can be determined given five point measurements: $\min(C_{max})$, $\max(C_{max})$, Y_m, $R_{0.5}$ at Y_m and Y_p. The value of C_{max} and corresponding depth, Y_m could be measured using a conductivity probe, for example, and similar for $\min(C_{max})$. With C_{max} determined, the radial location where $C = 0.5C_{max}$ can be found, and corresponds to $R_{0.5}$. Finally, Y_p could be determined by finding the depth location corresponding to the sharp drop in concentration. Inserting $R_{0.5}(Y_m)$ and $C_{max}(Y_m)$ into Equation 5.22 yields $C_Y(Y_m)$, which is by definition C_{Y_m}. Therefore, the total concentration, C_Y is fully described using Equation 5.18, $C_{max}(Y)$ is found from the curve in Figure 5-30(c) and Equation 5.22 sets $R_{0.5}(Y)$; Equation 5.19 can then be used to find the radial concentration profile at any depth.

5.6 Conclusions

The pervasiveness of bubbly flows in engineering applications demands continued advancement in measurement methods to fully understand physical processes, particularly in dense bubble flows. A method for reconstructing a dense three-dimensional bubble field was presented in this chapter. The bubble field was induced by a turbulent plunging jet impinging on a free-surface. Synthetic aperture refocusing of images captured by multiple cameras allowed for generation of a focal stack. Bub-

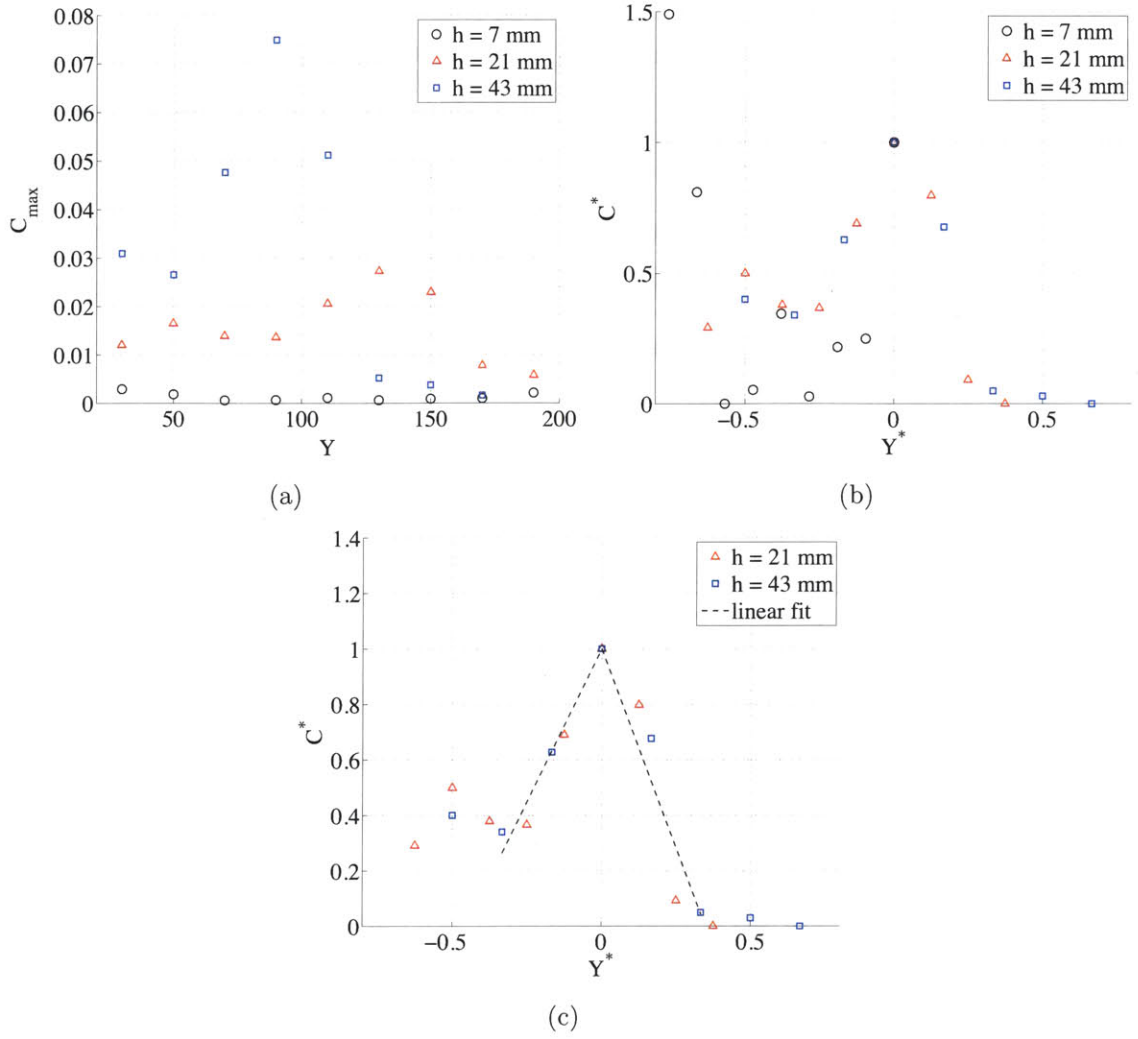


Figure 5-30: Maximum concentration as a function of depth for each jet height (a), scaled maximum concentration as a function of scaled depth(b) and linear fits to the increasing and decreasing regions of maximum concentration.

bles appear in sharp focus on the slice of the stack corresponding to the true depth of the bubble, and appear blurred on other slices. Two extraction algorithms enabled measurement of bubble size and location throughout the volume. The resulting size distributions and instantaneous location gives insight into the spatial distribution of various size bubbles in the flow. The data were shown to be in excellent agreement with existing work, and a new scaling law for the total concentration as a function of depth was proposed. Ultimately, time-resolved bubble tracks can be generated from the data sets. The technique has ramifications for a variety of multiphase flow problems which demand instrumentation capable of measuring in dense fields.

Acknowledgements

This work was funded by the Office of Naval Research, University Laboratory Initiative (ULI) program (Grant no. N00014-10-1-0342, monitored by Dr. Teresa McMullen) and carried out at the Naval Undersea Warfare Center, Newport, RI; we are thankful for the support of both. Also, we gratefully acknowledge Jason O'Connell of Tech Imaging and Karl Wiencek of Photron for providing the cameras, helping with the experiment and providing regular sanity checks. Finally, thanks to Michael Axiak for helping with the experiments.

Bibliography

[1] Hubert Chanson and Richard Manasseh. Air entrainment processes in a circular plunging jet: void-fraction and acoustic measurements. *Journal of Fluids Engineering*, 125(5):910–921, September 2003.

[2] D Bröder and M Sommerfeld. Planar shadow image velocimetry for the analysis of the hydrodynamics in bubbly flows. *Measurement Science and Technology*, 18(8):2513, July 2007.

[3] F. Pereira, M. Gharib, D. Dabiri, and D. Modarress. Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. Application to bubbly flows. *Experiments in Fluids*, 29(7):S078–S084, 2000.

[4] Lei Tian, Nick Loomis, Josè A. DomÌnguez-Caballero, and George Barbastathis. Quantitative measurement of size and three-dimensional position of fast-moving bubbles in air-water mixture flows using digital holography. *Appl. Opt.*, 49(9):1549–1554, 03 2010.

[5] Andrzej K. Bin. Gas entrainment by plunging liquid jets. *Chemical Engineering Science*, 48(21):3585 – 3630, 1993.

[6] Hubert Chanson. *Air Bubble Entrainment in Free-Surface Turbulent Shear Flows.* Academic Press, Inc.,, San Diego, CA USA, 1996.

[7] T. J. Lin and H. G. Donnelly. Gas bubble entrainment by plunging laminar liquid jets. *AIChE*, 12(3):563–571, 1966.

[8] H. N. Oguz, A. Prosperetti, and A. M. Lezzi. Examples of air-entraining flows. *Physics of Fluids A: Fluid Dynamics*, 4(4):649–651, 1991.

[9] Jens Eggers. Air entrainment through free-surface cusps. *Phys. Rev. Lett.*, 86(19):4290–4293, May 2001.

[10] Élise Lorenceau, David Quéré, and Jens Eggers. Air entrainment by a viscous jet plunging into a bath. *Phys. Rev. Lett.*, 93(25):254501, Dec 2004.

[11] Etienne Reyssat, Elise Lorenceau, Frédéric Restagno, and David Quéré. Viscous jet drawing air into a bath. *Physics of Fluids*, 20(9):091107, 2008.

[12] L. Davoust, J. L. Achard, and M. El Hmmoumi. Air entrainment by a plunging jet: the dynamical roughness concept and its estimation by a light absorption technique. *International Journal of Multiphase Flow*, 28(9):1541–1564, August 2002.

[13] H. Chanson, S. Aoki, and A. Hoque. Physical modelling and similitude of air bubble entrainment at vertical circular plunging jets. *Chemical Engineering Science*, 59(4):747–758, 2 2004.

[14] H. Chanson, S. Aoki, and A. Hoque. Bubble entrainment and dispersion in plunging jet flows: Freshwater vs. seawater. *Journal of Coastal Research*, 22(3):664–677, June 2006.

[15] Christophe Clanet and Juan C. Lasheras. Depth of penetration of bubbles entrained by a plunging water jet. *Physics of Fluids*, 9(7), 1997.

[16] George Dan Suciu and Octavian Smigelschi. Size of the submerged biphasic region in plunging jet systems. *Chemical Engineering Science*, 31(12):1217–1220, 1976.

[17] John W. M. Bush. Lecture 5: Fluid jets. MIT Lecture Notes on Surface Tension, May 2004.

[18] Jesse Belden, Tadd T. Truscott, Michael Axiak, and Alexandra H. Techet. Three-dimensional synthetic aperture particle image velocimetry. *Measurement Science and Technology*, 21(12), December 2010.

[19] Da-chuan Cheng and Hans Burkhardt. Template-based bubble identification and tracking in image sequences. *International Journal of Thermal Sciences*, 45(3):321–330, 3 2005.

[20] Aaron Isaksen, Leonard McMillan, and Steven J. Gortler. Dynamically reparameterized light field. In *SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 297–306, New York, NY, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[21] V. Vaish, G. Garg, E. Talvala, E. Antunez, B. Wilburn, M. Horowitz, and M. Levoy. Synthetic aperture focusing using a shear-warp factorization of the viewing transform. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR05') - Workshops*, volume 3, page 129. IEEE Computer Society, June 2005.

[22] G. Elsinga, F. Scarano, B. Wieneke, and B. van Oudheusden. Tomographic particle image velocimetry. *Experiments in Fluids*, 41(6):933–947, 2006.

[23] Roderick R. LaFoy. Personal Communication, 2010.

[24] Bernhard Wieneke. Volume self-calibration for 3D particle image velocimetry. *Experiments in Fluids*, 45:549–556, 2008.

[25] Fulvio Scarano and Christian Poelma. Three-dimensional vorticity patterns of cylinder wakes. *Experiments in Fluids*, 47(1):69–83, 07 2009.

[26] C. Bongiovanni, J. P. Chevaillier, and J. Fabre. Sizing of bubbles by incoherent imaging: defocus bias. *Experiments in Fluids*, 23(3):209–216, 07 1997.

[27] Jaroslav Borovička. Circle detection using hough transforms documentation. http://linux.fjfi.cvut.cz/ pinus/bristol/imageproc/hw1/report.pdf, April 2003.

[28] Yonghong Xie and Qiang Ji. A new efficient ellipse detection method. *Pattern Recognition, International Conference on*, 2:20957, 2002.

[29] T. Maxworthy, C. Gnann, M. Kürten, and F. Durst. Experiments on the rise of air bubbles in clean viscous liquids. *Journal of Fluid Mechanics*, 321:421–441, 1996.

[30] Jan Van De Donk. *Water aeration with plunging jets.* PhD thesis, Delft University of Technology, 1981.

Chapter 6

Summary and Conclusions

This thesis presented the development and implementation of a novel three-dimensional measurement system for resolving fluid flows based on synthetic aperture (SA) imaging. Synthetic aperture imaging using camera arrays allows for reconstruction of a 3D scene post-capture, and the scene may have partial occlusions in any view [1]. By extending these attributes to experimental fluid mechanics, this thesis exploited the capabilities to develop a 3DPIV method as well as a technique for resolving optically dense bubbly flows. The key steps involved in an SA imaging fluid flow measurement project are summarized in Figure 6-1. Synchronized image capture using low and high speed camera arrays was described in Chapter 3. Chapter 4 developed the auto-calibration method and referred to calibration point correspondence algorithms. Formation of SA refocused images and subsequent feature extraction and post-processing for 3DPIV and multiphase flows was presented in Chapters 2 and 5, respectively.

The main contributions of the thesis include:

- Development of a three-dimensional particle image velocimetry method (3D

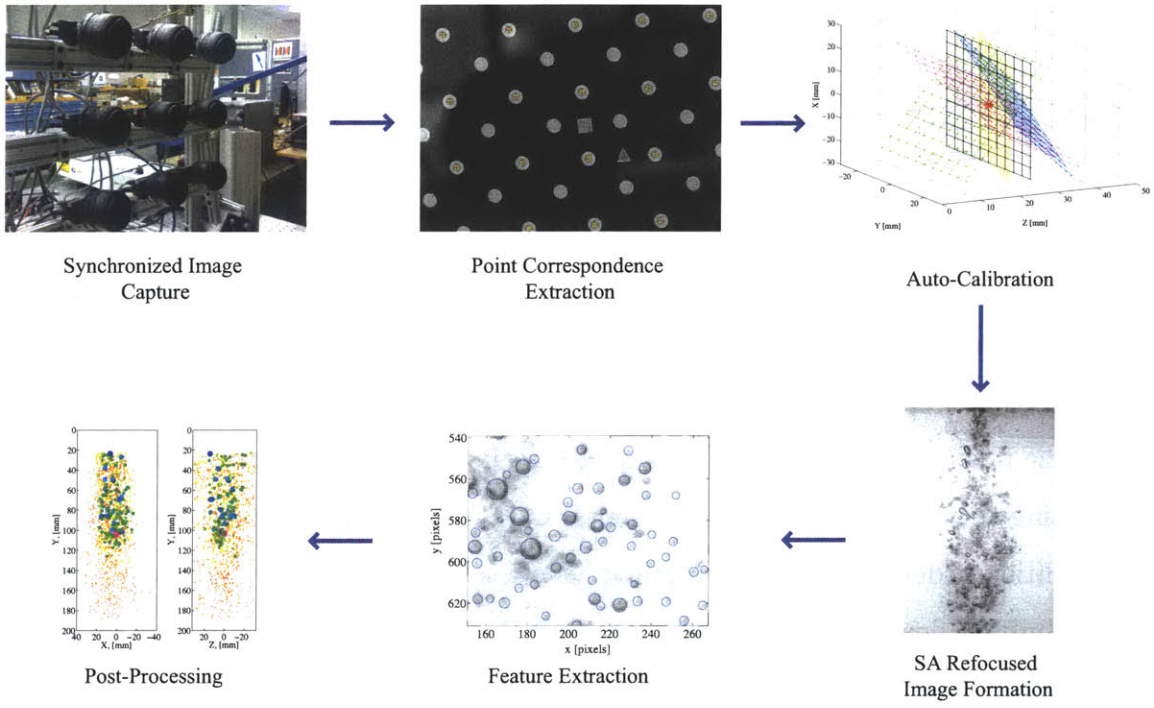


Figure 6-1: "Road map" for an SA imaging fluid flow measurement project.

SAPIV [2]) capable of resolving flows with larger seeding density than previously reported in the literature.

- Design and implementation of a low-cost camera array consisting of machine vision cameras for use in SA imaging fluid flow studies.
- Generation of a multi-camera auto-calibration technique capable of calibrating cameras when an air-glass-water transition exists between the cameras and the scene and the cameras are in general location and orientation.
- Creation of a novel bubble identification metric for SA refocused images.
- Development of a bubble property extraction algorithm based on the bubble feature metric.
- Generation of a quantitative measure for defining bubble penetration depth and bubble cloud width in plunging jet flows when only qualitative definitions existed previously.
- Proposal of a scaling law for total air concentration as a function of depth for plunging jet flows, regardless of jet height.
- Proposal of a scaling law for maximum air concentration as a function of depth for plunging jet flows, regardless of jet height.
- Enabled description of entire concentration distribution for any jet height based on scaling laws using only five point measurements.

By way of further demonstrating capabilities of the SA imaging measurement method for fluid flow, and to motivate future work, preliminary results from a liquid

jet in cross-flow experiment are presented[1]. This problem is important in combustion applications [3]. Figure 6-2 shows the experimental setup for the liquid jet in cross-flow experiment. The low-speed camera array is used to image the experiment with a white flash providing illumination. Because only a thin acrylic wall lies in the optical path (no water), the Svoboda [4, 5] auto-calibration toolbox accurately calibrates the cameras using a checkerboard grid placed at random locations and one reference location. Figure 6-3(a) shows a raw image from one camera of grid at the reference plane and Figure 6-3(b) displays a portion of an SA refocused image at depth of the reference plane. The sharpness of the SA refocused image qualitatively demonstrates the accuracy of the auto-calibration technique.

Figure 6-4 shows a raw image from one camera in the array of the liquid jet in cross-flow. The image depicts the complex structure of the jet breakup with a solid liquid core breaking up into ligaments and finally into droplets. A large amount of optical occlusion is present in the flow. Figure 6-5 shows the results of SA refocusing of the images. A range of focal depths is shown; $Z = 0$ mm coincides with the nozzle axis and Z decreases away from the cameras. The refocused images reveal some of the complex structure of the jet not seen in the single camera view, and enables placement of structures and droplets in 3D space. Initial results of feature extraction using edge detection are shown in Figure . The suitability of the technique for this difficult problem is evident, and the foundation has been laid for systematic analysis of the structure at various flow conditions in the future.

Other avenues for future work include:

- Improvement in bubble feature extraction algorithm accuracy through param-

[1]This work is in collaboration with Scott Phillips, Darin Knaus and Patrick Magari of Creare, Inc. and Tom Milnes, Barry Scharfman, Prof. Doug Hart and Prof. Alexandra Techet at MIT. The work was funded under Navy STTR N00014-10-M-0248, Technical Officer Dr. Clifford Bedford.

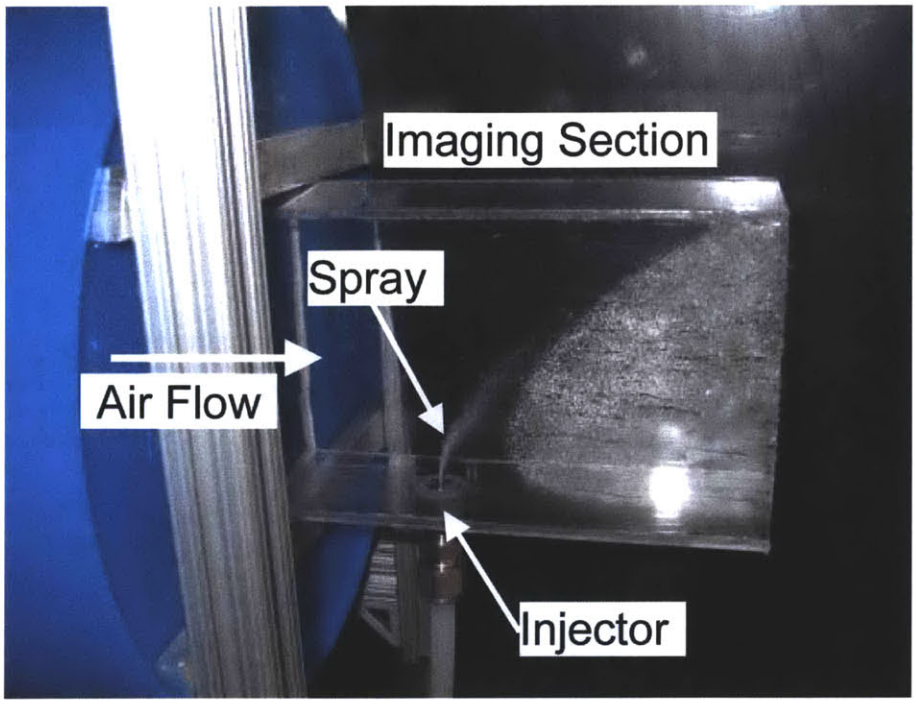
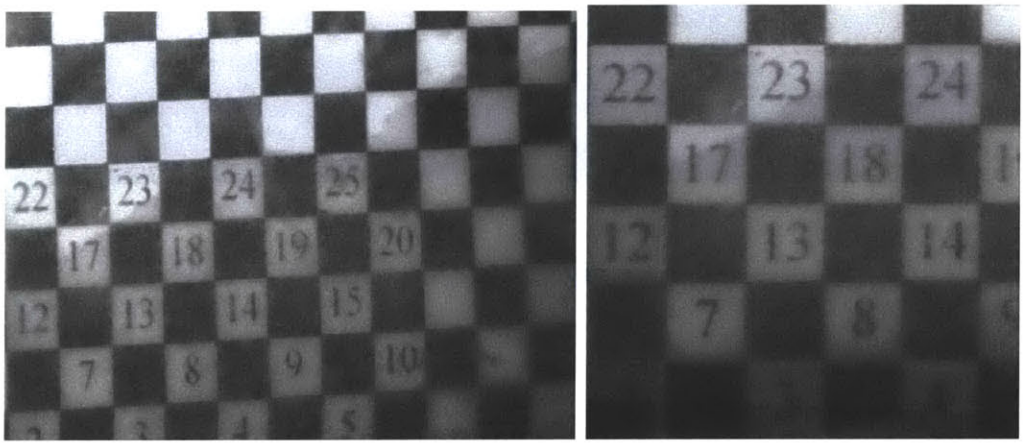


Figure 6-2: Jet in cross-flow experimental setup.



(a) (b)

Figure 6-3: Raw image from one camera of grid at the reference plane (a) and portion of an SA refocused image at depth of the reference plane (b).

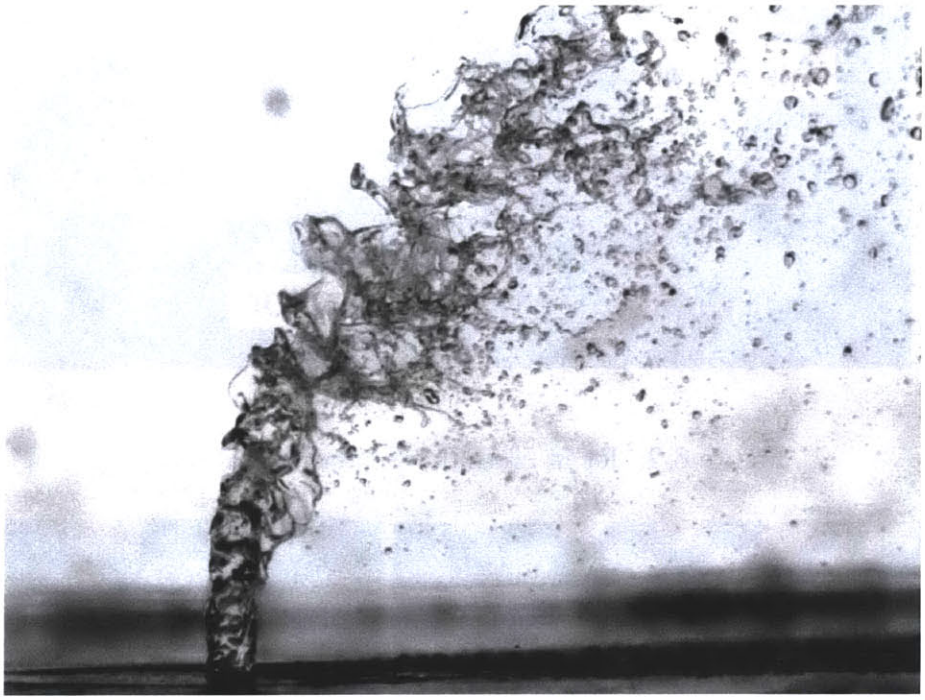
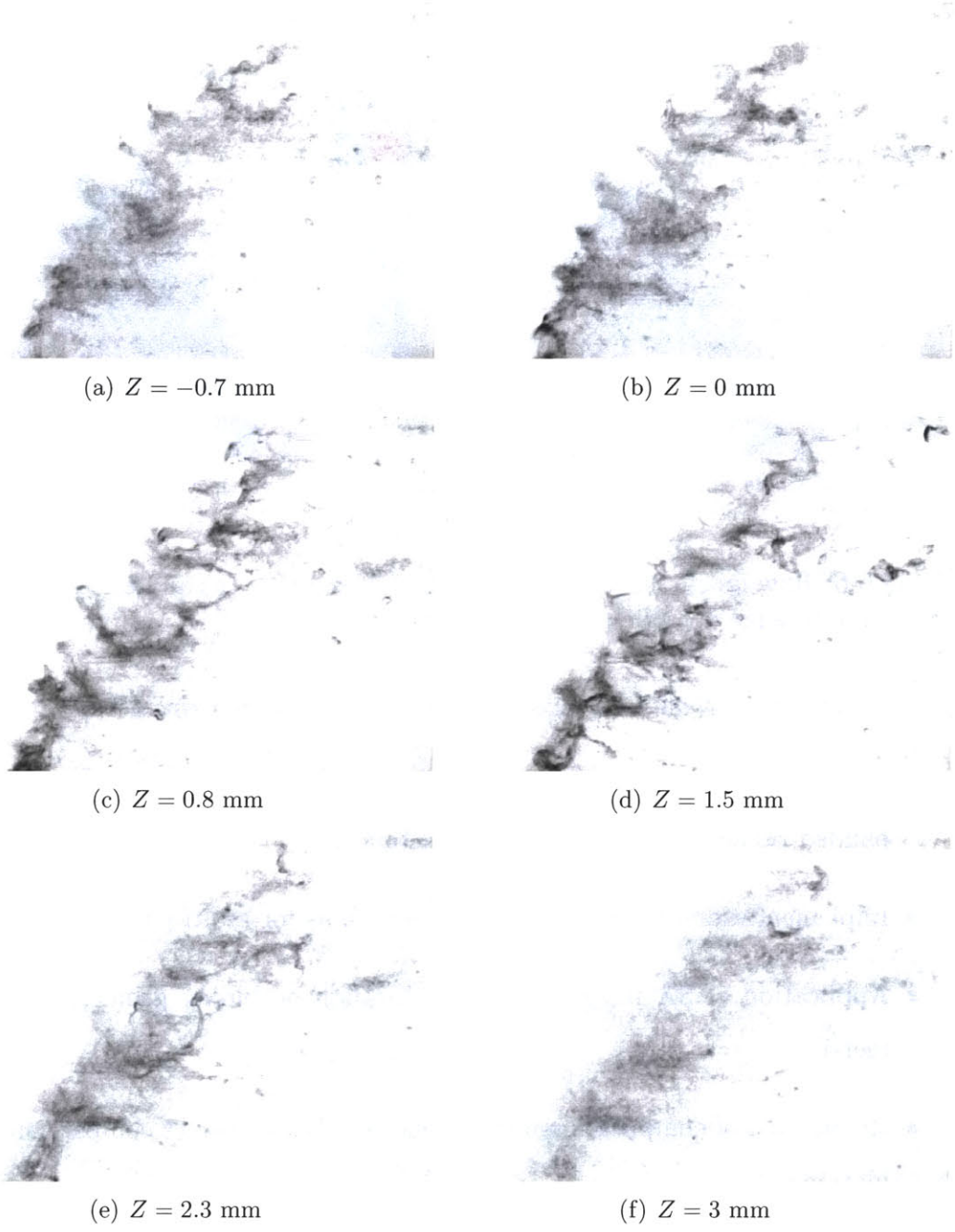


Figure 6-4: Raw image of liquid jet in cross-flow.



(a) $Z = -0.7$ mm

(b) $Z = 0$ mm

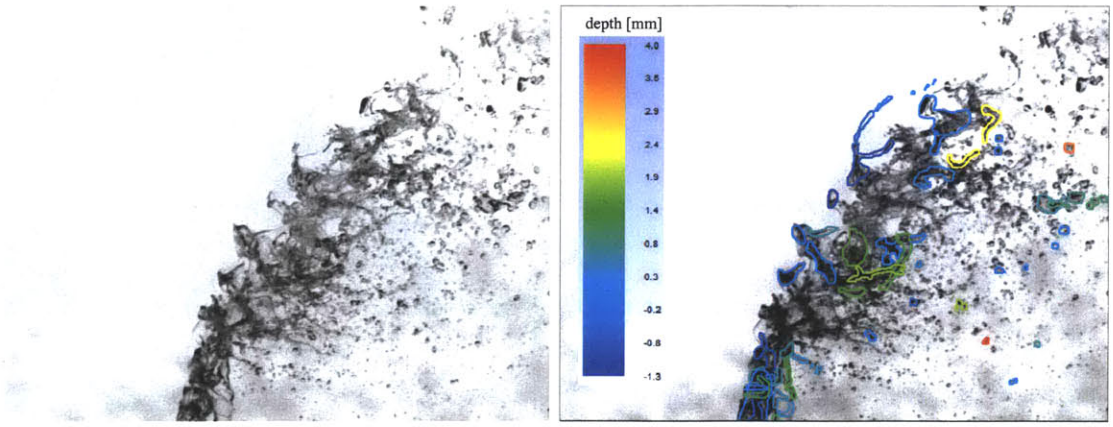
(c) $Z = 0.8$ mm

(d) $Z = 1.5$ mm

(e) $Z = 2.3$ mm

(f) $Z = 3$ mm

Figure 6-5: SA refocused images of the jet in cross-flow at various depths revealing complex structure.



(a) single camera raw image (b) feature depths

Figure 6-6: Raw image from a single camera of the array (a) and estimated depth of features from edge detection algorithm (b).

eterization of scale for gradients and iterative contour estimation.

- Further validation of the scaling laws proposed in Chapter 5 through an expanded range of experimental parameters.
- Implementation of bubble tracking algorithms for multi-phase flows.
- Application of SA imaging to flows through occluding geometry (e.g., coral reefs)
- Replication of computational turbulence studies for direct comparison and validation
- Application to three-phase bubble/liquid/solid particle flows

The synthetic aperture imaging based measurement system developed in this thesis provide a new and novel method for 3D-3C, quantitative flow velocimetry and

3D reconstruction of optically dense multiphase flows with potential to impact a wide range of applications.

Bibliography

[1] M. Levoy. Light fields and computational imaging. *Computer*, 39(8):46–55, Aug. 2006.

[2] Jesse Belden, Tadd T. Truscott, Michael Axiak, and Alexandra H. Techet. Three-dimensional synthetic aperture particle image velocimetry. *Measurement Science and Technology*, 21(12), 2010.

[3] Mark A. Linne, Megan Paciaroni, James R. Gord, and Terrence R. Meyer. Ballistic imaging of the liquid core for a steady jet in crossflow. *Appl. Opt.*, 44(31):6627–6634, 11 2005.

[4] Tomáš Svoboda, Daniel Martinec, and Tomáš Pajdla. A convenient multi-camera self-calibration for virtual environments. *PRESENCE: Teleoperators and Virtual Environments*, 14(4):407–422, August 2005.

[5] Tomáš Svoboda, Daniel Martinec, Tomáš Pajdla, Jean-Yves Bouguet, Tomas Werner, and Ondrej Chum. Multi-camera self-calibration. http://cmp.felk.cvut.cz/ svoboda/SelfCal/.

Appendix A

Reprojecting Images to Focal Planes

To reproject images onto each focal plane requires a transformation function derived from the camera calibration. The reprojection operations for both the pinhole model and the refractive model described in Chapter 4 are presented. Note that in deriving the number of operations in Chapter 3, the pinhole model was considered.

The projection of world points to image points in a pinhole camera camera is given by

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{A.1}$$

where $\mathbf{u}$ and $\mathbf{X}$ are written in homogeneous coordinates and P is the camera pinhole matrix. To reproject an images onto the kth focal plane at location Z, a 3×3

matrix is first defined by substitution of Z into Equation A.1

$$H = \begin{bmatrix} P_{11} & P_{12} & P_{13}Z + P_{14} \\ P_{21} & P_{22} & P_{23}Z + P_{24} \\ P_{31} & P_{32} & P_{33}Z + P_{34} \end{bmatrix} \tag{A.2}$$

The inverse of H takes points from image planes to world planes, but the scale and offset of the points in the world planes needs to be defined. Scale and shift can be incorporated into another 3×3 matrix

$$D = \begin{bmatrix} a & 0 & s_X \\ 0 & b & s_Y \\ 0 & 0 & 1 \end{bmatrix} \tag{A.3}$$

where a and b are user-supplied scale factors in X and Y and s_X and s_Y are user-supplied offsets in X and Y, respectively. Finally, the 3×3 transformation required to take image points to world points at the plane located at Z is

$$T = DH^{-1} \tag{A.4}$$

To reproject images to focal planes using the refractive camera model, the points are first projected to the front surface of the wall using Equations A.1 - A.4, where Z is replaced with Z_w. The angle of each back-projected ray incident upon the wall is calculated from the know camera center coordinates and the coordinates of the intersection point of the ray with the wall. Ray tracing through the remaining media (wall and water) and scaling and shifting yields the world points in the desired coordinate system.

Appendix B

Terms in Newton-Raphson Solver

The partial derivatives in equations 4.22-4.23 are given by

$$\frac{\partial f}{\partial r_A} = \frac{1}{\sqrt{{r_A}^2 + {d_A}^2}} - \frac{{r_A}^2}{\left({r_A}^2 + {d_A}^2\right)^{3/2}} + \frac{n_2/n_1}{\sqrt{\left(r_A - r_B\right)^2 + {d_B}^2}} - \frac{n_2/n_1 \left(r_A - r_B\right)\left(2r_A - 2r_B\right)}{2\left(\left(r_A - r_B\right)^2 + {d_B}^2\right)^{3/2}} \tag{B.1}$$

$$\frac{\partial f}{\partial r_B} = \frac{n_2/n_1 \left(r_A - r_B\right)\left(2r_A - 2r_B\right)}{2\left(\left(r_A - r_B\right)^2 + {d_B}^2\right)^{3/2}} - \frac{n_2/n_1}{\sqrt{\left(r_A - r_B\right)^2 + {d_B}^2}} \tag{B.2}$$

$$\frac{\partial g}{\partial r_A} = \frac{\left(r_A - r_B\right)\left(2r_A - 2r_B\right)}{2\left(\left(r_A - r_B\right)^2 + {d_B}^2\right)^{3/2}} - \frac{1}{\sqrt{\left(r_A - r_B\right)^2 + {d_B}^2}} \tag{B.3}$$

$$\frac{\partial g}{\partial r_B} = \frac{1}{\sqrt{(r_A - r_B)^2 + {d_B}^2}} + \frac{n_3/n_2}{\sqrt{(r_B - r_P)^2 + {d_P}^2}}$$
$$- \frac{(r_A - r_B)(2r_A - 2r_B)}{2\left((r_A - r_B)^2 + {d_B}^2\right)^{3/2}} - \frac{n_3/n_2 (r_B - r_P)(2r_B - 2r_P)}{2\left((r_B - r_P)^2 + {d_P}^2\right)^{3/2}} \quad \text{(B.4)}$$

Appendix C

Additional Calibration Figures

Additional plots from Chapter 4 are included in this appendix.

C.1 Convergence Plots

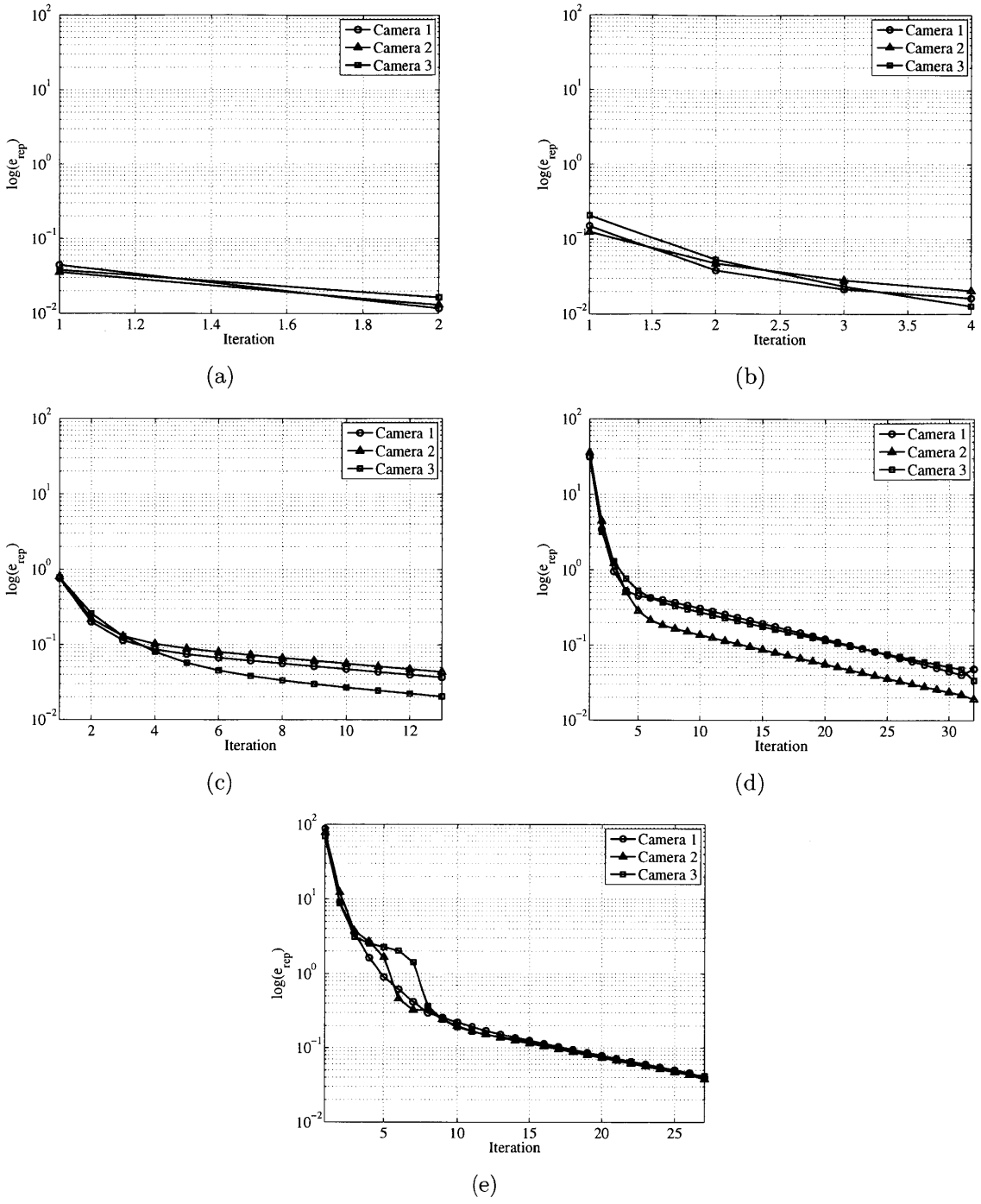


Figure C-1: Convergence plots of the auto-calibration algorithm applied to 3 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0 pixels. The initial error in world point coordinates increases from plot (a) to (e).

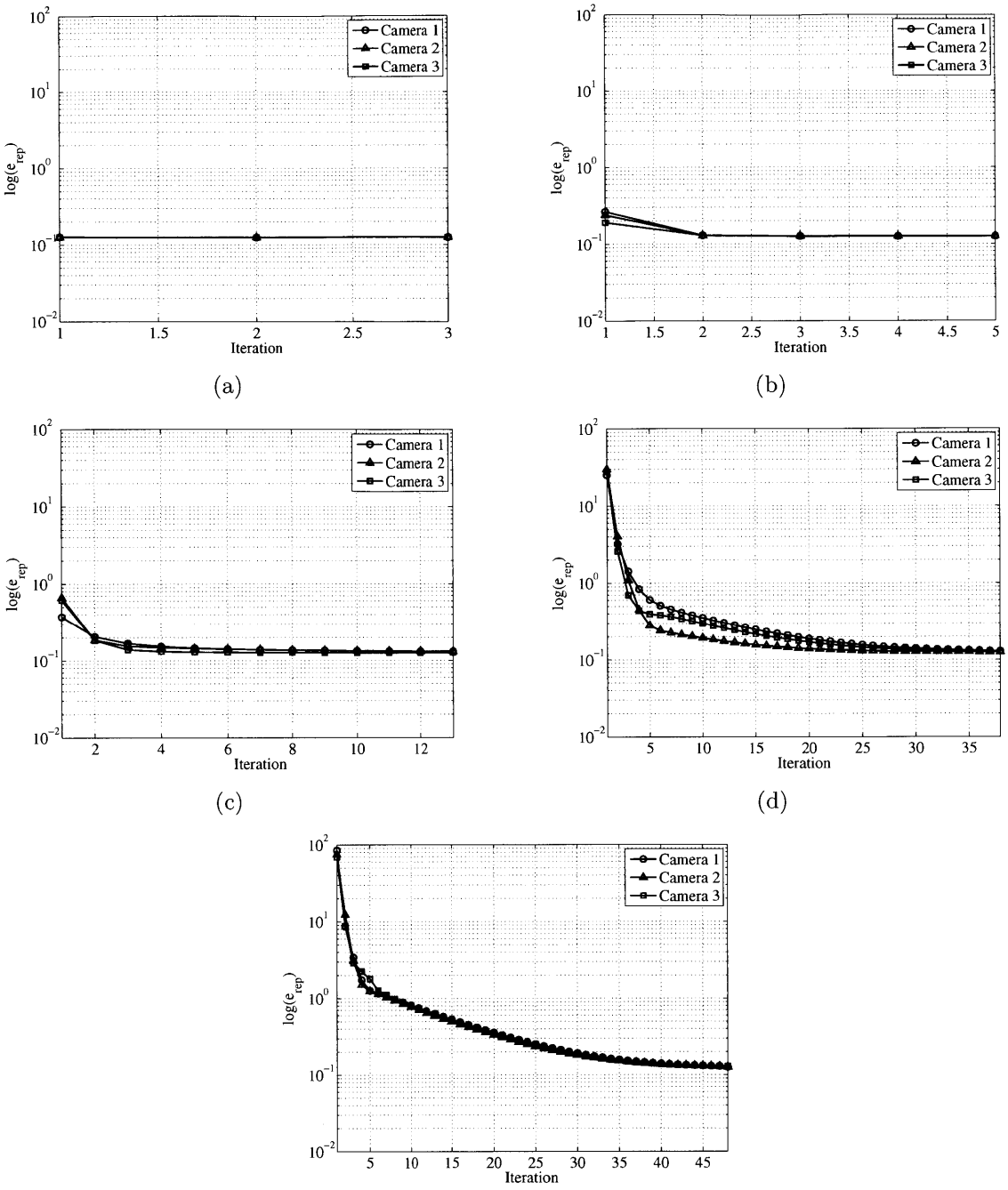


Figure C-2: Convergence plots of the auto-calibration algorithm applied to 3 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

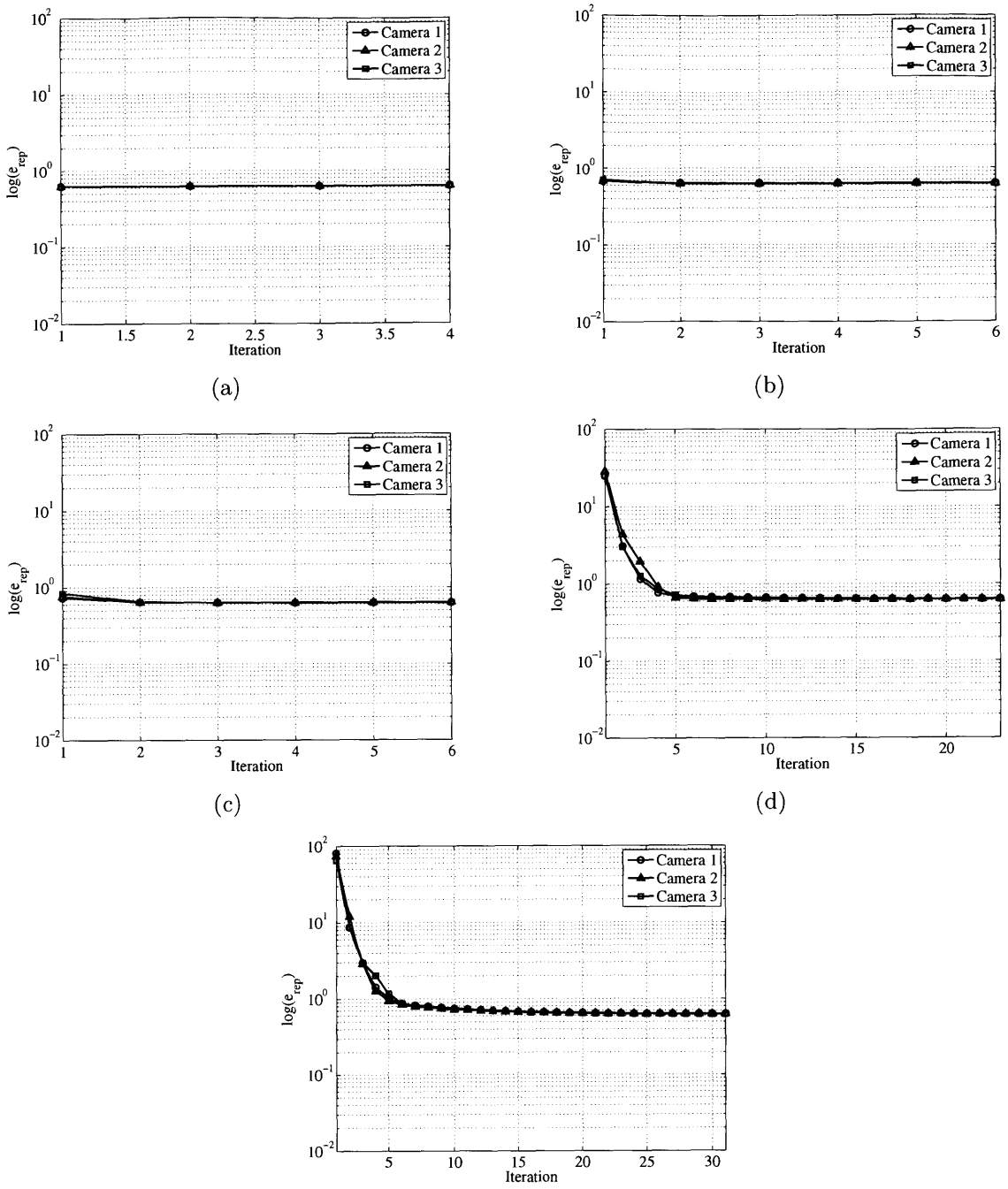


Figure C-3: Convergence plots of the auto-calibration algorithm applied to 3 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.5 pixels. The initial error in world point coordinates increases from plot (a) to (e).

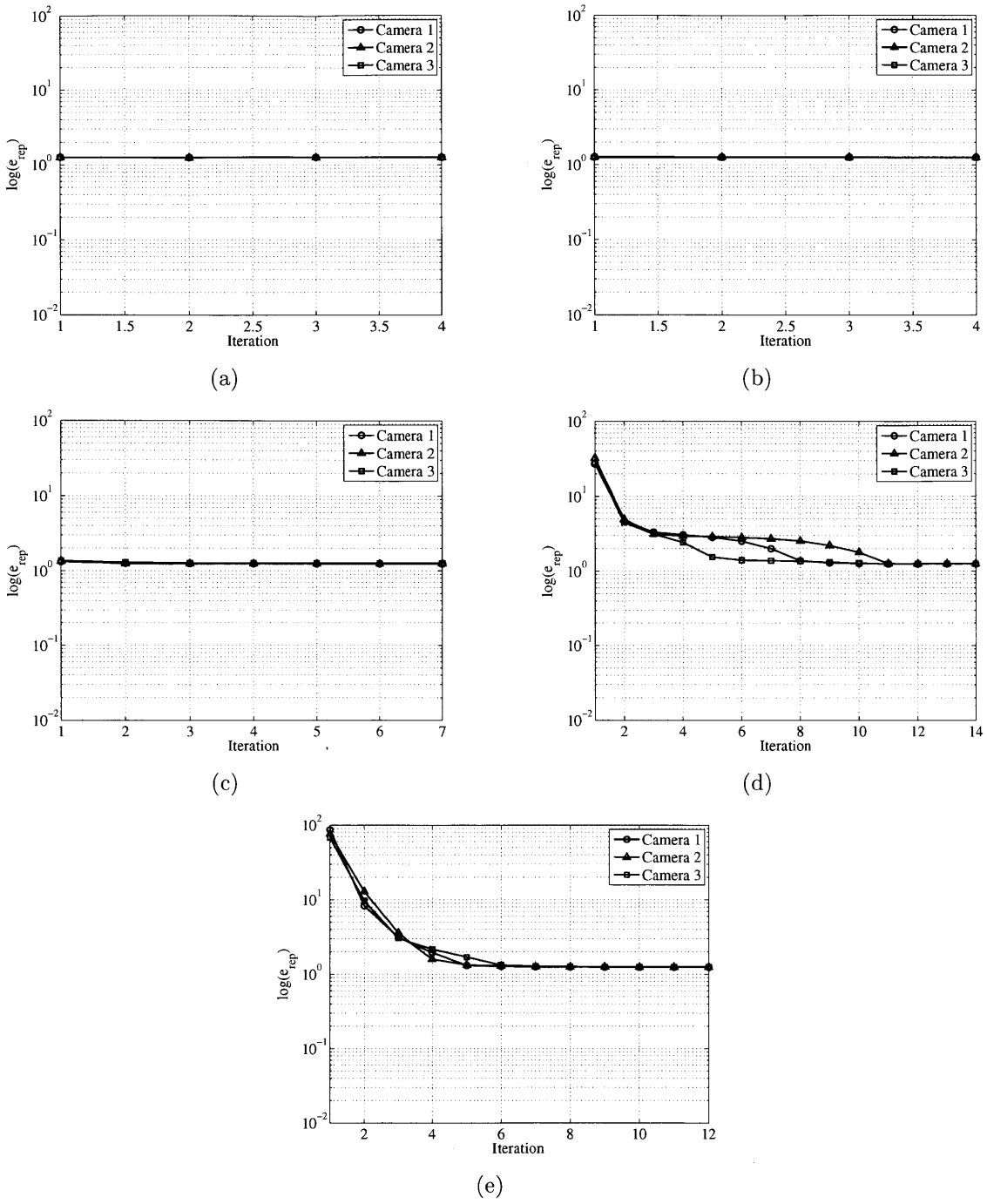


Figure C-4: Convergence plots of the auto-calibration algorithm applied to 3 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

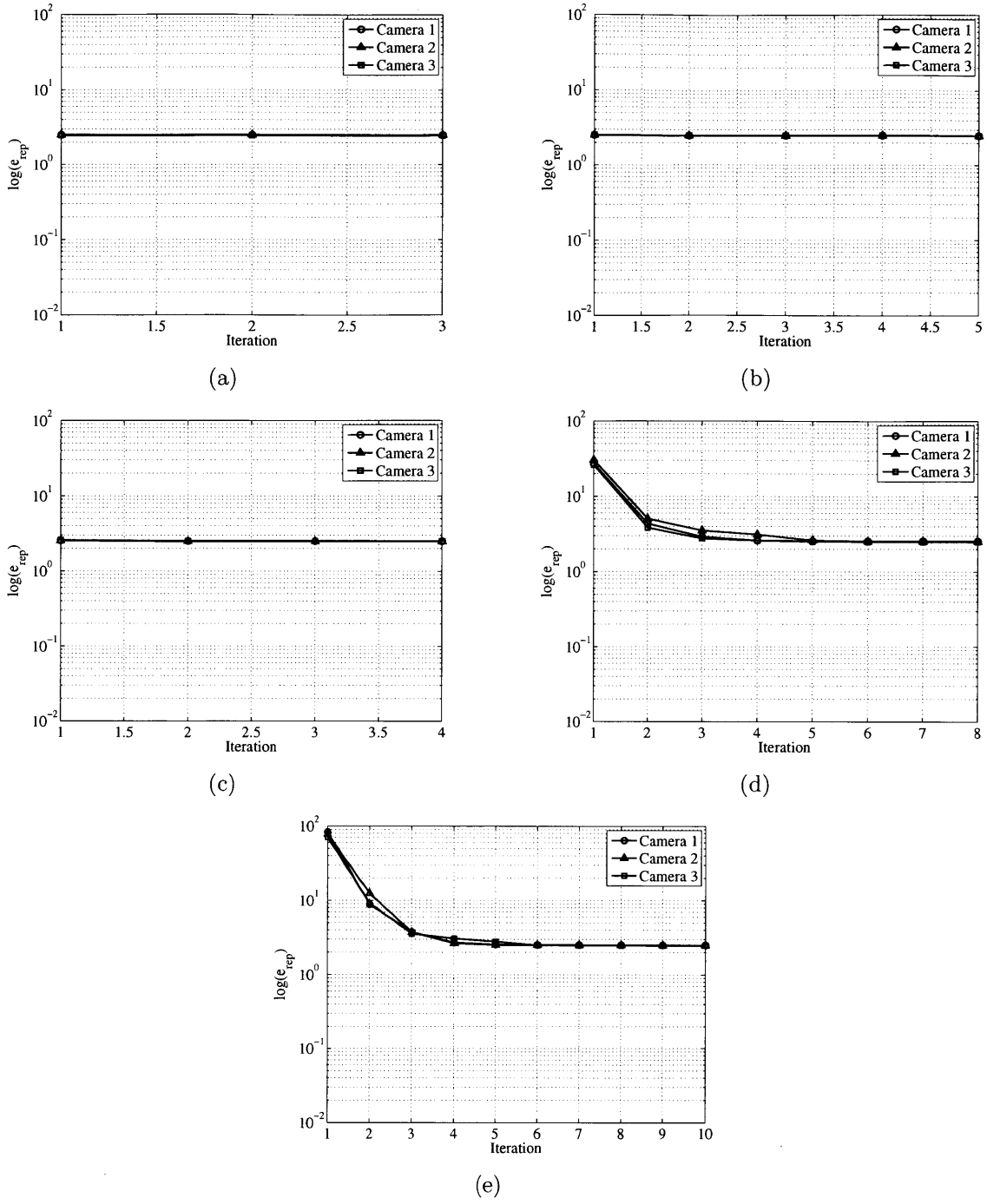


Figure C-5: Convergence plots of the auto-calibration algorithm applied to 3 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 2 pixels. The initial error in world point coordinates increases from plot (a) to (e).

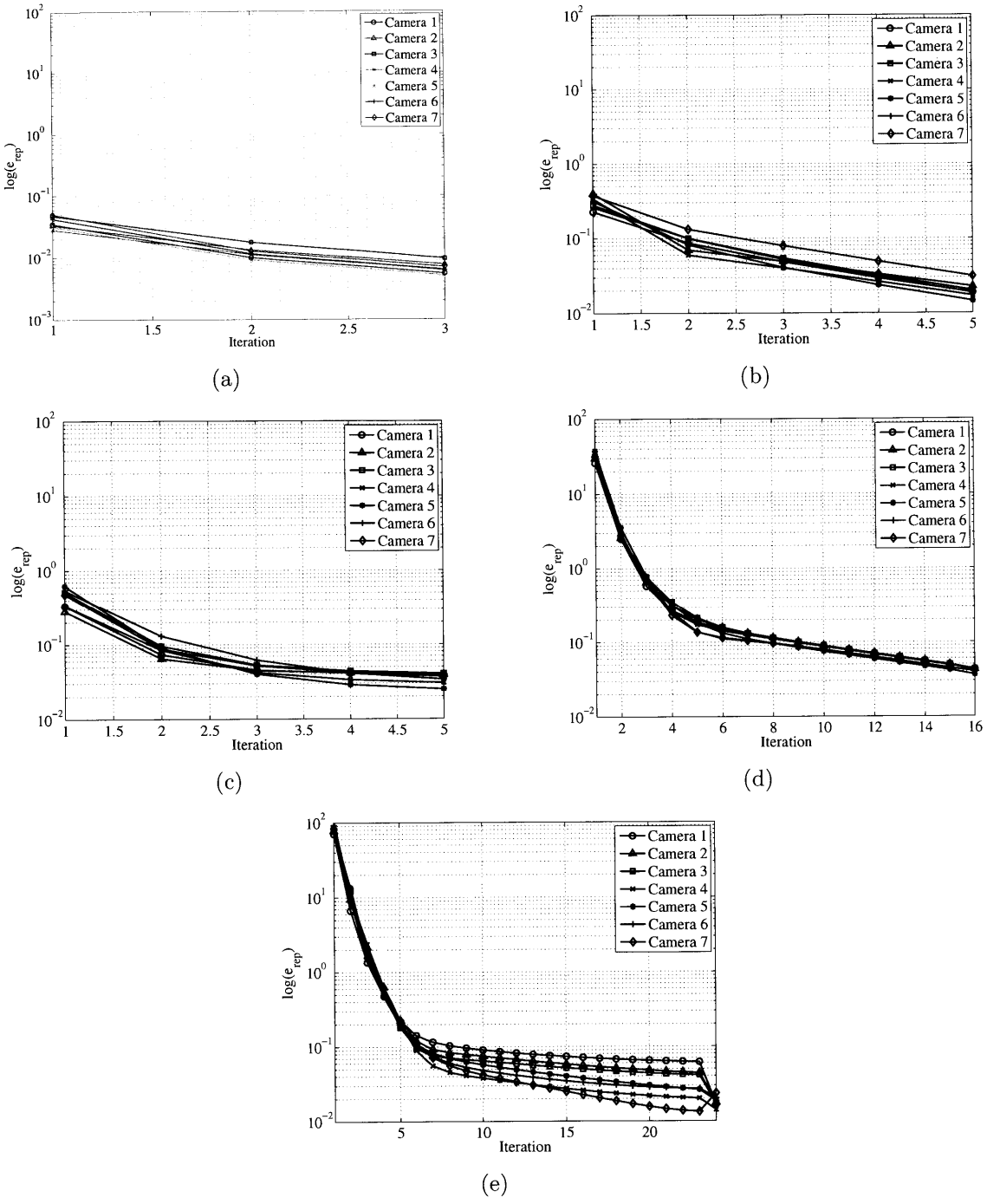


Figure C-6: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0 pixels. The initial error in world point coordinates increases from plot (a) to (e).

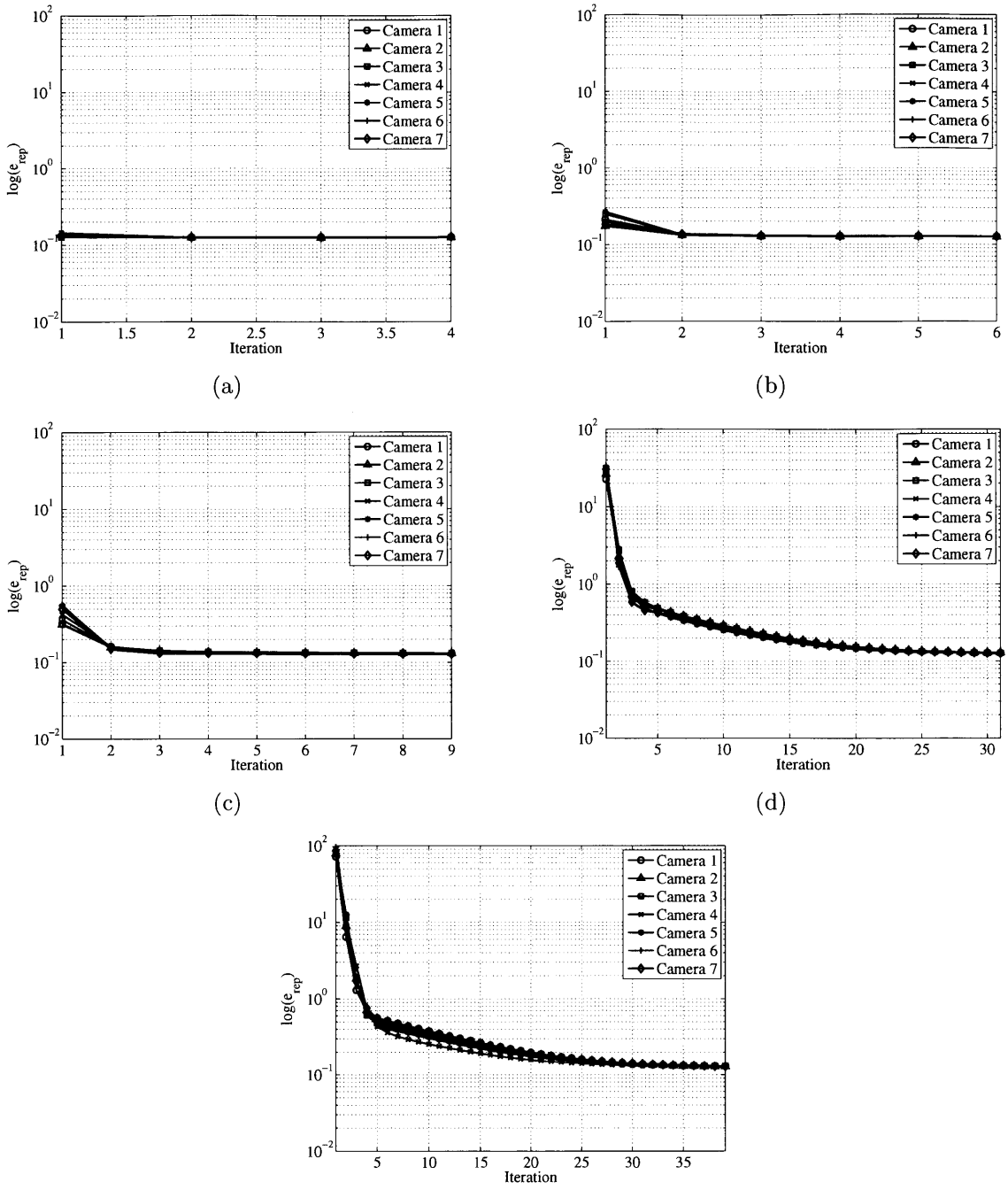


Figure C-7: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

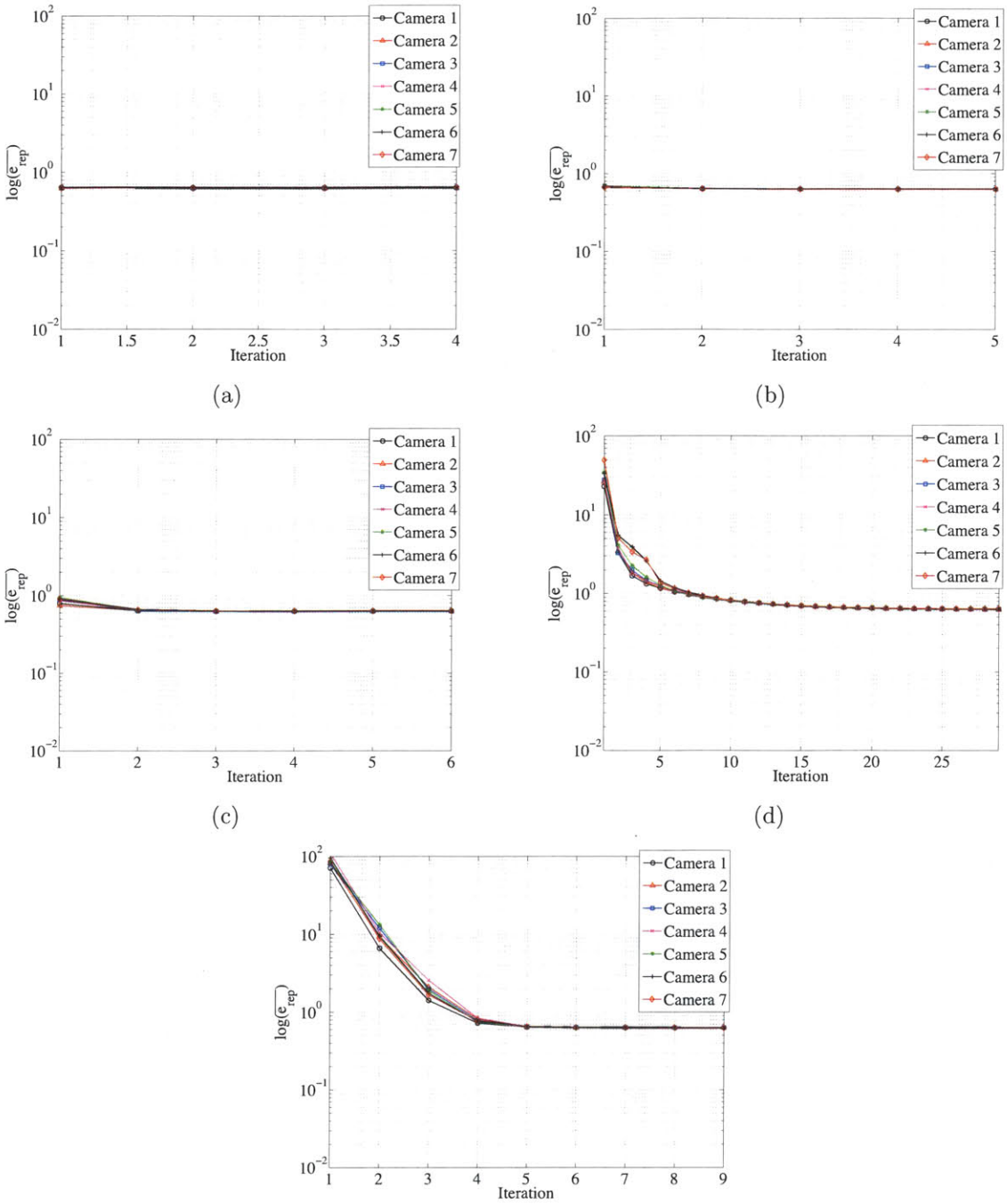


Figure C-8: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.5 pixels. The initial error in world point coordinates increases from plot (a) to (e).

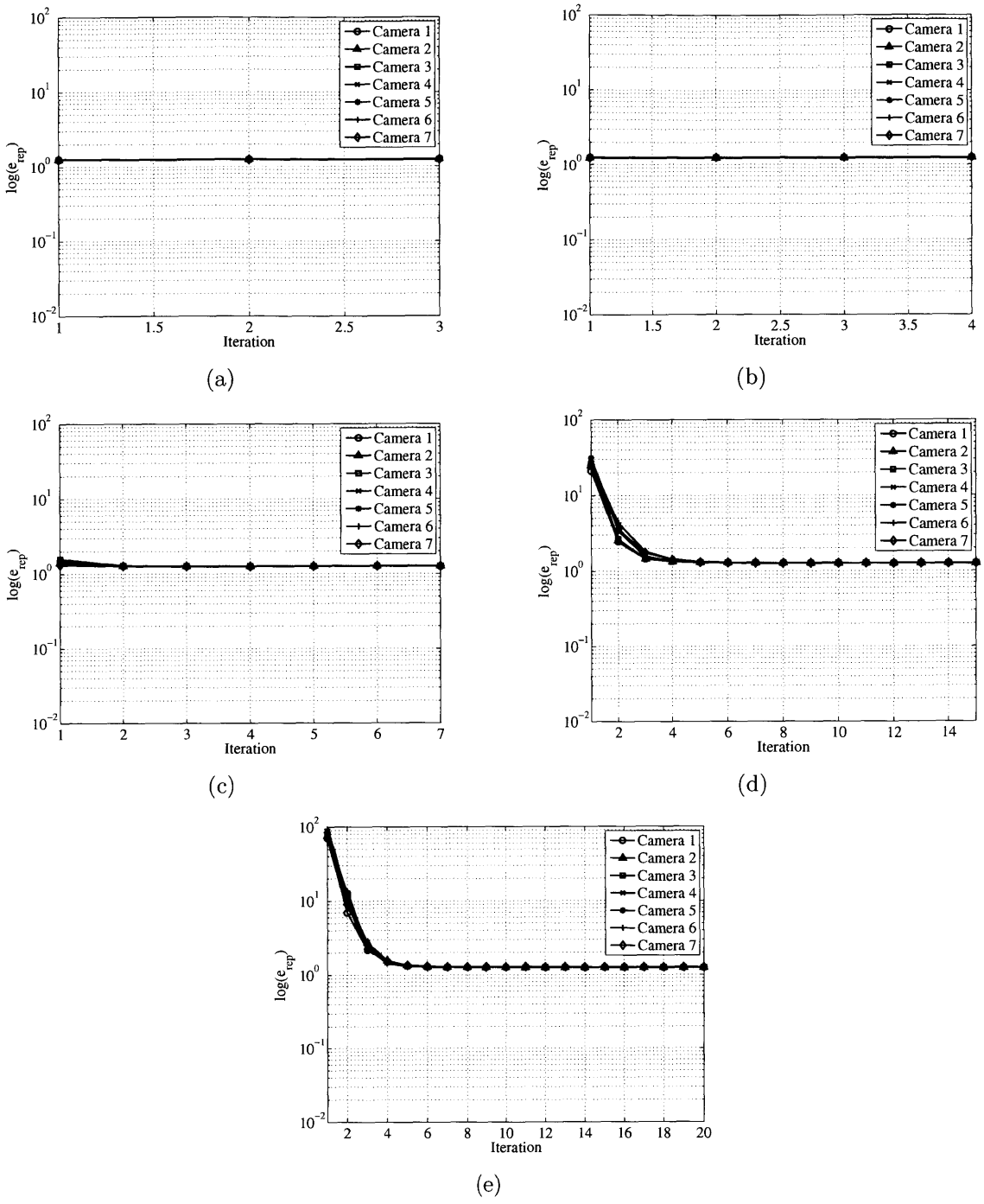


Figure C-9: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

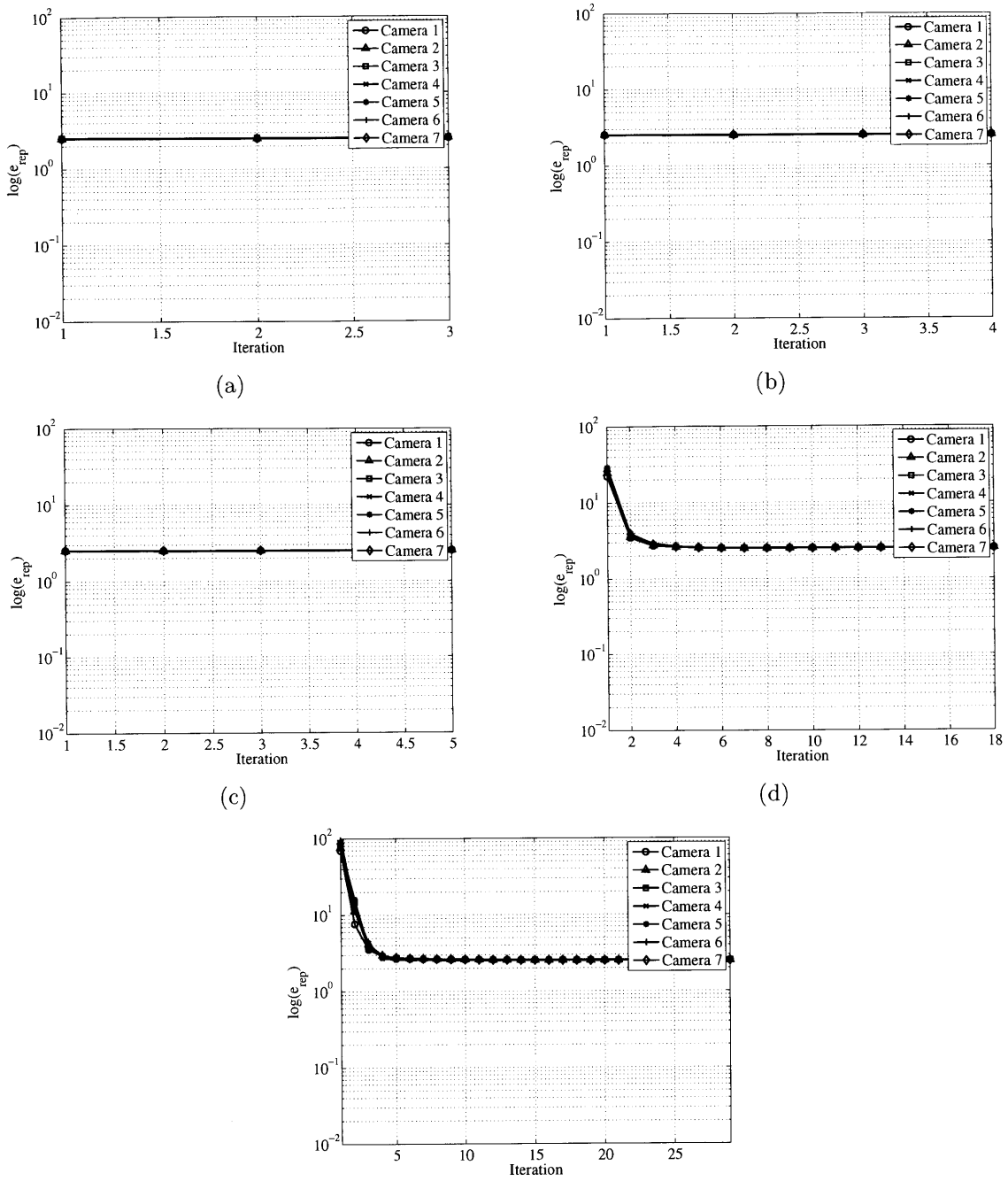


Figure C-10: Convergence plots of the auto-calibration algorithm applied to 7 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 2 pixels. The initial error in world point coordinates increases from plot (a) to (e).

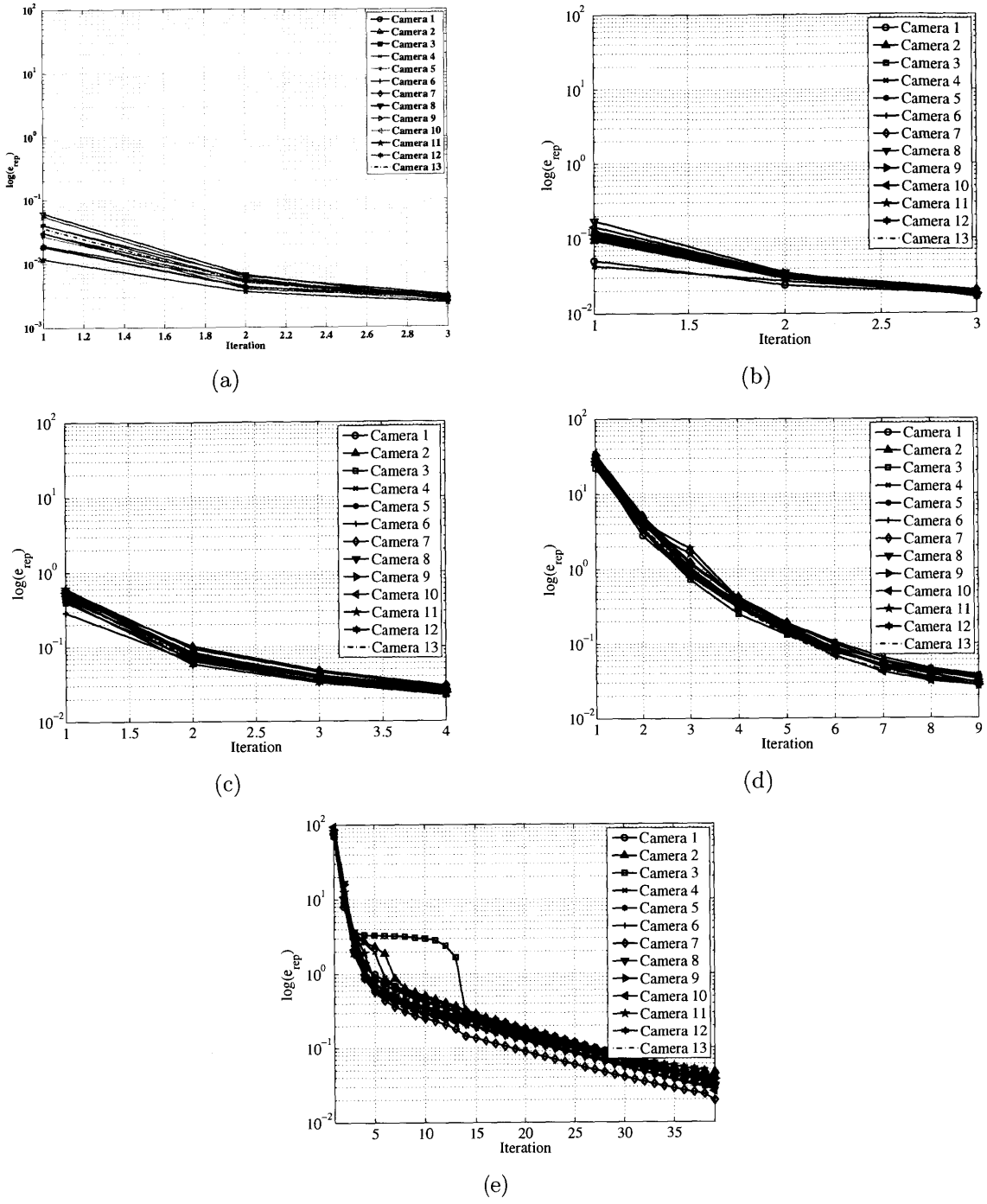


Figure C-11: Convergence plots of the auto-calibration algorithm applied to 13 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0 pixels. The initial error in world point coordinates increases from plot (a) to (e).

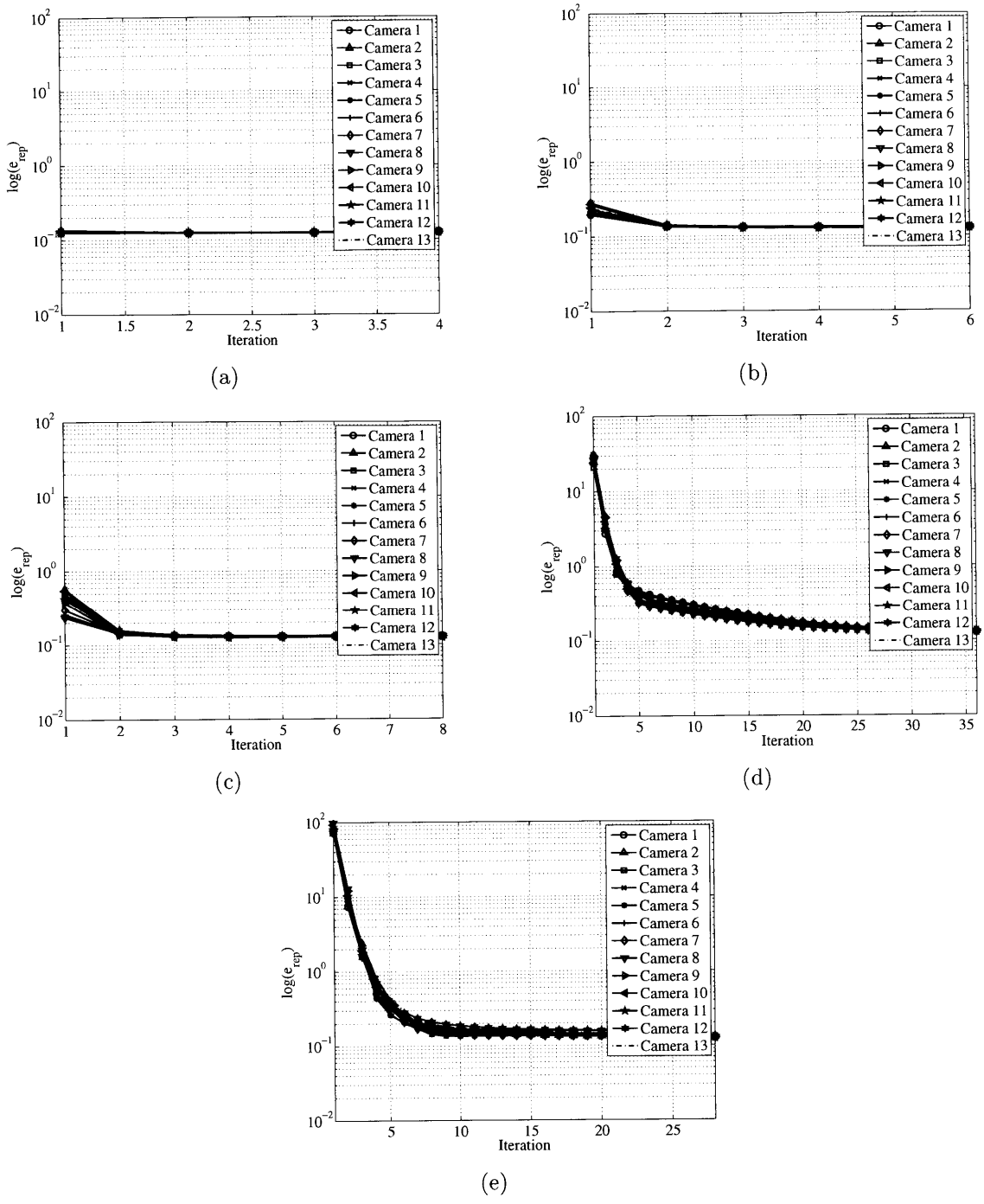


Figure C-12: Convergence plots of the auto-calibration algorithm applied to 13 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

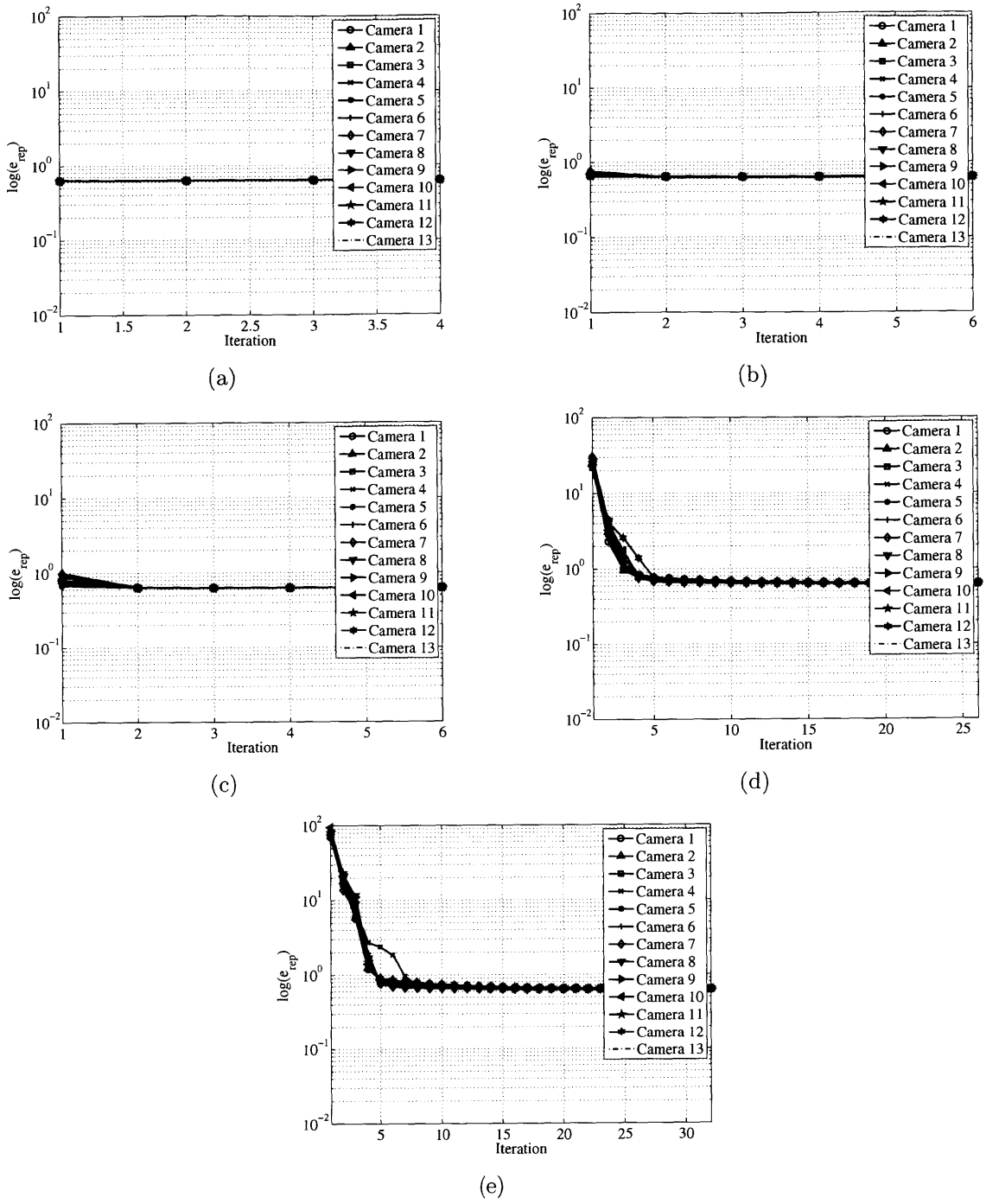


Figure C-13: Convergence plots of the auto-calibration algorithm applied to 13 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 0.5 pixels. The initial error in world point coordinates increases from plot (a) to (e).

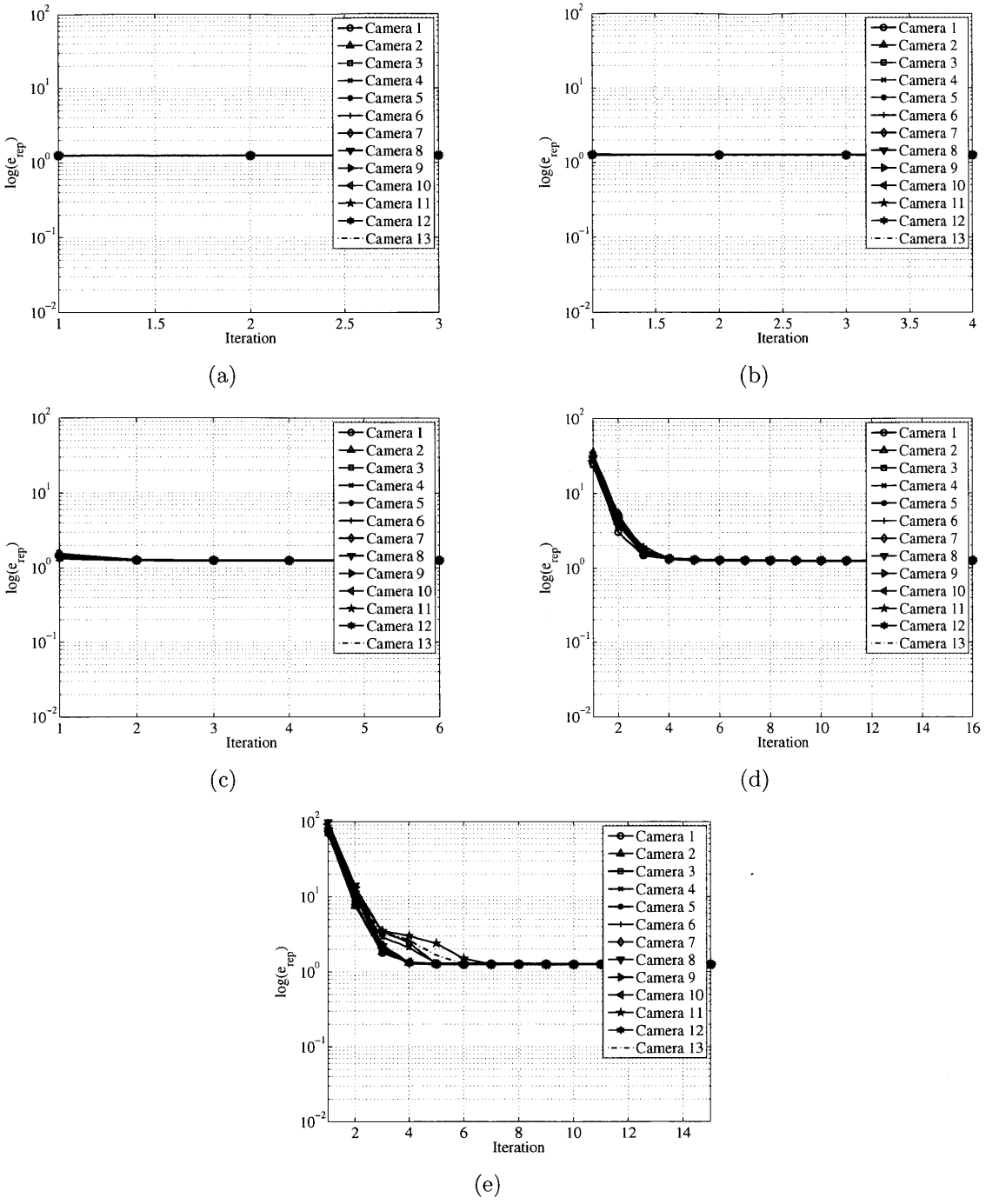


Figure C-14: Convergence plots of the auto-calibration algorithm applied to 13 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 1 pixels. The initial error in world point coordinates increases from plot (a) to (e).

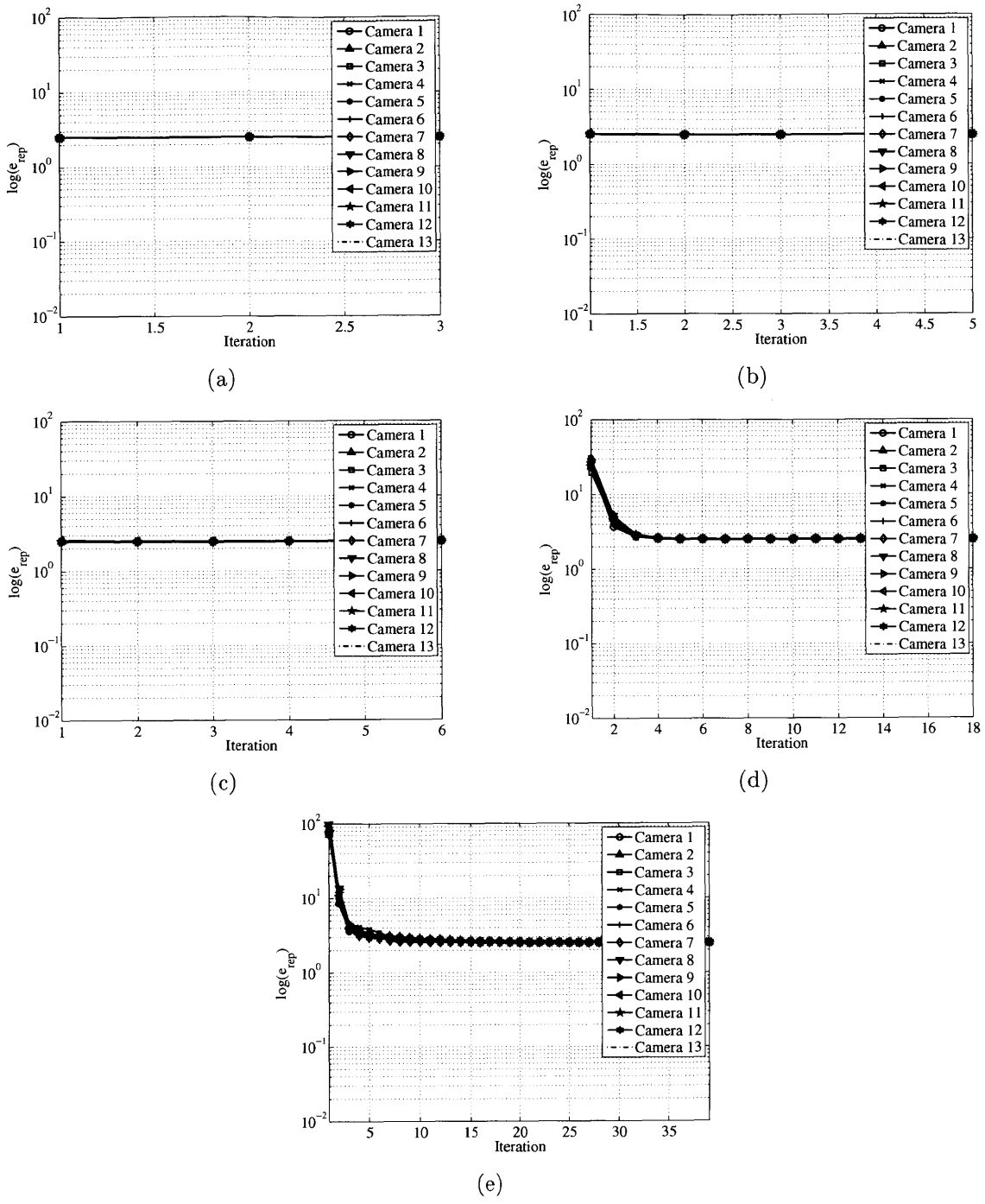


Figure C-15: Convergence plots of the auto-calibration algorithm applied to 13 simulated cameras; log of the mean reprojection error is plotted on the y-axis. For all plots, the standard deviation in the imposed image point measurements is 2 pixels. The initial error in world point coordinates increases from plot (a) to (e).

C.2 Final Error Summary Plots

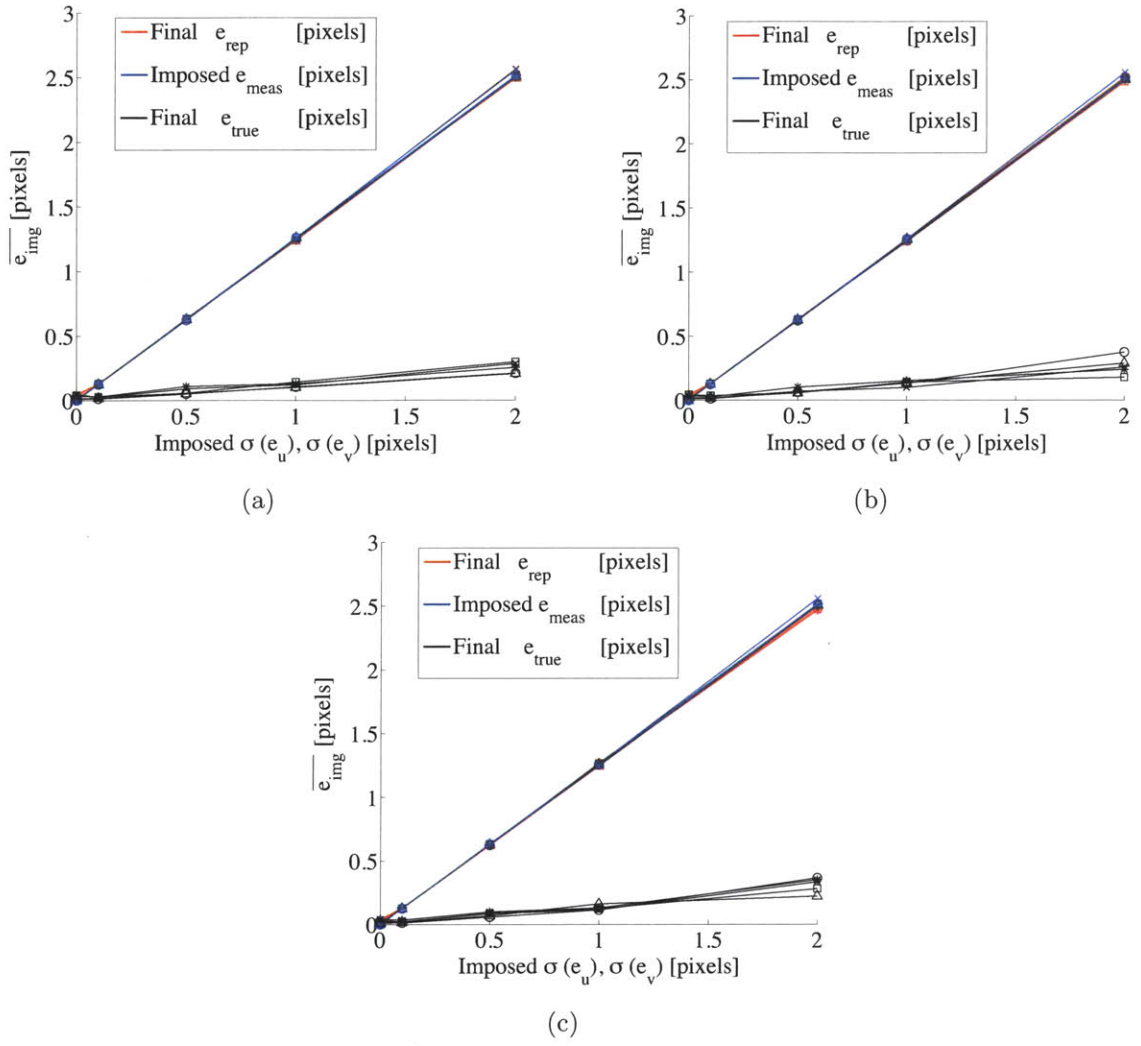


Figure C-16: Final mean reprojection error after convergence of the auto-calibration procedure applied to 3 cameras. The red and blue curves refer to the final mean reprojection error and the mean initial image measurement error, respectively; the final true mean error is shown in black. The symbols correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2.

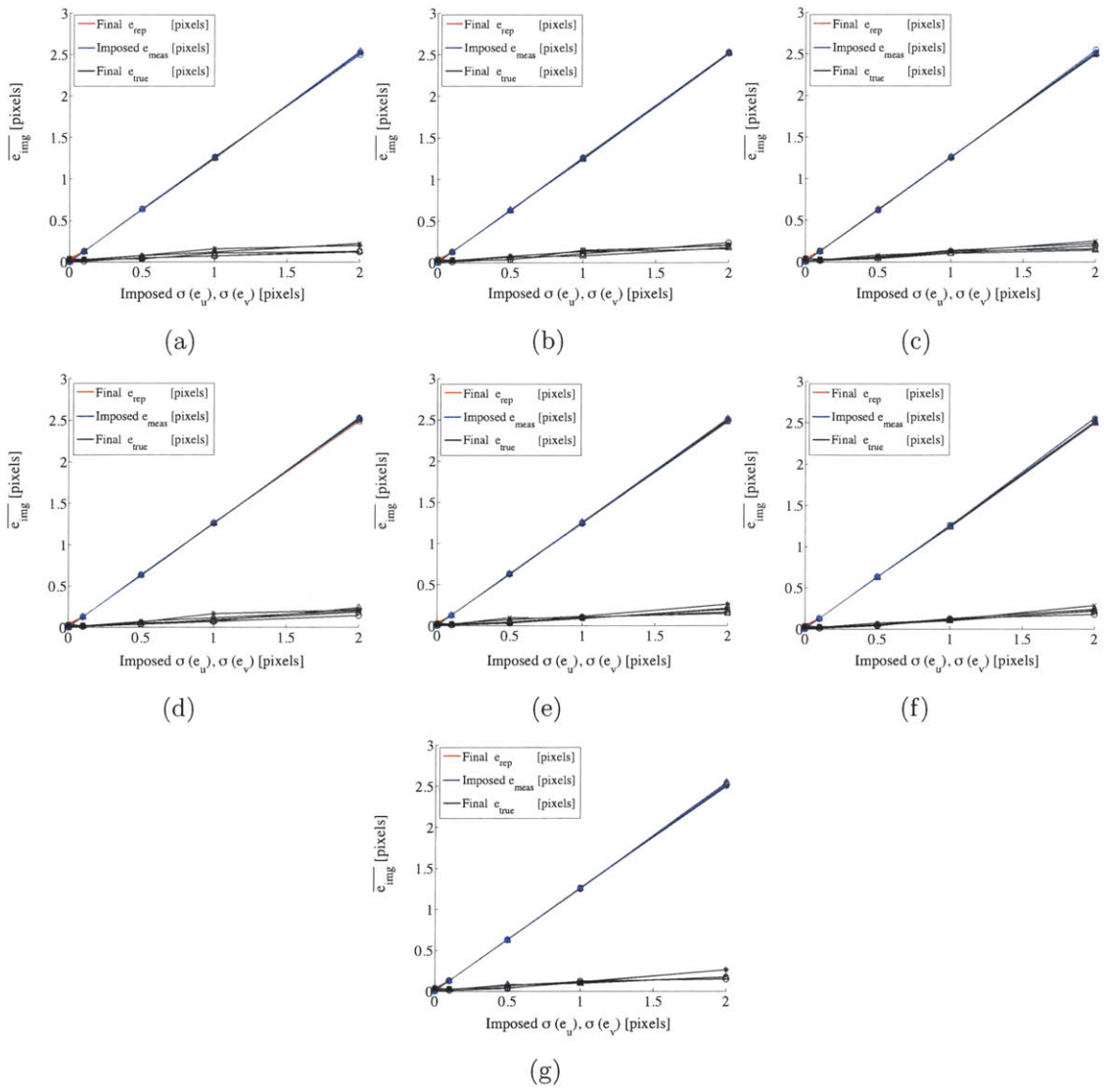


Figure C-17: Final mean reprojection error after convergence of the auto-calibration procedure applied to 7 cameras. The red and blue curves refer to the final mean reprojection error and the mean initial image measurement error, respectively; the final true mean error is shown in black. The symbols correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2.

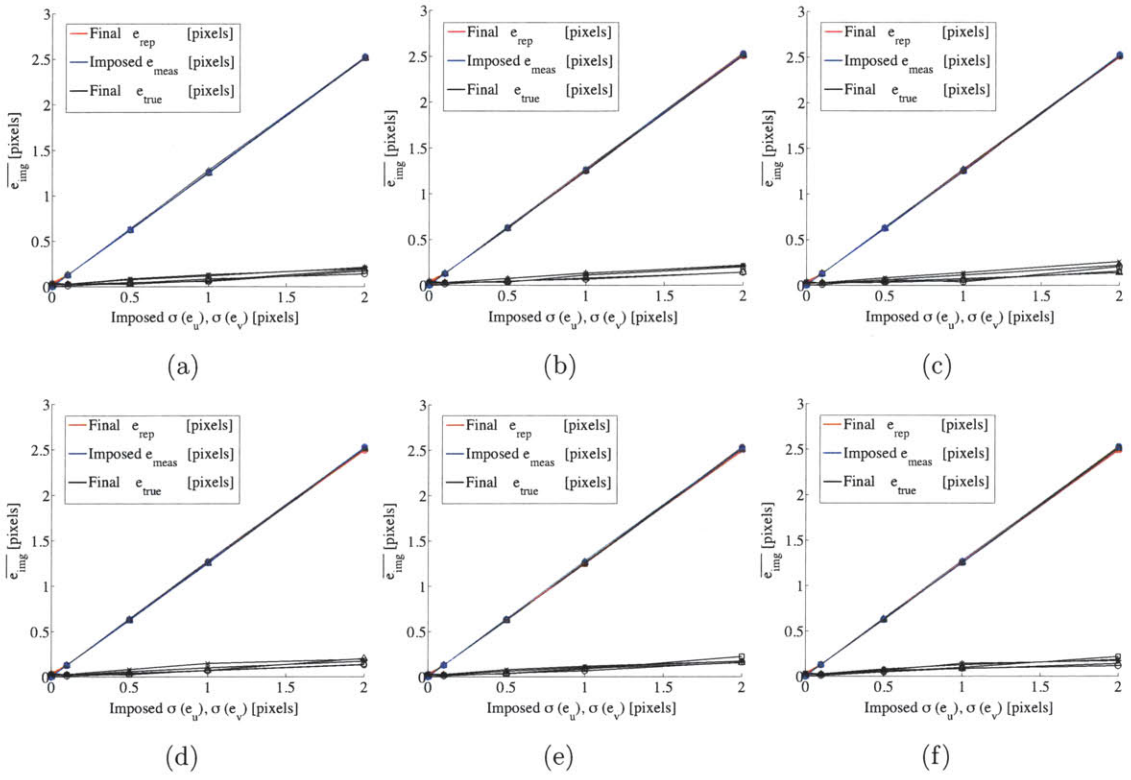


Figure C-18: Final mean reprojection error after convergence of the auto-calibration procedure applied to 13 cameras (plots for other 7 cameras shown in Figure C-19). The red and blue curves refer to the final mean reprojection error and the mean initial image measurement error, respectively; the final true mean error is shown in black. The symbols correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2.

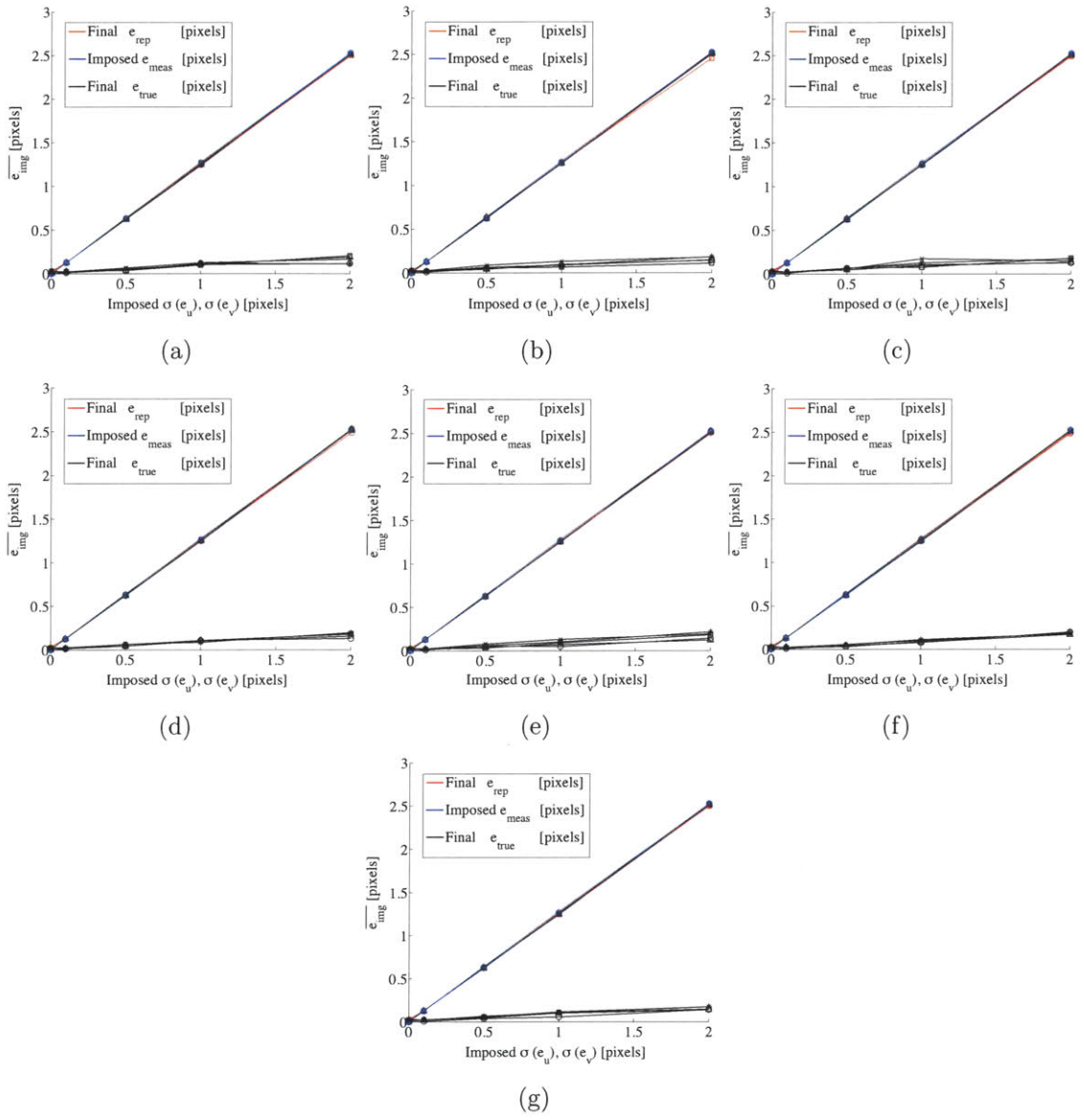


Figure C-19: Final mean reprojection error after convergence of the auto-calibration procedure applied to 13 cameras. The red and blue curves refer to the final mean reprojection error and the mean initial image measurement error, respectively; the final true mean error is shown in black. The symbols correspond to different initial values of e_W/D: o - 0.001, $\triangle$ - 0.005, $\square$ - 0.01, $\times$ - 0.1, $\star$ - 0.2..

C.3 Final World Point Error Summary Plots

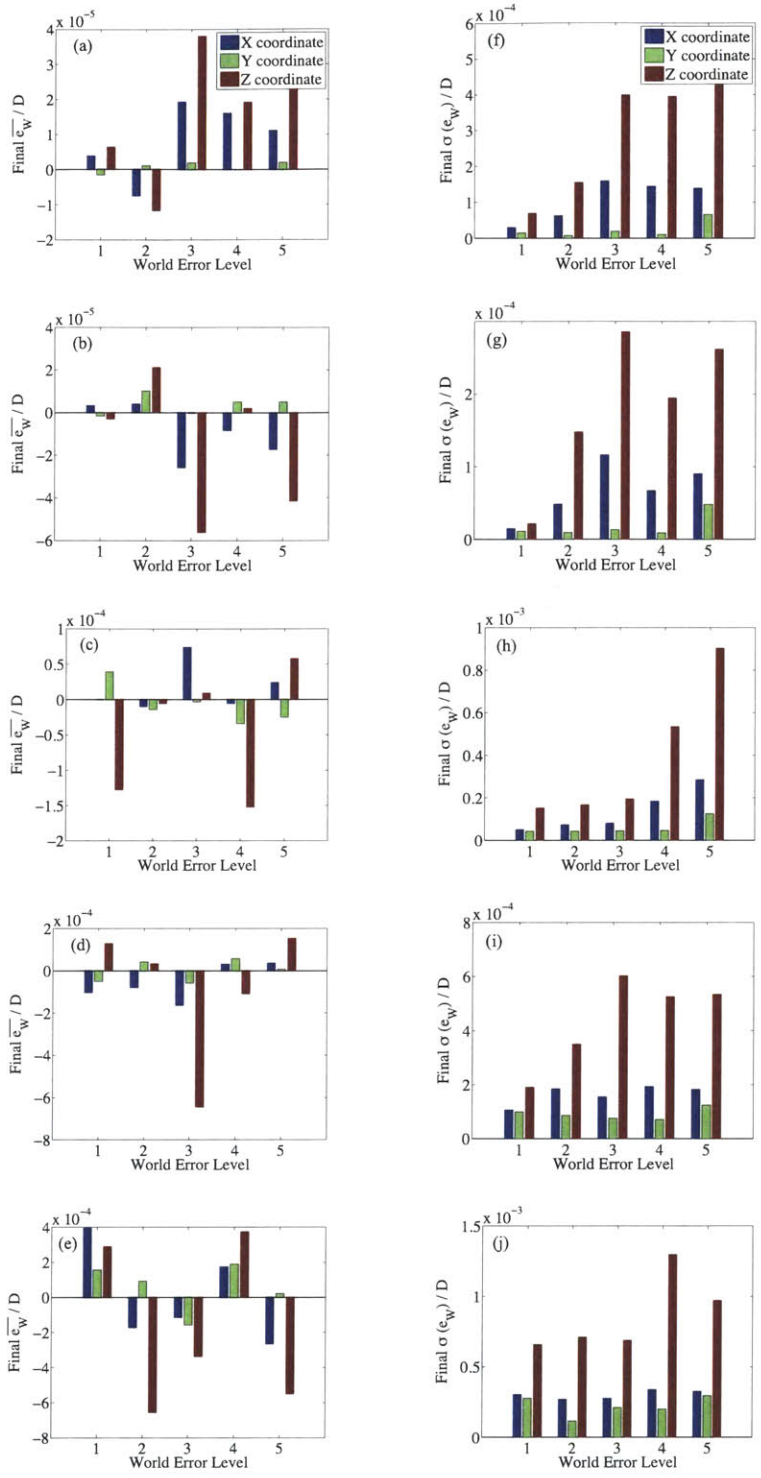


Figure C-20: Mean (a-e) and standard deviation (f-i) of the error in world point locations after the final iteration of calibration procedure applied to 3 simulated cameras. Each row of figures corresponds to an increasing value in the imposed image point measurement noise.

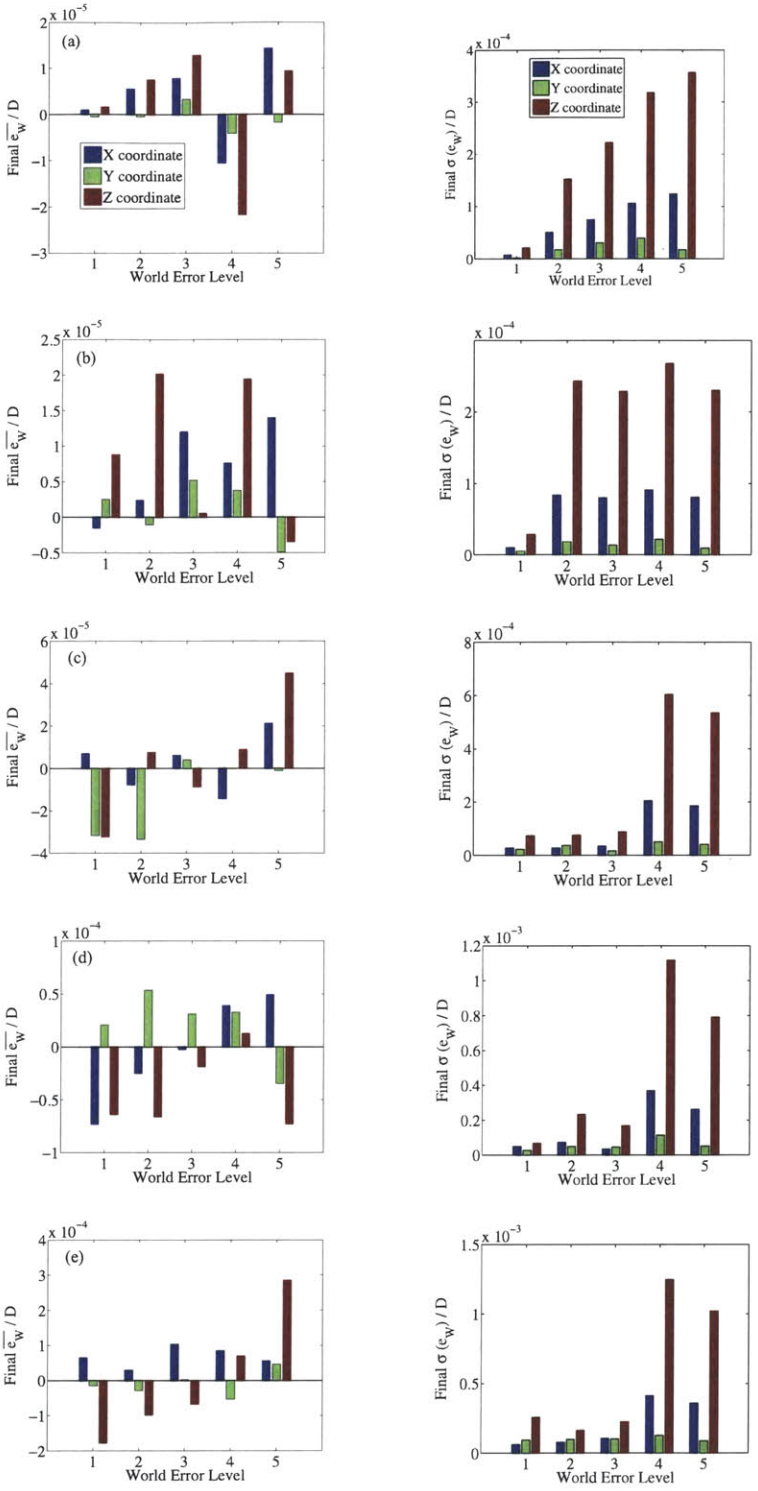


Figure C-21: Mean (a-e) and standard deviation (f-i) of the error in world point locations after the final iteration of calibration procedure applied to 13 simulated cameras. Each row of figures corresponds to an increasing value in the imposed image point measurement noise.