

© Copyright 2018

Charles Pecora

# Particle Tracking Velocimetry: A Review

Charles Pecora

A thesis

submitted in partial fulfillment of the  
requirements for the degree of

Master of Science in Aeronautics & Astronautics

University of Washington

2018

Committee:

Dana Dabiri, Chair

James Hermanson

Program Authorized to Offer Degree:

Aeronautics and Astronautics

University of Washington

**Abstract**

Particle Tracking Velocimetry: A Review

Charles Pecora

Chair of the Supervisory Committee:  
Dana Dabiri  
Aeronautics and Astronautics

Particle tracking velocimetry (PTV) and particle image velocimetry (PIV) are methodologies that allow for volumetric velocity measurements within a fluid flow by imaging the motion of suspended micron-size particles. The only difference between the techniques is that PTV tracks the motion of individual particles, while PIV measures the mean displacement of groups of particles. PIV is more commonly used than PTV, and thus the techniques are robust and have been reviewed extensively. PTV, however, offers the advantages of increased spatial resolution and decreased computational cost. The focus of this review is to present the work that has been done to improve PTV. The review begins with an overview of the equipment required for a PTV experiment. Then, data processing algorithms are discussed for identifying particle images,

mapping images to physical space, and tracking particles through time. Finally, post-processing techniques, which are used for calculating vorticity and pressure fields, are presented.

# TABLE OF CONTENTS

Chapter 1. Introduction .....	1
Chapter 2. Experimental Setup .....	4
2.1 Tracer Particles .....	6
2.1.1 Tracers in Liquid Flows .....	10
2.1.2 Tracers in Subsonic Gas Flows.....	13
2.1.3 Tracers in Supersonic Flows.....	14
2.2 Illumination.....	19
2.3 Area/Volume Illumination Optics.....	21
2.4 Camera .....	23
Chapter 3. Particle image identification.....	26
3.1 Non-Overlapped Particles.....	27
3.1.1 Threshold Binarization.....	27
3.1.2 Gaussian Centroid Estimation.....	28
3.2 Overlapped Particles .....	31
3.2.1 Threshold Binarization.....	31
3.2.2 Neural Network Particle Identification.....	32
3.2.3 Particle Mask Correlation .....	34
3.2.4 Optical Flow Feature Extraction.....	36

3.3	Particle Identification Comparison .....	37
3.3.1	Non-Overlapped Particle Identification Comparison .....	37
3.3.2	Overlapped Particle Identification Comparison.....	39
Chapter 4. Identification of particles' spatial locations .....		46
4.1	Spatial Localization in 2D .....	47
4.2	Spatial Localization in 3D .....	48
4.2.1	Photogrammetric PTV .....	48
4.2.2	Tomographic PTV .....	85
4.2.3	Synthetic Aperture PTV.....	90
4.2.4	Plenoptic Imaging .....	94
4.2.5	Holographic PTV .....	98
4.3	Optical Distortions and Calibration in PTV Systems .....	108
4.3.1	2D Calibration.....	109
4.3.2	Stereoscopic Calibration .....	110
4.3.3	3D Calibration.....	114
Chapter 5. Particle Tracking Techniques.....		122
5.1	Multi-Frame Approach .....	123
5.2	Cross-Correlation .....	127
5.3	Relaxation Methods .....	129
5.4	Neural Networks .....	141
5.5	Velocity Gradient Tensor.....	147
5.6	Polar Coordinate Similarity .....	150

5.7	Optimization Methods .....	153
5.7.1	Deterministic Annealing .....	153
5.7.2	Variational Approach.....	158
5.7.3	Genetic Algorithms.....	161
5.7.4	Ant Colony Optimization.....	164
5.7.5	Fuzzy Logic PTV .....	166
5.8	Delaunay Tesselation Methods .....	169
5.9	Voronoi Diagram Methods .....	174
5.10	Vision Based PTV.....	177
5.11	Outlier Detection.....	181
5.12	Combined tracking and localization for 3D PTV .....	189
5.12.1	Stereoscopic Pairing of Particle Tracks .....	190
5.12.2	Exclusion of Short Particle Tracks .....	190
5.12.3	Shake-The-Box .....	191
Chapter 6. Post-Processing .....		195
6.1	Interpolation .....	197
6.2	Pressure Calculation.....	207
Chapter 7. Conclusions .....		213

You may use the chapter/section, and sub-chapter/section designation system of your choice.

You may choose to use no numbers, Arabic numbers (1, 2, 3, etc.), Roman Numerals (I, II, III,

etc.) or written numbers (One, Two, Three, etc.). You may choose to list some, all, or none of your subchapter titles in the Table of Contents. You may choose to number your sub-headings, or not to number them. Whatever formatting choices you make, you must carry them out consistently through each chapter/section of your document.

## LIST OF FIGURES

Figure 1.1. Step response width of the estimated displacement with respect to the digital particle image diameter. ....	4
Figure 2.1. Typical layout of a 2D PTV system for use in a wind or water tunnel. ....	5
Figure 2.2. Typical layout of a 3D PTV system for use in a water tunnel. ....	6
Figure 2.3. Raw PIV images with a flat plate for Mie-scattering (A) and fluorescent light (B). .....	12
Figure 2.4. Step response of tracers of various sizes to a shockwave. ....	15
Figure 2.5. PIV measurement of the normal velocity of the aluminum nanoparticles (left) and TiO <sub>2</sub> particles (right) across an oblique shockwave located at 0 mm. ....	19
Figure 2.6. Dual-cavity Nd:YAG laser. ....	20
Figure 2.7. LED illumination for tomographic PIV. ....	20
Figure 2.8. Experimental setup using laser and LED illumination for investigation of bubbly flow. ....	21
Figure 2.9. Example optics for light sheet generation. ....	22
Figure 2.10. The use of dielectric mirrors to increase the light incident on a volume. ....	22
Figure 2.11. Dependence of signal to noise ratio on incident light exposure (left) and spatial pixel variance vs. the uniform signal level (right). ....	25
Figure 3.1. Single threshold binarization (a) and dynamic threshold binarization (b). <sup>61</sup> ..	27
Figure 3.2. Binary matrix showing three separate particle images. ....	28
Figure 3.3. One-dimensional three-point Gaussian estimator (a) and two-dimensional Gaussian regression (b). <sup>66</sup> ..	30
Figure 3.4. RMS error plotted for different particle position estimates a) without noise and b) with simulated photon noise. <sup>66</sup> ..	31
Figure 3.5. Correct barycenter recognition percent vs. (a) average number of pixels per particle and (b) number of particles in a 512x512 pixel image. ....	33
Figure 3.6. Flow chart for the modified cascade correlation particle identification algorithm. .....	35

Figure 3.7. RMS error of the center location with respect to the true value of a function of particle diameter and the standard deviation of the Gaussian Noise $\sigma_n$ (maximum peak intensity =200) <sup>65</sup> .....	38
Figure 3.8. (left) RMS error and error of the mean value as a function of the maximum peak intensity( $\sigma_n = 5$ , particle diameters = 4 pixels); (right) RMS error and the error of the mean value as a function of the maximum peak intensity ( $\sigma_n = 5$ , particle diameters = 4 pixels) <sup>65</sup> .....	38
Figure 3.9. Comparison of the cascade correlation method with the particle mask correlation method.....	40
Figure 3.10. Averaged particle location error versus particle overlap ratio for a) Poisson noise levels at infinity, 25, and 10. B) Gaussian noise levels at 2.5% and 5%. .....	41
Figure 3.11. Averaged particle location error versus particle diameter for different overlap ratios a) 2.5% Gaussian noise b) 5% Gaussian noise. ....	41
Figure 3.12. Averaged particle location error with various particle diameter ratios and noise levels. (a) 2.5% Gaussian noise versus particle overlaps. (b) 5% Gaussian noise versus particle overlap ratio. ....	43
Figure 3.13. Comparison of particle image reconstruction performance based on different assumptions. <b>a</b> Original image, <b>b</b> reconstructed image by current work, <b>c</b> comparison of profiles of the 6th row.....	43
Figure 3.14. Tracer density influence on (a) the normalized number of identified particles, $N_{pi}$ , and (b) the reliability, $E_r$ . <sup>72</sup> .....	43
Figure 3.15. Particle radius influence on (a) the normalized number of identified particles, $N_{pi}$ , and (b) the reliability, $E_r$ . <sup>72</sup> .....	44
Figure 3.16. Background mean noise influence on (a) the normalized number of identified particles, $N_{pi}$ , and (b) the reliability, $E_r$ . <sup>72</sup> .....	45
Figure 3.17. Background standard deviation noise influence on (a) the normalized number of identified particles, $N_{pi}$ , and (b) the reliability, $E_r$ . <sup>72</sup> .....	45
Figure 4.1. The geometry used in the pinhole model for determining 2D spatial coordinates of particles. ....	47

Figure 4.2. Setup for Stereoscopic PIV used to investigate a jet flow by Watanabe et al. (left) 3D PTV setup used to investigate the wake of a sphere by Doh et al. (right) .....	49
Figure 4.3. Experimental setup of the single camera 3D PTV system. ....	51
Figure 4.4. Principle of intersection of epipolar lines for photogrammetric PTV systems.	52
Figure 4.5. (a) The fraction of correctly estimated particle locations (within one voxel radius from the true particle location) and (b) the fraction of ghost particles compared to the number of true particles. ....	53
Figure 4.6. (a) Spatial particle location uncertainty and (b) the effective particles per pixel. ....	54
Figure 4.7. Flow diagram for iterative particle reconstruction. ....	56
Figure 4.8. Fraction of detected true particles vs seeding density for IPR and other 3D reconstruction algorithms.....	56
Figure 4.9. Mean 3D localization error vs particle seeding density for IPR and other 3D reconstruction algorithms.....	57
Figure 4.10. Schematics of stereoscope configurations with CMO (translation) and Greenough (angular displacement) type microscope systems. ....	58
Figure 4.11. Bias and RMS errors in the $x$ and $z$ -directions using stereo-PTV (circles) and stereo-PIV (squares) with Greenough type (outlined) and CMO type (filled). ....	60
Figure 4.12. A single camera using two off-center optical paths through a single lens. ..	61
Figure 4.13. Geometry of defocused image formation. <sup>93</sup> .....	62
Figure 4.14. Experimental setup with a laser beam being expanded to a sheet and scanned through a volume using a rotating eight face prism.....	67
Figure 4.15. The principle of defocused imaging. ....	68
Figure 4.16. Defocused images obtained from a black particle illuminated with a white backlight and three color filters. <sup>100,101</sup> .....	70
Figure 4.17. Particle image peak shifts due to color aberration: a color aberration due to white light illumination, b the skewed Gaussian distribution.....	71
Figure 4.18. Improved illumination configuration <sup>94</sup> .....	71
Figure 4.19. Out-of-plane location error as a percentage of the calibration range <sup>101</sup> . ....	71
Figure 4.20. Concept of the proposed multi-spectral 3D $\mu$ PTV system. ....	71

Figure 4.21. Implementation of 3D defocusing PIV.<sup>103</sup> ..... 73

Figure 4.22. Location of the optical axis for a rectangular imaging domain..... 73

Figure 4.23. Tetrahedral (left) and rectangular (right) imaging domains for defocusing PIV.  
..... 73

Figure 4.24. Error ratio of the out-of-plane to in plane measurements.  $Z$  and  $X$  are both  
normalized with respect to  $d$ , the length of a side of the equilateral triangle aperture  
arrangement.<sup>106</sup> ..... 75

Figure 4.25. Schematics of the anamorphic technique using cylindrical optics when applied to  
particle imaging. The left part of this picture illustrates how the imaging system works in  
the  $x$  direction, and the right part shows the system operation in the  $y$  direction.<sup>109</sup> 76

Figure 4.26. Anamorphic images of a fluorescent micron-sized particle traversed along the  
optical axis (the yellow numbers indicate the particle displacement in microns)<sup>109</sup> 78

Figure 4.27. (left) Error in axis ratio and (right) in-plane displacement error with respect to noise  
level for single camera astigmatism particle localization. .... 78

Figure 4.28. Qualitative uncertainty between astigmatism PTV techniques.<sup>114</sup> ..... 79

Figure 4.29. Flow image at 0.225s.<sup>124</sup> ..... 80

Figure 4.30. The concept of deconvolution microscopy application for three-dimensional micro-  
PTV technique; **a** Critical parameters of experimental setup such as specifications of  
objective lens, defocus distance between particles and lens's focus, thicknesses, and  
refractive indices of cover glass and immersion medium, **b** PSF of diffracted particle image  
in each indicated defocus distance, **c** Graphical diffraction images, which are drawn based  
on the corresponding PSF, and **d** A relation between a defocus distance ( $\Delta z$ ) and a radius of  
the out-most fringe ( $R_{OMF}$ ) ..... 82

Figure 4.31. **a** A raw diffraction image captured with an intensity resolution of 8-bit and a shutter  
speed of 30-ms. The dashed line indicates an interrupting micro-sphere. **b** After an image  
processing the 8-bit grayscale image is converted to a binary image that contains only 0's  
and 1's. **c** An identification of three-dimensional particle position is conducted by using  
axisymmetric 24-point data. **d** A few identified three-dimensional particle positions:  $A(x-, y-,$   
 $R_{OMF}$ -pixel numbers: 158, 178,16),  $B(140, 245, 22)$ ,  $C(240, 200,40)$  and  $D(264, 261, 29)$   
..... 83

Figure 4.32. Four camera tomographic PTV setup. <sup>48</sup> .....	86
Figure 4.33. Steps for handling tomographic PIV data. <sup>48</sup> .....	86
Figure 4.34. 2D representation of pixel and voxel interaction in tomo-PTV. ....	87
Figure 4.35. Volume definition with pixel interaction for telecentric imaging. ....	88
Figure 4.36. Tomographic reconstruction quality compared to exact intensity distribution as a function of setup and calibration.....	90
Figure 4.37. Schematic showing the reference plane and focal plane as well as illustrating the concept of parallax. <sup>131</sup> .....	92
Figure 4.38. Projection and backprojection using the homography fit method for synthetic aperture refocusing. <sup>135</sup> .....	92
Figure 4.39. Reconstruction quality, Q, as a function of number of cameras (a) and reference plane mapping error (b) in a 50 x 50 x 10 mm <sup>3</sup> volume. C is the particle seeding density. <sup>131</sup> .....	93
Figure 4.40. Schematic showing the application of micro-lenses for capturing the angle of incoming light in a plenoptic camera. ....	94
Figure 4.41. Errors in plenoptic imaging reconstructions using filtered refocusing (left) and MART (right). ....	96
Figure 4.42. Reconstructed 3D volume using deconvolution (top), and MART (bottom) algorithms. “o” indicates point locations and inverted grayscale is used. ....	97
Figure 4.43. Typical lens-less digital in-line holographic PTV setup. ....	100
Figure 4.44. Double-exposure off-axis holographic PTV setup using two separate reference beams. ....	100
Figure 4.45. Microscopic digital in-line holographic PTV setup. ....	101
Figure 4.46. Part of a raw holographic recording (a) and a reconstruction of the plane located 800 µm from the hologram plane. The in-focus particles appear as dark spots on the bright background.....	101
Figure 4.47. Variation of reconstruction efficiency and spatial resolution for a range of particle concentrations. ....	107
Figure 4.48. Original hologram (left) and residual hologram after using the IIPE algorithm (right). ....	108

Figure 4.49. The object points of the light sheet determined using a disparity vector. .	112
Figure 4.50. Calibration points obtained using laser reflection from measurement volume boundaries. ....	115
Figure 4.51. Schematic of the AFRT procedure used by Belden. P is the point of interest and $X_c$ is the image coordinate. ....	117
Figure 4.52. 3-D positioning errors from camera azimuthal misalignment.....	118
Figure 4.53. Steps required for an OTF calibration. ....	121
Figure 4.54. Comparison of reconstruction quality for OTF (black) B-spline (green) bilinear interpolation (red) and MART (blue) using synthetic astigmatic data. ....	122
Figure 5.1. The heuristics use for multi-frame particle tracking algorithms: a) nearest neighbor, b) three frame minimum acceleration, c) four frame minimum change in acceleration, and d) four frame best estimate. In each case the arrow points to the particle image that will be selected.....	124
Figure 5.2. Comparison of Nearest Neighbor (NN) method, 3 frame minimum acceleration (3MA), 4 frame minimum change in acceleration (4MA), and 4 frame best estimate algorithms(4BE) <sup>181</sup> .....	126
Figure 5.3. Particle positions from consecutive PTV frames with reference particle $i$ overlapped with candidate particle $j$ . ....	127
Figure 5.4. Initialization of match probabilities for reference particles in relaxation method. ....	131
Figure 5.5. Flow chart for the Brevis <i>et al.</i> hybrid algorithm. ....	135
Figure 5.6. Long particle tracks obtained using 3D implementation of the ICCRM algorithm. ....	137
Figure 5.7. Comparison of the unidirectional and bidirectional algorithms in the presence of removed and added particles.....	138
Figure 5.8. comparison of bidirectional (Dual RM-PTV) and unidirectional (RM-PTV) algorithms for four VSJ 3D standard images. ....	139
Figure 5.9. Inputs and activation of an artificial neuron (left) and structure of neurons in a feed-forward network (right).....	142
Figure 5.10. Example input and output of the neural network used by Grant and Pan. .	143

Figure 5.11. Visualization of a self-organizing map, in which particles from two frames are overlaid and organized until a match is found for each. ....	145
Figure 5.12. Displacement fields determined by the neural nets of Labonte (left) and Grant and Pan (right). ....	146
Figure 5.13. Comparison of the original SOM algorithm (left) and the modified SOM (right) for PIV VSJ standard image #01. ....	147
Figure 5.14. Flow chart for the pairing method. ....	149
Figure 5.15. Performance comparison of four-frame minimum acceleration (4PTV), binary image cross-correlation (BICC), and velocity gradient tensor (VGT) tracking methods for Karman vortex-shedding flow. ....	150
Figure 5.16. Concept of polar coordinate similarity between particles $i$ and $j$ . ....	150
Figure 5.17. Schematic of mapping between frames $X$ and $Y$ . <sup>204</sup> ....	154
Figure 5.18. Pseudocode for deterministic annealing minimization. ....	155
Figure 5.19. Comparison of the deterministic annealing approach (left) and super resolution PIV (right). ....	155
Figure 5.20. Performance of Deterministic annealing tracking (here labeled particle matching) for (a) rotational and (b) shearing flows. ....	156
Figure 5.21. Comparison of two deterministic annealing techniques with cross-correlation and nearest neighbor matching techniques in 3D. <sup>205</sup> ....	157
Figure 5.22. Nearest neighbor estimate of particle matches (left) and the result of regularization (right). ....	159
Figure 5.23. Relaxation parameter used as a fitness function by Ohmi and Panday. ....	162
Figure 5.24. Multi-point inverse crossover (left) and single-point partially matched crossover (right). ....	163
Figure 5.25. Original particle locations (left) and particle matching result (right) using the genetic algorithm for PIV standard image #301. ....	164
Figure 5.26. Concept of relaxation lengths for ant colony optimization. ....	165
Figure 5.27. Particle centroids from two exposures superimposed. Potential candidates are shown for each particle in exposure 1 with a vector. ....	167
Figure 5.28. Representation of membership based on a fuzzy logic input. ....	168

Figure 5.29. (a) Super-triangle and (b) triangle division for a new point, $P$ introduced in Delaunay Tessellation. ....	170
Figure 5.30. Translation and rotation of polygons for coefficient calculation. ....	171
Figure 5.31. Comparison of DT-PTV methods with the improvements suggested by Zhang <i>et al.</i> for VSJ standard image #301 (~0.06 ppp). ....	173
Figure 5.32. Particle images (left) and corresponding Voronoi diagram (right). ....	174
Figure 5.33. The development of the Voronoi cell characteristic curve. ....	176
Figure 5.34. Comparison of (a) VD-PTV (b) Relaxation (c) DT-PTV with % proper matches, $\eta$ , vs. the search radius, $R_s$ . $\overline{C_{PTV}}$ is the ratio of average particle displacement to average particle spacing. ....	177
Figure 5.35. Influence of erased (left) and added (right) particles on the performance of VD-PTV for the homemade flow with $\overline{C_{PTV}} = 8$ . ....	177
Figure 5.36. Proximity surface with (right) and without (left) modifications made by Paul <i>et al.</i> <sup>223</sup> ....	180
Figure 5.37. Comparisons of the original and modified VB PTV algorithms with a maximum velocity of 25 pixels. D1 and D2 are the modifications of Paul <i>et al.</i> <sup>223</sup> that use displacement and directional guidance, respectively. ....	180
Figure 5.38. Bidirectional computing to remove spurious vectors. ....	182
Figure 5.39. Simulated vortex flow with spurious vectors (left) and after spurious vectors have been removed using fuzzy logic algorithm. ....	184
Figure 5.40. Sum of weighted vector magnitudes and calculation of the average-weighted median estimate. ....	187
Figure 5.41. Comparison between universal outlier detection (NMT), distance-weighted outlier detection (DW-NMT), NMT with a variable neighborhood size (ANMT), DW-NMT with a variable neighborhood (ADW-NMT), and the proposed adaptive weighted angle and magnitude thresholding (AWAMT). ....	189
Figure 5.42. Stereo matching for 3D positions of 2D particle tracks. ....	190
Figure 5.43. Reconstructed trajectories color coded by time step. Short and long ghost trajectories are included. ....	191
Figure 5.44. Shake-the-box performance vs. a MART algorithm. ....	193

Figure 5.45. Temporal development of STB for 0.01 and 0.05 ppp with different image noise levels, $\sigma$ , (a) fraction of undetected particles, (b) particle position accuracy, and (c) fraction of ghost particles. ....	194
Figure 5.46. Isosurfaces of vorticity with superimposed particle tracks. ....	195
Figure 6.1. Velocity measurement locations used for estimation of spatial derivatives using (a) second order central difference, (b) fourth-order central difference, and (c) 8-pt circulation methods. ....	196
Figure 6.2. RMS of vorticity estimation on interpolation parameters, $H/\delta$ and $h/\delta$ . Unsmoothed vorticity (circle); bootstrap predicted variance ( $\nabla$ ); smoothed vorticity ( $\Delta$ ). $H/\delta=1.24$ (solid line); $H/\delta=0.62$ (dashed line). ....	198
Figure 6.3. Accuracy of interpolated velocity field as (a) bias error with 0% noise, (b) random error with 0% noise, and (c) random error with 6% noise. ....	199
Figure 6.4. Accuracy of out-of-plane vorticity field computed for different values of $R/\delta$ using four different differentiation methods; (a) bias error, (b) random error with 0% added noise, and (c) random error with 6% added noise. ....	200
Figure 6.5. Plot of the vorticity correlation coefficient (left) and normalized vorticity error (right) of the proposed ellipsoidal equation algorithms with and without smoothing. ....	202
Figure 6.6. Normalized mean square error versus normalized cut-off filter widths at various data densities, $\psi$ , and a sub-volume size of $L_v/\lambda=3.756$ . The dashed line shows the error for AGW for $\psi=2.412$ . ....	204
Figure 6.7. Isovorticity surfaces using RBF at $\omega_n=0.5$ (left) and AGW at $\omega_n=0.25$ (right). ....	205
Figure 6.8. Turbulence statistics from different velocity field reconstructions in a turbulent boundary layer. ....	207
Figure 6.9. Example of Voronoi tessellation in a two-dimensional space. ....	210
Figure 6.10. Comparison of spatial spectra of pressure fluctuations along 40 horizontal lines within the shear layer calculated using PIV (left) and PTV (right) algorithms. ....	213

## LIST OF TABLES

Table 2-1: Seeding particles used in liquid flows.....	10
Table 2-2: Particle response in turbulent flow ( $\eta=0.99$ ).....	12
Table 2-3: Seeding particles used in gas flows.....	12
Table 2-4: Cameras used in comparison by Hain <i>et al.</i> Cameras are labeled with superscripts <b>a</b> (CCD), <b>b</b> (CMOS), <b>c</b> (Intensified CMOS), <b>d</b> (identical with LaVision Flowmaster 3s), <b>e</b> (CMOS inframe time), and <b>f</b> (Identical with IDT XS5-i). fps refers to frames per second, and IF means interframe time. ....	23
Table 3-1: Comparison of identification algorithms for the PIV standard image 01. <sup>61</sup> ....	40
Table 4-1: Camera setups for 3D PTV, their respective error factors, E, and sensitivity coefficients, B, for in-plane measurements. $\Delta c$ is the horizontal distance between cameras and $\Delta d$ is the vertical distance between cameras. <sup>78</sup> .....	49
Table 4-2: Ant Colony optimization results for Stereo image pairing. ....	66
Table 4-3: Comparison between two-dimensional, Kajitani & Dabiri, and Grothe & Dabiri system uncertainties. ....	73
Table 5-1: Comparison of BCC and cross-correlation algorithm proposed by Saga et al. using VSJ standard image 01.....	129
Table 5-2: Comparison of four particle tracking algorithms using the PIV standard image 301. ....	134
Table 5-3: VSJ 3D standard images used for comparison of unidirectional and bidirectional relaxation algorithms. ....	138
Table 5-4: Comparison of EPTV on a VSJ standard image #301 <sup>67</sup> .....	140
Table 5-5: Comparison of the number of calculated vectors using the polar coordinate similarity method (PCSS), cross correlation (BICC), and 4 frame in line tracking (4-PTV). ....	151
Table 5-6: Comparison of the number of calculated vectors and the number of valid vectors using polar coordinate similarity, 3-frame minimum acceleration (3FIT), and relaxation (NRX) methods using VSJ PIV standard image #301 (0.06 ppp). <sup>71</sup> .....	152
Table 5-7: Computational speed of tracking algorithms for 40 particle tracks. <sup>205</sup> .....	158
Table 5-8: Comparison of four frame tracking (FIT), binary image cross correlation (BCC), relaxation (NRX), and variational approach (VAR) for the VSJ 301 image sequence. ....	160

Table 5-9: Testing of the variational approach for VSJ standard image 331. ....	160
Table 5-10: Comparison of ant colony optimization using the relaxation minimization to other PTV tracking algorithms using VSJ image #301 (0.008-0.015 ppp). ....	166
Table 5-11: Confidence outputs based on interaction of fuzzy logic inputs. ....	168
Table 5-12: Comparison of computation time and number of vectors obtained for BICC and DT- PTV tracking algorithms. ....	172
Table 5-13: PTV results with known and unknown particle locations for VB PTV and previous tracking algorithms using VSJ #301. ....	181
Table 5-14: Fuzzy logic rule base for identifying spurious vectors. ....	183
Table 5-15: Outlier detection removal rate for PIV and PTV data with added outliers. ....	185
Table 24: Global pressure RMS error for PTV pressure extraction techniques for clean and noisy data. ....	212

## **ACKNOWLEDGEMENTS**

I would like to thank both Professor Dana Dabiri for his guidance and contribution to this thesis, and the University of Washington department of Aeronautics and Astronautics for the financial support during my studies and research.

## Chapter 1. INTRODUCTION

Turbulent fluid flow occurs in all aspect of our lives, both in nature and within our engineered society. At the very least, it affects the atmosphere and air that we and nature breathes; the oceans, seas, rivers and therefore our natural environment; the heat transfer within it that allows for life; it is what is responsible for allowing airplanes to fly and boats and ships to navigate the oceans; when combined with chemical reactions, it is responsible for the power generation and its consequential pressure and temperature distributions within our vehicles' (i.e., cars, ships, airplanes) engines; and it is responsible for the oxygen, heat and pressure transport when we breath air into our lungs, as well as the blood flow within our hearts and our cardiovascular system and its necessary pressure distributions that keep us alive and healthy. Because of its all-pervasiveness in all matters of life, understanding turbulent flow is paramount for the progress of our society, and therefore its accurate measurements are critical to attaining this achievement.

Historically, flow visualization has been the primary technique for analyzing turbulence. As early as the 1400s, Leonardo da Vinci recognized that flow needed to be visualized and used fine particles to trace turbulent flows in the aortic heart valve<sup>1</sup>. Ludwig Prandtl observed flow separation over an airfoil and other two-dimensional bodies using floating mica particles in a water tunnel<sup>2</sup>. Fage and Townend<sup>3</sup> quantified flow visualization by capturing streak photographs of fluid flow and manually measuring fluid motion<sup>4</sup>. Pickering and Halliwell, using Laser Speckle Photography (LSV) methodologies, heavily seeded the flow with reflective particles such that individual particles could not be distinguished, illuminated a cross-section of the flow of interest with a laser sheet, and photographically recorded sequences of images of the flow evolution. Using LSV methodologies, they were able to quantify the kinematic flow field, thereby initiating the field

of automated quantitative particle velocimetry<sup>5-8</sup>. Shortly thereafter, reduced seeding allowed for imaging of individual particles, giving rise to Particle Image Velocimetry (PIV), and Particle Tracking Velocimetry (PTV).

PIV techniques do not match individual particles to determine flow velocities. Instead, an interrogation window subsamples the image pair, which are analyzed with a spatial correlation method, resulting in a single velocity vector that represents the average velocity of particles within the interrogation window. The interrogation window is then systematically moved through the image pair, yielding an evenly spaced velocity vector field. This averaged result introduces smoothing and bias errors in the presence of sharp flow gradients, such as in boundary layers and in shockwaves, thereby reducing the accuracy of the measurements. In addition, the size of this correlation window limits the spatial resolution of PIV, as the interrogation windows are typically not overlapped more than 50%.

The field of PTV became prevalent when Adamczyk and Rimai<sup>9</sup> introduced an automatic particle tracking algorithm for low particle densities. This improved upon previous work in PTV, which was founded on manual measurements of particle displacements<sup>10</sup>. This early work streak-exposed each frame; however, modern PTV uses singly-exposed frames with a pulsed laser, and matches particle pairs from frame to frame. Particle velocities are then found by calculating the displacement between matched particle pairs, and dividing by their displacement times.

Naturally, the question arises as to whether the limitations of PIV, as described above, can be overcome by using PTV. Using simulated images of a step-like displacement profile, Kähler *et al.*<sup>11</sup>, studied the resolution limits of PIV. Characterizing the vector spatial resolution as the width of the response function (step response width or SRW), they compared PIV processed with 16x16

windows, single-pixel ensemble-correlations, and PTV (see Figure 1.1), and found that PTV showed an order of magnitude lower SRW compared with PIV. In addition, they showed that the dynamic spatial range (DSR)<sup>12</sup> was almost three orders of magnitude larger than PIV processed with 8x8 windows. PTV therefore promises the opportunity to substantially increase the spatial resolution as well as the measurement accuracies.

While PIV has generously thrived since its development,<sup>4,13-21</sup> PTV has until recently received less attention. It is therefore the goal of this article to provide a review of Particle Tracking Velocimetry, focusing on methods and techniques that have been implemented towards developing and improving the PTV methodology, rather than on its applications. In this regard, Chapter 2 discusses experimental components and their setup. Chapter 3 through Chapter 5 are laid out in the steps that PTV processing algorithms typically operate: Chapter 3 discusses particle identification algorithms within images; Chapter 4 then discusses algorithms that use these to identify particle locations in real space for both 2D and 3D methodologies; Chapter 5 then discusses algorithms on how the particles are tracked between image pairs in both 2D and 3D. 5.12 explains methods that do not follow the sequential approach typically used in PTV methodologies. Chapter 6 discusses post-processing techniques, and finally, Chapter 7 provides concluding remarks.

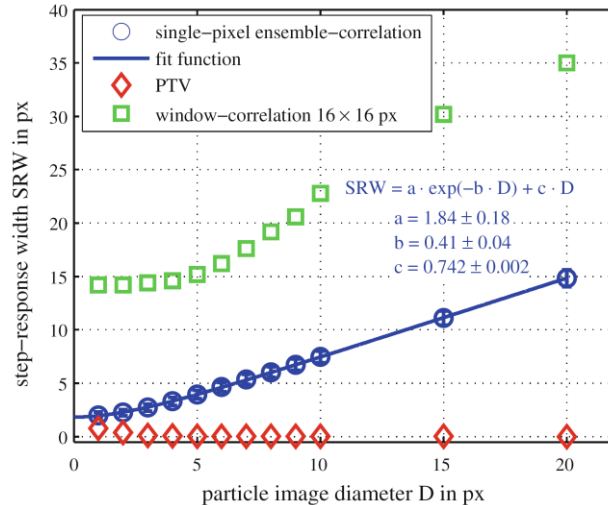


Figure 1.1. Step response width of the estimated displacement with respect to the digital particle image diameter.<sup>11</sup>

## Chapter 2. EXPERIMENTAL SETUP

Figure 2.1 and Figure 2.2 show typical setups for 2D and 3D PTV systems, respectively, and as can be seen, they are not different from setups used for 2D and 3D PIV systems. In both cases, the flow is first seeded with particles that act as flow tracers. A pulsed laser light source and appropriate optics are used to create either a thin light sheet for 2D PTV, or a light volume for 3D PTV. For 2D setups, a CCD (charged couple device) or CMOS (complementary metal-oxide semiconductor) video camera, positioned perpendicular to the light sheet, is synchronized with the pulsed laser in order to capture the positions of the tracer particles at discrete points in time. For 3D setups, multiple CCD or CMOS video cameras are situated about the volume of interest, which are also synchronized with the illuminating laser to capture tracer particle images. A data acquisition system records these images from the camera(s) in real-time. We refer the reader to PIV reviews that detail specifics of these experimental components<sup>4,13,20</sup>.

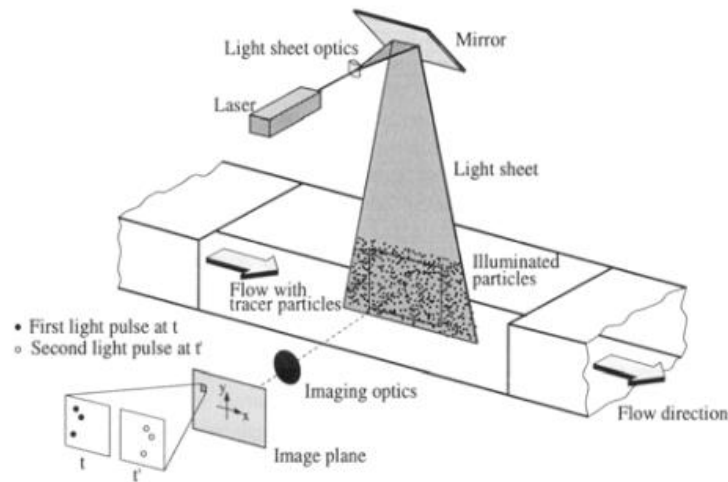


Figure 2.1. Typical layout of a 2D PTV system for use in a wind or water tunnel<sup>18</sup>.

Time-resolved PTV experiments track fluid particles for multiple consecutive time steps in order to analyze transient flow phenomena. Standard CCDs and Nd:YAG lasers, however, limit data acquisition to a rate of 30Hz, which is far lower than the frequency at which turbulent flow fluctuations typically occur. The recent development of high-speed CMOS cameras and Nd:YLF lasers have made time-resolved measurements on the order of kHz possible<sup>19</sup>. Recently, time-resolved PTV systems have become capable of making measurements up to 400 kHz. Though image quality and resolution are compromised for speed, these systems allow for acquisition of flow spectra at frequencies that are an order of magnitude larger than what hot-wires and LDV can provide.<sup>22,23</sup>

In this section, we discuss experimental setup features that are relevant to PTV. Section 2.1 discusses tracer particles and their response times, as this is directly relevant to PTV velocity measurements. Sections 2.2-2.4 discuss the interrogation region's illumination and its associated optics, and image acquisition cameras. As these latter components are common to PIV, they are briefly reviewed, and the reviewer is referred to PIV literature for further details on these topics.

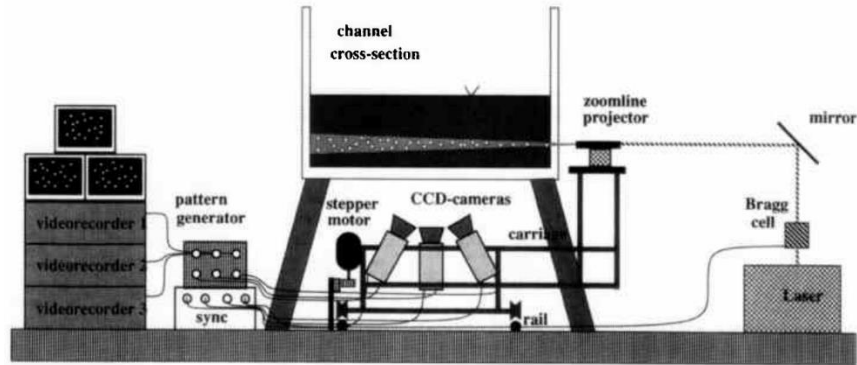


Figure 2.2. Typical layout of a 3D PTV system for use in a water tunnel.<sup>24</sup>

## 2.1 TRACER PARTICLES

In their review and analysis of tracer particles for particle velocimetry, Melling<sup>25</sup> states that a good tracer particle achieves two goals: it follows the local fluid motion and scatters sufficient light for imaging. These requirements are a tradeoff, as smaller particles accurately follow fluid motion with higher frequency response, while larger particles scatter more light for imaging. An excellent discussion of tracer particle analysis and selection can be found in Chapter 2 of Particle Image Velocimetry<sup>19</sup>, which is clearly relevant to PTV as well. In this section, we summarize considerations for selecting appropriate tracer particles to achieve accurate flow tracing.

The most basic approach for characterizing a particle response time is by assessing its response to a step function change in fluid velocity. Given that PTV tracer particles' diameters and their velocity lag to this sudden velocity changes are small suggests that the particle Reynolds number ( $Re_p$ ), based on the difference between the flow and the particle velocity (i.e. particle slip velocity), is also small, and that the particle drag is due only to viscous forces. In this case, the particle motion equation is

$$\frac{dU_p}{dt} = K(U - U_p), \quad (2.1)$$

$$\text{Re}_p = (U - U_p)d_p / \nu_f, \quad (2.2)$$

where  $U$  is the steady gas velocity,  $U_p$  is the particle velocity, and  $K = 3C_D \text{Re}_p \mu_f / 4\rho_p d_p^2$  is the entrainment coefficient, which from Stoke's law is given as  $18\mu_f / \rho_p d_p^2$ . Here,  $\mu_f$  and  $\nu_f$  are the fluid dynamic and kinematic viscosity, respectively, and  $\rho_p$  and  $d_p$  are the particle's density and diameter, respectively. Given that the particle is not in motion at  $t=0$ , this equation integrates to

$$U_p(t) = U \left[ 1 - e^{-(t/t_s)} \right], \quad (2.3)$$

where  $t_s$  is the particle relaxation time, and is given by

$$t_s = \frac{\rho_p d_p^2}{18\mu_f}. \quad (2.4)$$

The particle relaxation time should be less than the smallest time scale of the flow, in order to accurately represent the local fluid velocity.<sup>26</sup>

Particle response to unsteady flow (i.e. turbulence), however, is not as straightforward. In this case, Mei<sup>27</sup> modified the Maxey-Riley equation for low particle Reynolds number considerations for both solid particles and bubbles. To assess the response of the particles to the high-frequency turbulent fluctuations, the velocity was allowed to oscillate as

$$U(t) = \tilde{U}(\omega)e^{-i\omega t}, \quad (2.5)$$

where  $\tilde{U}(\omega)$  is the oscillation amplitude, and the particle Reynolds number is now defined as  $\text{Re}_p = \tilde{U}d_p / \nu_f$ . Using Equation 2.5, Mei expressed the modified Maxey-Riley equation in terms of frequency, and derived an energy response function for the particle that was a function of both the particle Stokes number,  $\varepsilon = \left[ (\omega d_p^2) / (8\nu_f) \right]^{0.5}$  and particle-to-fluid density ratio,  $\rho = \rho_p / \rho_f$

<sup>28</sup>. Then, defining cut-off frequencies of the particle based on either 50% or 200% of the energy response, Mei derived approximations for the high-frequency cutoff for density ratios greater than one (solids) less than one (bubbles). For example, for  $\rho < 0.561$ , the cut-off frequency is

$$f_{cut-off} \approx \frac{4\nu_f}{\pi d_p^2} \left[ 2.38^{0.93} + \left( \frac{0.659}{0.561 - \rho} - 1.175 \right)^{0.93} \right]^{2/0.93}, \quad (2.6)$$

while for  $\rho > 1.621$ , the cut-off frequency is

$$f_{cut-off} \approx \frac{4\nu_f}{\pi d_p^2} \left[ \left( \frac{3}{2\rho^{0.5}} \right)^{1.05} + \left( \frac{0.932}{\rho - 1.621} \right)^{1.05} \right]^{2/1.05}, \quad (2.7)$$

where from the definition of the Stokes number, the high frequency cut-off can be related to the Stoke's cut-off value

$$\mathcal{E}_{cut-off} \approx \frac{d_p}{2} \sqrt{\frac{\pi f_{cut-off}}{\nu_f}}. \quad (2.8)$$

Then by combining the appropriate estimate for the high-frequency cut-off from Equations 2.6 and 2.7 with Equation 2.8, one can find the largest particle diameter that would that would respond to the unsteady fluid flow.

For cases in which the density ratio is in between these two ranges, the energy transfer function is shown to be less than two, thereby implying a very good particle response. For neutrally buoyant particles,  $\rho = 1$ , the energy transfer function is equal to one, implying perfect particle response. In further evaluations of the particle response to isotropic turbulence, turbulence energy and Taylor micro-scale structures, Mei concluded that the particle inertia parameter (the inverse of the particle response time) defined as

$$\beta = \frac{9}{2} \frac{\nu}{(\rho + 1/2)a^2}, \quad (2.9)$$

should be large for solid particles in gas flow, and also large with  $\rho \approx 1$  for solid particles in liquids.

By further considering the motion of particles in turbulent flow, Melling<sup>25</sup> derived the energy response function for the simplified case of high particle-to-fluid density ratios and low particle Reynolds number, and showed that the ratio of fluctuation intensities of particle and fluid motions is

$$\frac{\overline{u_p^2}}{\overline{u_f^2}} = (1 - t_s \omega_c)^{-1}, \quad (2.10)$$

where  $\omega_c$  is the highest turbulent frequency of interest, and  $\overline{u_p^2}$  and  $\overline{u_f^2}$  are the particle and fluid velocity fluctuations, respectively. Lower relaxation times allow for better flow tracking by a particle. For example, a relaxation time of  $8.3\mu\text{s}$  roughly corresponds to a  $1\mu\text{m}$  water droplet in air.

These previous works and others<sup>29-32</sup> have suggested that a low particle Stokes number, defined as the ratio of the tracer relaxation time to the Kolmogorov time scale of the flow, would allow for a tracer to reliably follow fluid motion, Recently, however, Mathai et al.<sup>33</sup> suggested that even for sufficiently small Stokes numbers, the density ratio can cause an increase of acceleration variance. To understand this, they derive the non-dimensional particle equation of motion,

$$\ddot{\mathbf{x}} = \zeta \frac{D\mathbf{u}}{Dt} + \frac{1}{St} (\mathbf{u} - \dot{\mathbf{x}}) + \frac{1}{Fr} \hat{e}_z, \quad (2.11)$$

$$St \equiv \frac{d_p^2}{12\zeta\nu\tau_n}, \quad (2.12)$$

$$Fr \equiv \frac{a_n}{(\zeta - 1)g}, \quad (2.13)$$

where  $\ddot{\mathbf{x}}$  is the acceleration of a tracer particle,  $\zeta$  is the density ratio defined as  $3\rho_f / (\rho_f + 2\rho_p)$ ,  $D\mathbf{u}/Dt$  is the fluid acceleration,  $St$  is the Stokes number,  $Fr$  is the Froude number,  $\mathbf{u}$  is the fluid velocity,  $\dot{\mathbf{x}}$  is the particle velocity,  $d_p$  is the particle diameter,  $\nu$  is the fluid kinematic viscosity,  $\tau_\eta$  is the Kolmogorov time scale,  $a_\eta$  is the turbulent acceleration, and  $g$  is the gravitational constant. Note that all dimensionless numbers and the fluid acceleration term are adjusted with the density ratio,  $\zeta$ . They therefore suggest that an additional constraint for tracer selection should be that the ratio of the Stokes to the Froude number should be less than unity. This constraint is particularly relevant for flows with small  $Fr$ , as this indicates a significant contribution from gravitational acceleration.

### 2.1.1 *Tracers in Liquid Flows*

As can be concluded from the previous section, one way to ensure fast response times of tracer particles is to have a particle density nearly the same as the fluid. In liquid applications, near unity density ratio is achievable, and thus tracer sizes can be increased to improve light scattering. For this reason, less powerful illumination sources can be used for liquid than for gas in experiments. An overview of tracers that have been documented for liquid experiments is shown below in Table 2-1.

Table 2-1: Seeding particles used in liquid flows.<sup>25</sup>

Material	$d_p$ ( $\mu\text{m}$ )	Laser	CW power or energy, time	Light sheet		Reference
				$w$ (mm)	$t$ (mm)	
TiO <sub>2</sub>	3	Nd:YAG				Longmire and Alahyari (1994)
Al <sub>2</sub> O <sub>3</sub>	9.5	Ruby	2 J, 30 ns	100	0.8	Liu <i>et al</i> (1991)
Conifer pollen ( $\rho = 1000 \text{ kg m}^{-3}$ )	50–60	Ar ion	1–2 W			Westergaard <i>et al</i> (1993) McCluskey <i>et al</i> (1995) Gallagher and McEwan (1996)
Polymer ( $\rho = 1030 \text{ kg m}^{-3}$ )	30	Ar ion	0.5–5 W		0.5	Draad and Westerweel (1996) McCluskey <i>et al</i> (1996)
Phosphorescent polymer	80	Ar ion	5 W		1	Willert and Gharib (1991)
Fluorescent	50	Nd:YAG				Hart (1996)
	20	Cu vapour	45 W		1	Roth <i>et al</i> (1995)
Polystyrene ( $\rho = 1050 \text{ kg m}^{-3}$ )	500					Khoo <i>et al</i> (1992)
	15	Ruby	25 mJ, 20 ns			Zhang <i>et al</i> (1996)
Thermoplastic ( $\rho = 1020 \text{ kg m}^{-3}$ )	6	Nd:YAG		50	2	Hassan <i>et al</i> (1994)
Reflective ( $\rho = 1010 \text{ kg m}^{-3}$ )	60	Ar ion	18 W			Grant <i>et al</i> (1992)
	30	Ar ion	12–18 W	200		Grant and Wang (1994)
Metallic coated	4	Ar ion	2 W		2	Magness <i>et al</i> (1993)
	14	Ar ion			1	Johari <i>et al</i> (1996)
Microspheres ( $\rho = 700 \text{ kg m}^{-3}$ )	<30	Ar ion				Graham and Soria (1994)
H <sub>2</sub> bubbles		Ar ion	1 W		0.3	Dieter <i>et al</i> (1994)

Fluorescent-dyed particles, which emit a different wavelength of light than they absorb, are commonly used in micro-PTV and liquid flows for the purpose of removing reflected light and noise from images. They are typically used with optical filters, which can isolate light fluoresced by fluorescent particles from light reflected by surfaces. Two commonly used fluorescent dyes are Fluorescein and Rhodamine 6G.<sup>19,34</sup> Fluorescent-dyed particles have also been used for particle velocimetry measurements near walls to allow for improved laser flare removal from acquired images.<sup>35,36</sup> KR620-doped polystyrene microspheres, for example, have demonstrated the ability to effectively reduce laser reflections, allowing for near-wall measurements, as shown in Figure 2.3.

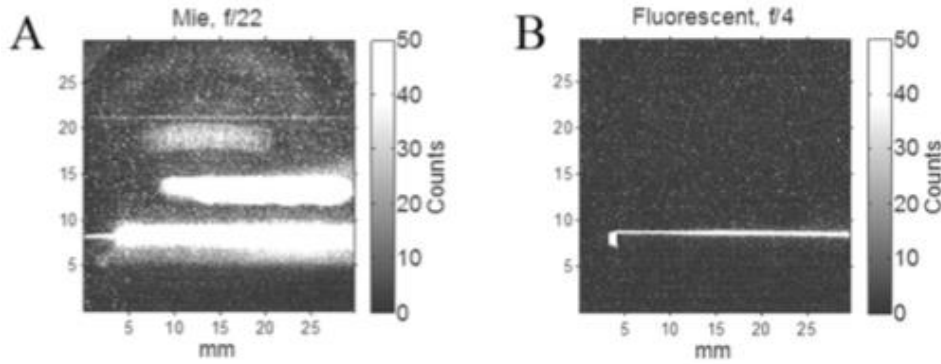


Figure 2.3. Raw PIV images with a flat plate for Mie-scattering (A) and fluorescent light (B)<sup>35</sup>.

According to Melling, particle seeding in liquid flows is typically done by simply mixing solid particles into the flow. Overbruggen *et al.*<sup>37</sup> suggest that in some liquid flows, solid particle sedimentation can be an issue, and thus they introduced a gaseous bubble seeding technique. The technique generated air bubbles in water with a median diameter of 10.64  $\mu\text{m}$  and a standard deviation of 5.3  $\mu\text{m}$ . Despite the low density ratio of the bubbles to water, the relaxation time is estimated to be 0.629 ns and the buoyancy effects were assumed to be negligible.

Table 2-2: Particle response in turbulent flow ( $\eta=0.99$ ).<sup>25</sup>

Particle	$\rho_p$ ( $\text{kg m}^{-3}$ )	Gas ( $10^5$ Pa)	Density ratio $s$	Viscosity $\nu$ ( $\text{m}^2 \text{s}^{-1}$ )	$f_c$ (kHz)	$Sk_c$	$d_p$ ( $\mu\text{m}$ )
TiO <sub>2</sub>	3500	Air (300 K)	2950	$1.50 \times 10^{-5}$	1	0.0295	1.44
					10		0.45
Al <sub>2</sub> O <sub>3</sub>	3970	Flame (1800 K)	20250	$3.00 \times 10^{-4}$	1	0.0113	2.46
					10		0.78
Glass	2600	Air (300 K)	2190	$1.50 \times 10^{-5}$	1	0.0342	1.67
					10		0.53
Olive oil	970	Air (220 K)	617	$1.45 \times 10^{-5}$	1	0.0645	3.09
					10		0.98
Microballoon	100	Air (300 K)	84.5	$1.50 \times 10^{-5}$	1	0.1742	8.50
					10		2.69

Table 2-3: Seeding particles used in gas flows.<sup>25</sup>

Material	$d_p$ ( $\mu\text{m}$ )	Laser	Pulse energy, pulse time	Light sheet		Reference
				$w$ (mm)	$t$ (mm)	
TiO <sub>2</sub> ( $m = 2.6$ , $\rho = 3500 \text{ kg m}^{-3}$ )	<1	Nd:YAG	10 mJ, 20 ns	15	0.3	Reuss <i>et al</i> (1989)
TiO <sub>2</sub> , ZrO <sub>2</sub>	0.7–1	Nd:YAG	110 mJ, 12 ns			Paone <i>et al</i> (1996)
Al <sub>2</sub> O <sub>3</sub> ( $m = 1.76$ , $\rho = 3970 \text{ kg m}^{-3}$ )	0.3	Nd:YAG	400 mJ		0.2	Muniz <i>et al</i> (1996)
	3	Nd:YAG	9 mJ, 6 ns			Anderson <i>et al</i> (1996)
	0.8	Ruby	20 ns	150	$\approx 1$	Krothapalli <i>et al</i> (1996)
Polycrystalline	30	Nd:YAG	135 mJ, 6 ns			Grant <i>et al</i> (1994)
Glass	30	Ruby	30 mJ, 30 ns			Schmidt and Löffler (1993)
Oil smoke	1	Ruby	5 J			Stewart <i>et al</i> (1996)
Corn oil	1–2	Nd:YAG	100 mJ			Jakobsen <i>et al</i> (1994)
Oil	1–2	Nd:YAG	120 mJ		0.4	Westerweel <i>et al</i> (1993)
Olive oil ( $m = 1.47$ , $\rho = 970 \text{ kg m}^{-3}$ )	1.06	Nd:YAG	70 mJ, 16 ns	200	0.5	Höcker and Kompenhans (1991)
						Fischer (1994) Raffel <i>et al</i> (1996)

### 2.1.2 Tracers in Subsonic Gas Flows

Tracer particles in gas flow typically have a density ratio much greater than 1. For a given gas and tracer particle density, the particle diameter must be selected to achieve a sufficiently small relaxation time. Proper generation of seeding particles is therefore important,<sup>25</sup> and is typically done with a Laskin nozzle.<sup>38</sup> Table 2-2 shows that for a 1 kHz response, 2–3 $\mu\text{m}$  diameter tracer particles suffice, while a 10 kHz response requires less than 1 $\mu\text{m}$  diameter particles; these frequency limits were calculated using the simplified frequency response for high particle-to-fluid density ratios given in Equation 2.9.<sup>25</sup> Some applications of particle imaging in gas flows to various experiments are listed in Table 2-3, showing that they generally adhere to the calculated maximum diameters.

In gas flows, liquid droplets are typically introduced using a Laskin nozzle, which reproducibly creates droplets on the order of 1  $\mu\text{m}$  at concentrations as high as 1–10  $\text{mm}^{-3}$ . Commercially-available smoke generators, on the other hand, create a water or oil based smoke using the principle of evaporation followed by condensation. Though less consistent, these smoke generators can create higher particle concentrations and smaller tracer particles. Solid particles are commonly seeded via atomization, creating an aerosol.<sup>25</sup>

Helium-filled soap bubbles (HFSB) have recently been implemented as tracer particles in wind tunnel experiments<sup>39-41</sup>. Unlike the micron-scale tracer particles typically used, these soap bubbles are estimated to be around 400 micrometers and neutrally buoyant. The larger tracers scatter light with intensity several orders of magnitude higher than micron sized droplets. Due to this increased brightness, the HFSB tracers are useful in PTV experiments with wind tunnels and measurement volumes that are two orders of magnitude larger than that of a conventional experiment. HFSB generation is achieved by feeding coaxial channels with constant flow rates of helium, bubble fluid solution (water, glycerin, and soap), and air. The coaxial channels terminate with a small circular orifice. The flow rates of helium and bubble fluid can be adjusted so as to match the density of air for a minimum response time. The minimum characteristic response time was been reported to be 10  $\mu\text{s}$ .<sup>37</sup>

### 2.1.3 *Tracers in Supersonic Flows*

In transonic and supersonic flows, particle diameters must be further reduced, as shockwaves create substantial instantaneous changes in velocity,<sup>26</sup> thereby requiring faster particle response times for accurate flow determination. Seeding particles for supersonic flows are typically liquids, which can be seeded using a Laskin nozzle.<sup>38</sup> Using Equations 2.1 through 2.4. Chen and Emrich (1963) estimated the time required for a tracer particle at rest in a stream of velocity  $u$  to acquire the fraction  $v/u$  ( $v$  is the particle velocity) of the final speed in a shock tube with a pressure ratio of 2.0 in Figure 2.4.

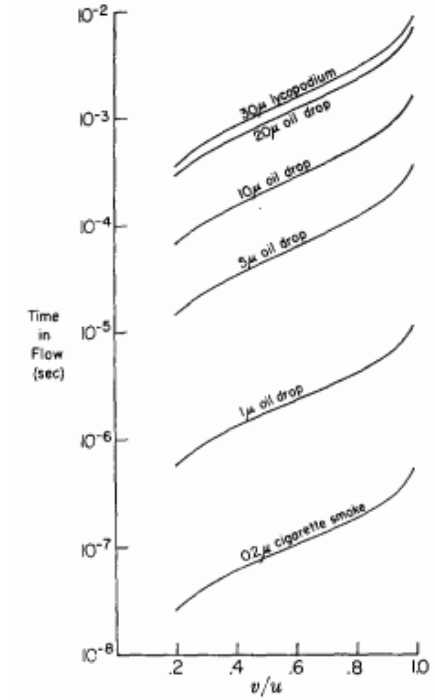


Figure 2.4. Step response of tracers of various sizes to a shockwave<sup>26</sup>.

Melling<sup>25</sup> recognized the expression for the drag coefficient needed to be modified for compressibility effects in transonic & supersonic flow, and therefore proposed the modified Stokes drag, by incorporating the particle Knudsen number,  $Kn_p$ , as shown in Equation 2.14. This was then used in the entrainment coefficient within a two-dimensional form of Equation 2.1 to obtain the particle relaxation times.

$$C_D = \frac{24}{Re_p(1 + Kn_p)}. \quad (2.14)$$

Tedeschi *et al.*<sup>42</sup> proposed a model for the drag coefficient of tracer particles that takes into account compressibility and rarefaction effects in all rarefaction regimes ( $Kn_p$ ) when  $M \leq 1$  and  $Re_p \leq 200$ :

$$C_D = \frac{24}{Re_p} k \left[ 1 + 0.15 (k Re_p)^{0.687} \right] \xi(Kn_p) C, \quad (2.15)$$

where

$$k = \left(1 + \frac{9}{2} Kn_p\right)^{-1} \quad (2.16)$$

$$C = 1 + \frac{Re_p^2}{Re_p^2 + 100} e^{-0.225/M^{2.5}}, \quad (2.17)$$

$$\xi(Kn) = 1.177 + 0.177 \frac{0.851Kn_p^{1.16} - 1}{0.851Kn_p^{1.16} + 1} \quad (2.18)$$

and  $M$  is the relative Mach number between the particle and the fluid. This expression is an extension of Cunningham's<sup>43</sup> method to higher velocities, and was shown to agree with experimental results across an oblique shock wave.

Loth<sup>44</sup> introduced a drag model for tracer particles that divides the problem into a compression-dominated ( $Re \leq 45$ ) and a rarefaction-dominated ( $Re \geq 45$ ) regime. Within the compression-dominated regime, the drag coefficient is:

$$C_D = \frac{24}{Re_s} \left[1 + 0.15 Re_s^{0.687}\right] H_M + \frac{0.42C_M}{1 + \frac{4200G_M}{Re_s^{1.16}}}, \quad (2.19)$$

$$H_M = 1 - \frac{0.258C_M}{1 + 514G_M}, \quad (2.20)$$

$$G_M = \begin{cases} 1 - 1.525Ma_s^4 & Ma_s < 0.89 \\ 0.0002 + 0.0008 \tanh[12.77(Ma_s - 2.02)] & Ma_s > 0.89 \end{cases}, \quad (2.21)$$

$$C_M = \begin{cases} 5/3 + 2/3 \tanh[3 \ln(Ma_s / 1.5)^2] & Ma_s < 1.45 \\ 2.044 + 0.2 \exp[-1.8 \{\ln(Ma_s / 1.5)\}^2] & Ma_s > 1.45 \end{cases}. \quad (2.22)$$

For drag within the rarefaction-dominated regime:

$$C_D = \frac{C_{D,Kn,Re}}{1 + M_s^4} + \frac{M_s^4 C_{D,fm,Re}}{1 + M_s^4}, \quad (2.23)$$

$$C_{D,Kn,Re} = \frac{24}{Re_s} \left(1 + 0.15 Re_s^{0.687}\right) f_{Kn}, \quad (2.24)$$

$$f_{Kn} = \frac{1}{1 + Kn \left[2.514 + 0.8 \exp(-0.55 / Kn)\right]}, \quad (2.25)$$

$$C_{D,fm,Re} = \frac{C_{D,fm}}{1 + \left(\frac{C_{D,fm}}{1.68} - 1\right) \sqrt{\frac{Re_s}{45}}}, \quad (2.26)$$

$$C_{D,fm} = \frac{(1 + 2s^2) \operatorname{erf}(-s^2)}{s^3 \sqrt{\pi}} + \frac{(4s^3 + 4s^2 - 1) \operatorname{erf}(s)}{2s^4} + \frac{2}{3s} \sqrt{\pi}, \quad (2.27)$$

$$s \equiv M_s \sqrt{\frac{\gamma}{2}}. \quad (2.28)$$

The two regimes reach a nexus point of  $Re = 45$ , where the drag coefficient is equal to 1.63, and is independent of Mach number and Knudsen number. This model was created using a wide range of experimental data and proved to better predict tracer response compared to previous models. The model correlated well with experimental data; however, it does not incorporate unsteadiness or nonsphericity effects.

Most recently, Williams et al.<sup>45</sup> suggest that particle response tests done across a strong shockwave tend to overestimate the response of particles to turbulence. It was also shown that rarefaction effects reduce the frequency response of particles in turbulent high-speed flows. Thus, in order to have confidence in tracer selection for supersonic flows, for weakly turbulent flow, corresponding to  $Re_s$  less than

$$Re_{s,\max} = \begin{cases} 8.36Kn + 1.2 & Kn < 0.5 \\ 5.39 / Kn & Kn > 0.5 \end{cases}, \quad (2.29)$$

they recommend iteratively solving for the particle size and density, and then using the Cunningham<sup>43</sup> slip correction,  $C_c$ , to correct the Stokes response frequency:

$$\frac{f}{f_s} = \frac{1}{C_c}, \quad (2.30)$$

$$C_c = 1 + Kn \left( A_1 + A_2 \exp\left(\frac{-A_3}{Kn}\right) \right), \quad (2.31)$$

where the constants are given as  $A_1 = 2.541$ ,  $A_2 = 0.8$ , and  $A_3 = 0.55$ . For strong turbulence, they recommend performing a full simulation using the Loth drag model to obtain the particle drag coefficient. Williams *et al.* mentioned that the maximum slip Mach number that can be encountered by a particle crossing an unsteady shock is 1.89, and thus impose this as an upper limit.

For time-resolved supersonic flows systems, droplets have been generated with diameters between 0.6 and 0.7  $\mu\text{m}$  using an oil based smoke generator, where their highest Stokes number have been estimated *a posteriori* to be 0.05, and were therefore assessed to be sufficiently small to neglect particle lag errors.<sup>22</sup>

Ghaemi *et al.*<sup>46</sup> investigated the use of aluminum nanoparticles with diameters around 10 nm by imaging the tracers in a 3 x 3  $\text{cm}^2$  supersonic wind tunnel. The particles were imaged in an oblique shock wave at Mach 2. The aluminum nanostructured aggregates were generated by the spark discharge mechanism, in which two opposing cylindrical electrodes are located a few millimeters apart. The electrodes were continuously charged, until a breakdown voltage was reached, at which point a spark discharge vaporizes the electrode material. The material rapidly cooled, generating a high concentration of nanoparticles. The velocity normal to the shockwave

was measured using PIV for these nanoparticles as well as commonly used TiO<sub>2</sub> particles with primary diameter of 50 nm. The velocity across a shockwave was located at 0 mm in Figure 2.5. The aluminum nanoparticles reached the downstream velocity after traveling 0.3 mm, while the TiO<sub>2</sub> particles required about 2 mm. The corresponding relaxation times were determined to be 0.27 and 2.0  $\mu$ s for aluminum and TiO<sub>2</sub> particles, respectively.

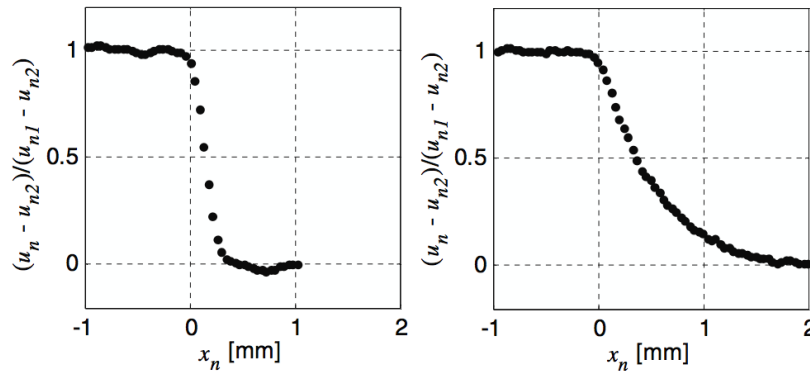


Figure 2.5. PIV measurement of the normal velocity of the aluminum nanoparticles (left) and TiO<sub>2</sub> particles (right) across an oblique shockwave located at 0 mm.<sup>46</sup>

## 2.2 ILLUMINATION

Modern PTV systems, like PIV systems, require a pulsed light source with a short pulse duration and fast pulse frequency. The most commonly used lasers are Nd:YAG, which emit infrared radiation at 1064 nm that are doubled to 532 nm, allowing for pulsed visible illumination. The lasers can typically deliver between 12 and 1000 mJ per pulse at 10 to 30 Hz<sup>19</sup>. The laser pulse frequencies are synchronized with the video camera frame rate to ensure singly-exposed images by exposing the first image of the pair at the end of its frame, and exposing the second image of the pair as the beginning of its frame. This is a process commonly called “frame-straddling”. To ensure this type of asynchronous laser pulsing of the image pairs, PIV/PTV lasers are two Nd:YAG lasers housed together, where their beams are overlapped and combined into a single beam (see Figure 2.6).

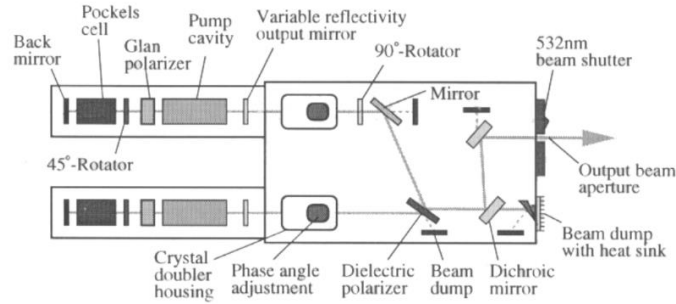


Figure 2.6. Dual-cavity Nd:YAG laser.<sup>47</sup>

Recent technological advances have allowed for the development of the Nd:YLF laser, which outputs for 10-20mJ per pulse at 527nm at rates up to 10kHz. These lasers are also dual-cavity and share the same setup as Nd:YAG lasers shown in Figure 2.6. When combined and synchronized with high-speed cameras, these lasers have allowed for time-resolved (TR) studies, which allow for high temporal resolution data acquisition that capture the time evolution of turbulent flows.<sup>48</sup>

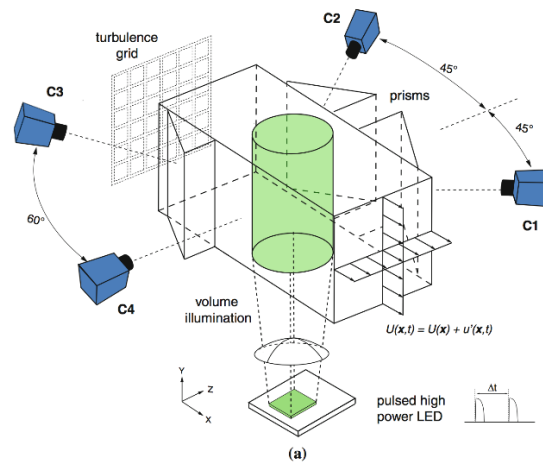


Figure 2.7. LED illumination for tomographic PIV.<sup>52</sup>

An emerging technology for volume illumination is provided by pulsed LED arrays, which provide 1-10 mJ of energy per pulse and pulse rates in excess of 2kHz<sup>49-52</sup>. The benefits of using LED illumination for PTV are that they are a low cost and easy to use alternative to lasers. Additionally, LEDs can be bundled together into compact arrays to increase the illumination volume size or intensity.<sup>52</sup> A 3D PIV setup using LED volume illumination is shown in Figure 2.7.

Also, Figure 2.8 shows an experiment that combines LED and laser illumination for recording of particle images in bubbly flow. The laser illuminated fluorescent tracer particles within a 2D light sheet. Reflected laser light is filtered and the LED backlight created shadow images of the bubbles. This setup resulted in a single camera recording containing both particle and bubble images.

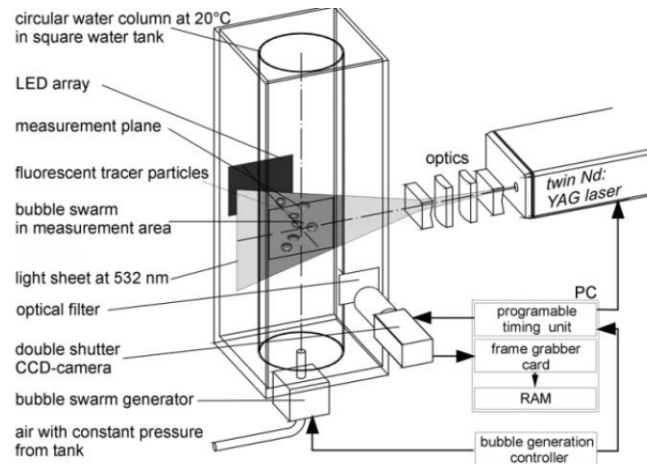


Figure 2.8. Experimental setup using laser and LED illumination for investigation of bubbly flow.<sup>49</sup>

### 2.3 AREA/VOLUME ILLUMINATION OPTICS

The purpose of the laser light source is to illuminate a desired area or volume of interest. A series of lenses are typically arranged to manipulate the light into a sheet or volume. Figure 2.9 shows an example configuration that uses three cylindrical lenses for a laser sheet generation. The first diverging lens stretches the width of the laser sheet. The following converging and diverging lenses are rotated 90 degrees with respect to the first lens, where their separation controls the sheet thickness.

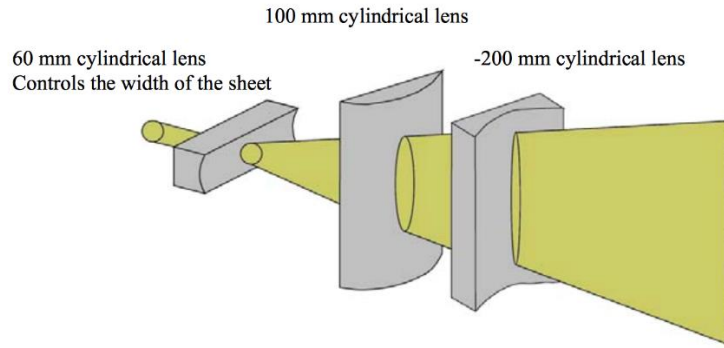


Figure 2.9. Example optics for light sheet generation<sup>53</sup>.

A laser can be manipulated to illuminate a volume in a similar manner shown above. One technique is to use a beam expander with variable magnification. This has been used to illuminate a cylindrical volume with a 30 mm diameter<sup>54,55</sup>. Alternatively, two diverging cylindrical lenses can be oriented perpendicular to one another in order to expand the laser into an ellipsoidal cone<sup>56</sup>. Another common practice in volume illumination is the use of beam blocks to remove the low intensity edges of the light and sharply define the volume thickness.<sup>56,57</sup>

Due to the high energy required to illuminate a volume sufficiently for PTV, a technique of using two dielectric mirrors to reflect light back and forth through the test section can be used. This technique has been used in volume illumination to effectively increase the laser pulse energy from 8 mJ to 60 mJ.<sup>56,57</sup>

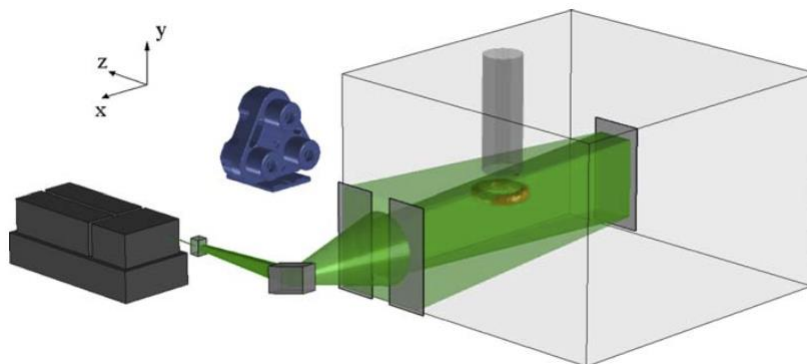


Figure 2.10. The use of dielectric mirrors to increase the light incident on a volume.<sup>56</sup>

## 2.4 CAMERA

When quantitative particle velocimetry methods were initially developed, film cameras were used. Due to their frame rate limitation, frames were typically doubly-exposed<sup>19</sup>. With the advent of technology, however, particle velocimetry image acquisition systems now use CCD or CMOS video cameras<sup>58</sup>. These video cameras have an  $m \times n$  light-sensitive pixel array (available from 1 megapixel array to 50 megapixel array), which collect reflected light from the tracer particles. The CCD video cameras typically incorporate interline frame-transfer chips that allow for inter-frame laser pulse separations as small as 200 ns<sup>19</sup>. The CMOS video cameras incorporate circuitry for each pixel that performs signal processing, which allows for very fast readout. This makes CMOS cameras suitable for high-speed image acquisition as high as 5 kHz. Time-resolved PTV is typically achieved using CMOS cameras and Nd:YLF lasers<sup>19</sup>. An analog-to-digital converter discretizes the analog intensity values read by the camera. A frame grabber that resides on the computer interfaces with the camera to collect data and store it on the computer memory.

Table 2-4: Cameras used in comparison by Hain *et al.*<sup>59</sup> Cameras are labeled with superscripts **a** (CCD), **b** (CMOS), **c** (Intensified CMOS), **d** (identical with LaVision Flowmaster 3s), **e** (CMOS inframe time), and **f** (Identical with IDT XS5-i). fps refers to frames per second, and IF means interframe time.

Camera	Resolution (px <sup>2</sup> )	Pixel size ( $\mu\text{m}^2$ )	fps (1/s)	IF (ns)
PCO Sensicam <sup>a,d</sup>	1,280 × 1,024	6.7 × 6.7	8	300
DantecDynamics NanoSense MKIII <sup>b</sup>	1,280 × 1,024	12 × 12	1,024	100
DantecDynamics iNanoSense MK7 <sup>c,f</sup>	1,280 × 1,024	12 × 12	1,024	100 <sup>e</sup>
PCO 1200hs <sup>b</sup>	1,280 × 1,024	12 × 12	638	75
Photron APX <sup>b</sup>	1,024 × 1,024	17 × 17	2,000	2000
Photron APX-i2 <sup>c</sup>	1,024 × 1,024	17 × 17	2,000	2000 <sup>e</sup>
Redlake HG 100K <sup>b</sup>	1,504 × 1,128	12 × 12	1,000	3000

In using image recording devices, noise is inherent in each pixel. This noise is random and introduces error in particle identification. Three types of error exist in digital recording:

background noise, shot noise, and device noise. Background noise refers to the camera output even in the absence of incident light, typically referred to as “dark current.” It includes both a mean component, which is typically about 30 counts out of 256 for 8-bit cameras, and a random component that is characterized by a Poisson distribution. Shot noise is the random fluctuation inherent to the conversion of light energy to electron flow. It is independent for each pixel and is also characterized by a Poisson distribution. Device noise is caused by a device’s internal electronics, which includes thermal noise, semiconductor surface noise, and switching noise. These sources of noise adhere to a Gaussian distribution. For the purposes of PTV, a camera with high signal-to-noise ratio (SNR) is desirable, and is commonly designed by parametric optimization of these sources of noise<sup>19</sup>.

Hain et al.<sup>59</sup> performed a study of digital cameras to quantify and compare the performance of different digital cameras, which can be used to help select an appropriate sensor for a desired PTV application. The CCD, CMOS, and intensified CMOS cameras used are listed in Table 2-4. Note that the CMOS cameras have pixels four to six times larger and frame rates two to three orders of magnitude higher than the CCD.

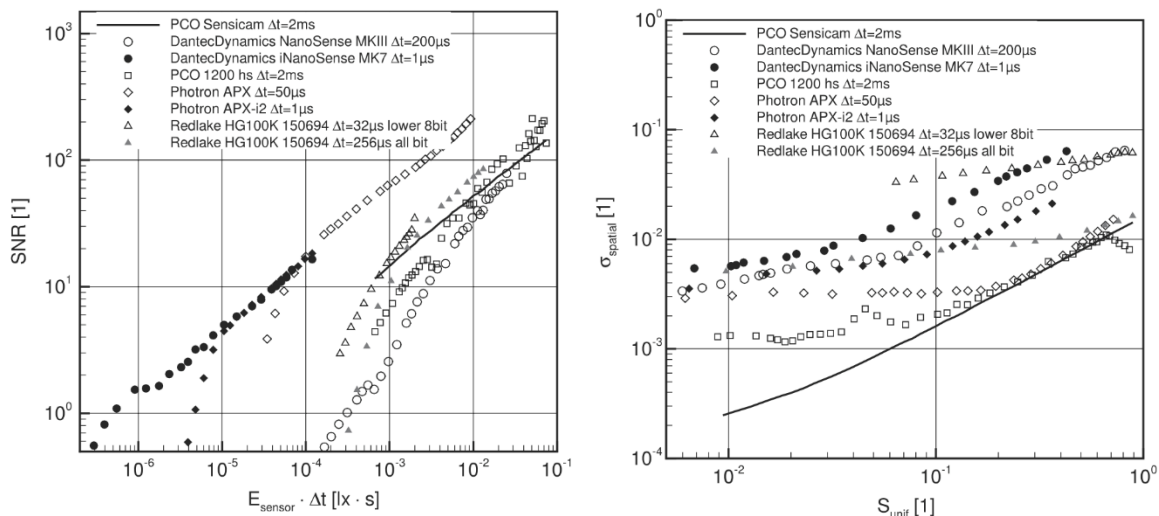


Figure 2.11. Dependence of signal to noise ratio on incident light exposure (left) and spatial pixel variance vs. the uniform signal level (right)<sup>59</sup>.

Figure 2.11a shows how the SNR of the tested cameras varies over a range of light exposure values, measured in lux-seconds. In considering SNR, intensified CMOS cameras (iNanoSense MK7 and Photron APX-i2) perform well with low light intensity, and are recommended when illumination is limited. The CMOS cameras (NanoSense MKIII, PCO 1200hs, Photron APX, and Redlake HG 100K) generally also have a higher SNR for a given light intensity than the CCD (PCO Sencicam) due to their larger pixel size and fill factor. In addition to the testing of SNR, a uniform signal,  $S_{\text{unif}}$ , was generated for each light intensity by temporally averaging 20 images. For each of these averaged uniform signals, the spatial standard deviation of pixel gray values was calculated and is shown in Figure 2.11b as a function of  $S_{\text{unif}}$ . The CCD has the lowest  $\sigma_{\text{spatial}}$ , and therefore superior image uniformity, in all conditions. The quality images can be further improved by CCD cooling. The CMOS cameras perform almost as well as the CCD for highly uniform signals. The intensified CMOS cameras have inferior uniformity for all signal levels. For this reason, Hain recommends that CCDs should be used for the highest precision measurements. CMOS cameras should be used for time-resolved studies, as they are capable of higher frame rates

than CCDs. A CMOS with an intensifier is recommended only when illumination levels are otherwise insufficient.

## Chapter 3. PARTICLE IMAGE IDENTIFICATION

The first step in processing PTV images is identification of particle images for 2D and most 3D experiments. Some 3D techniques, such as tomographic PTV (Section 4.2.2), synthetic aperture PTV (Section 4.2.3), plenoptic imaging (Section 4.2.4), and holographic PTV (Section 4.2.5), perform the particle identification step as part of the volumetric reconstruction step, which will be discussed in their respective sections.

There are two steps in particle identification: first an algorithm must determine which group of pixels constitute an image of a particle; second the location of this particle image must be accurately determined. Accuracy in this step is of the utmost importance, as random errors in image position estimates propagate to both spatial position and velocity estimates. Accurate position estimates are also integral for reliable determination of depth coordinates using a multi-camera setup (Section 4.2).

The ideal raw frame from the CCD would contain bright particle images on a black background. Typically, however, noise exists in the images due to fluctuations in illumination intensity, video camera noise, and particle shadows. Section 3.1 discusses particle identification of these non-overlapped particles, which is applicable to lowly seeded flows. Additionally, some particle images overlap, causing ambiguity as to the number and location of present particles. A number of creative techniques have therefore been developed for extracting particle information and are discussed in Section 3.2, where their development, implementation, and strengths are also described. This is followed by a comparison of published results in Section 3.3.

### 3.1 NON-OVERLAPPED PARTICLES

#### 3.1.1 *Threshold Binarization*

A typical approach for identifying particle images is the use of segmentation. This is done using threshold binarization, which sets a pixel intensity to 1 if the gray level exceeds a set threshold and 0 otherwise<sup>24</sup>. Coherent groups of white pixels are then labeled as separate particle images. While this works for ideal cases, often illumination is inconsistent and bright background shapes can exceed the global threshold.

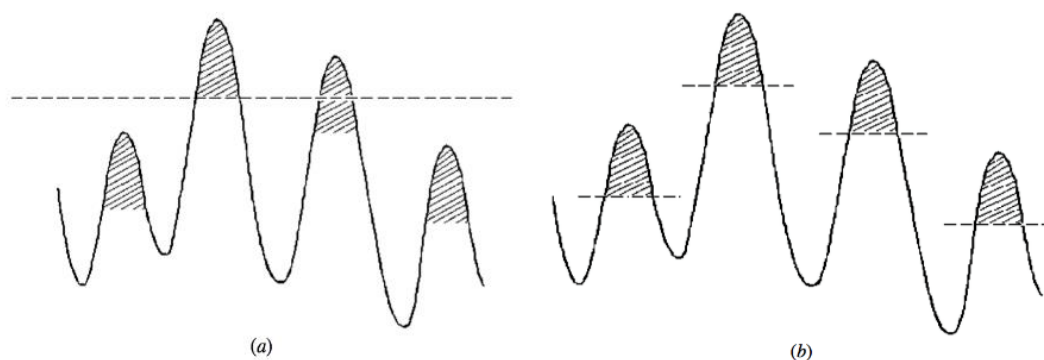


Figure 3.1. Single threshold binarization (a) and dynamic threshold binarization (b).<sup>61</sup>

Thus improvements were made using local threshold binarization<sup>60</sup> and dynamic threshold binarization<sup>61</sup>. Local threshold binarization determines a threshold based on the local average and variance of pixel intensity. Dynamic threshold binarization, on the other hand, uses a global base threshold. Adjacent pixels that exceed the threshold are then labeled together as an image and compared to the threshold value. If the contrast does not exceed a preset threshold, then the procedure for that image is finished; otherwise, one level of brightness is subtracted from each pixel in the area. This is repeated until all images have a contrast within the preset threshold.

Once identified in the CCD frame, an estimate of the particle's location must be found for tracking. This can be calculated as the area centroid of the pixels that make up the particle image.

The binarized image contains separate segments of connected pixels. Each segment is considered a separate particle image, as shown below. Simple threshold binarization algorithms typically will calculate the mean x-coordinate and mean y-coordinate of each particle image to determine its location. The centroid calculations can also be weighted by pixel gray levels.

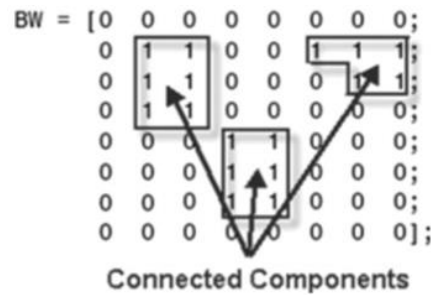


Figure 3.2. Binary matrix showing three separate particle images<sup>62</sup>.

The area centroid estimate is a non-parametric estimator of the particle image location. The estimate is unbiased when particle images are symmetrical and the product of the optical bandwidth and the pixel resolution is less than one. Asymmetrical particle images introduce a systematic error in centroid estimation. Undersampling introduces a centroid estimation bias toward integer pixel values<sup>19,63</sup>.

### 3.1.2 *Gaussian Centroid Estimation*

Once particle image blobs have been separated from the image background, particle positions in the pixel domain need to be determined in order to accurately measure the flow velocities for PTV. Although the particle images are pixelated, techniques for determining particle positions at a sub-pixel accuracy can be used. Initial studies using non-overlapped particle images investigated the parabolic, three-point center-of-mass, five-point center-of-mass, and three-point Gaussian estimators and identified the Gaussian estimator as the best estimator for particle positions.<sup>64</sup> In a later study, the three-point Gaussian (Equation 3.1) and a least-square Gaussian estimator

(Equation 3.2) were studied, concluding that both results were comparable, but recommending the three-point Gaussian estimator as it was 100 times faster.<sup>65</sup> In both of these equations,  $G$  is the array of grayscale pixel intensities,  $x_0$  and  $y_0$  are the coordinates of the brightest pixel of a particle image,  $I_0$  is the intensity, and  $\sigma$  is the standard deviation of the Gaussian spread. These algorithms are based on the assumption that particle images are Gaussian and circular or elliptical and axially oriented, which means the semi-major axis is parallel to the x or y direction. This is maintained for spherical particles with short exposure times compared to particle velocity. In situations with elliptical particle images that are not axially oriented, however, pixel-locking errors occurs.

$$x_c = x_0 + \frac{\ln G(x_0 - 1, y_0) - \ln G(x_0 + 1, y_0)}{2(\ln G(x_0 - 1, y_0) + \ln G(x_0 + 1, y_0) - 2\ln G(x_0, y_0))} \quad (3.1)$$

$$C^2(x_c, y_c, I_0, S) = \sum_{i,j=-1}^1 \left( G(x_i, y_i) - I_0 e^{-\frac{(x_i - x_c)^2 + (y_i - y_c)^2}{2S^2}} \right)^2 \quad (3.2)$$

An improved method presented by Nobach and Honkanen<sup>66</sup> allows for an elliptical particle to be oriented in any direction so long as it maintains a Gaussian shape. The regression is compared visually to a standard three-point estimator in Figure 3.3. The two-dimensional fit uses a least-square fit of a Gaussian as outlined in Equation 3.2. The elliptical Gaussian intensity function can be reorganized and the coefficients can be solved for a least-squares fit, yielding a maximum that would identify the particle position. This algorithm can effectively locate rotated elliptical images, although it cannot handle a background gray value. This means that a pre-processing is required to remove background gray offset. The performance of the two dimensional Gaussian estimator with and without a low pass filter is compared in Figure 3.4. The simulated photon noise is based on the Poisson distribution of the number of photons, where 10,000 photons is the maximum intensity of a particle image.

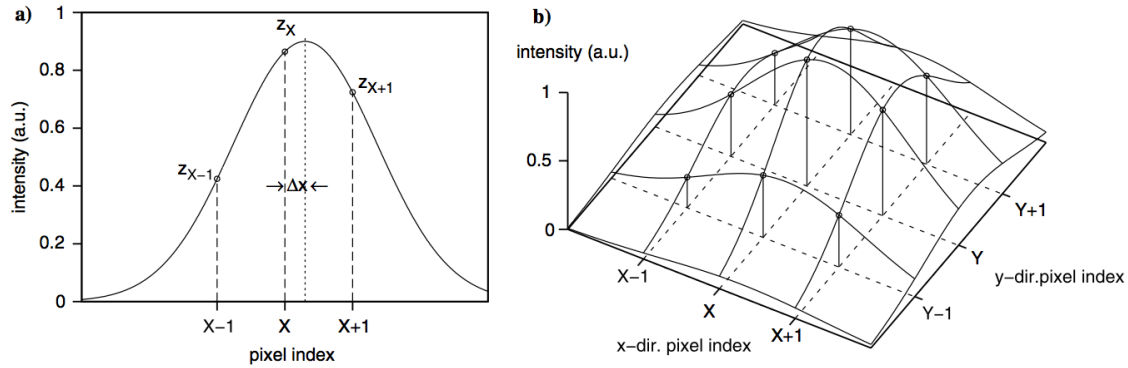


Figure 3.3. One-dimensional three-point Gaussian estimator (a) and two-dimensional Gaussian regression (b).<sup>66</sup>

Both methods have significant systematic deviations for particle diameters below 2 pixels, as the particle images in this range do not match the Gaussian model used. For particle image diameters greater than 3 pixels, and no photon noise, the two algorithms perform comparably. With the introduction of photon noise, bias error increases with image diameter, and the 2-D estimator has less RMS bias than the 2x3 interpolation. For particle image diameters between 2 and 3 pixels, the 9-point regression has substantially higher bias than the 2x3 interpolation. The use of a Gaussian low-pass filter improved the accuracy of the 9-point regression for particle image diameters greater than 2 pixels. The low-pass filter reduces the noise and makes the particle image close resemble the assumed Gaussian shape.

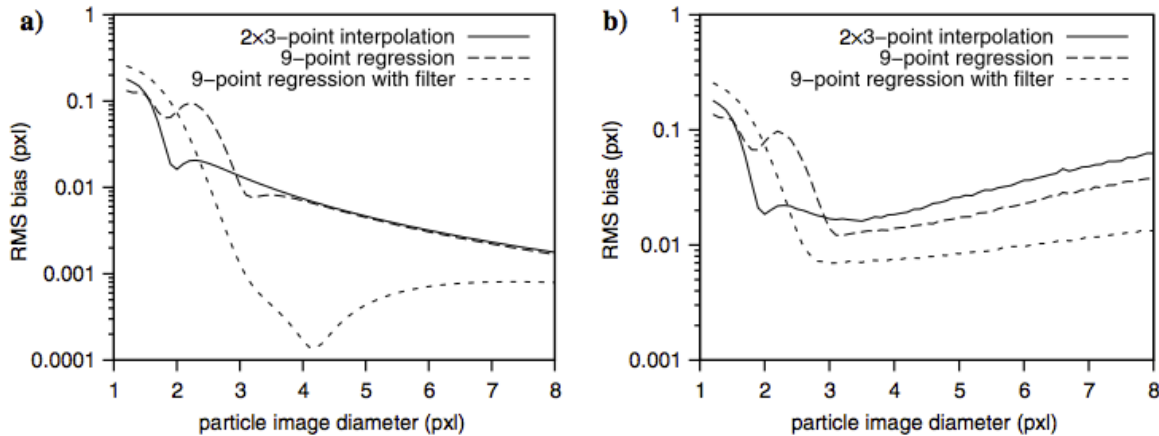


Figure 3.4. RMS error plotted for different particle position estimates a) without noise and b) with simulated photon noise.<sup>66</sup>

## 3.2 OVERLAPPED PARTICLES

In the studies described in the previous section, it was recognized that increasing the particle density resulted in increasing overlapped particles that were incorrectly identified as single particles. This resulted in an incorrect position estimates and thus erroneous results. While prevalent in highly seeded 2D applications, particle overlap is most common in 3D applications. According to Maas *et al.*<sup>24</sup>, for a 512x512 pixel image with 2000 particles (0.0076 particles per pixel or ppp) with each covering 10 pixels on average (or the average particle diameter is 3.6 pixels), the number of overlapped particles increases linearly with the particle image size and with the square of the number particles per image. For this case, 300 cases of overlapping particles were found. Consequently, many early PTV studies worked with sparsely seeded flows in order to avoid this limitation.

### 3.2.1 *Threshold Binarization*

Many techniques have been proposed to identify overlapped particles. Maas *et al.*<sup>24</sup> developed a modified anisotropic thresholding operator that looked for grey value discontinuities within

segmented blobs, and if a discontinuity exceeded a given threshold, the operator split the segmented blobs. While this method was capable of finding multiple particle centroid within an image, the assumption that gray level increases continuously up to local peaks sometimes seems too rigorous when one considers the sensitivity fluctuation of CCD sensors.<sup>61</sup>

Mikheev and Zubtsov<sup>67</sup> modified the dynamic threshold segmentation proposed by Ohmi and Li<sup>61</sup> to detect the inflection point between two neighboring intensity peaks. This allowed them to identify multiple particle positions within segmented blob.

### 3.2.2 *Neural Network Particle Identification*

Carosone *et al.*<sup>68</sup> proposed a neural network for identification of overlapped particle images. Input sensors contain information about each particle image regarding its shape. The inputs interact with a set of neurons with weights for each input, outputting an activation value for each neuron. The neural network weights were taught with a learning data set composed of example images that are properly labeled as an overlapped or a single particle. The neuron with the minimum activation output is then selected and used to label the image. Half of the neurons classify the image as an overlap, while the other half classify the image as a single particle. The authors first defined a circularity measure, *crf*, as  $4\pi$  times the particle's area divided by the particle's squared perimeter, and a convexity measure, *cnv*, as the ratio of the perimeter of the smallest possible convex shape that encloses the image and the actual shape perimeter, where both of these measures are equal to one for a circle. These were two inputs among many other defined parameters, and based on trials using synthetic data, the combination of these two inputs gave the highest percent of average recognition of particle images. Once particle images were labeled as overlapped, a boundary

erosion algorithm was used to separate the individual particles. Once separated, the center of area of each particle image was calculated using the method outlined in the previous section.

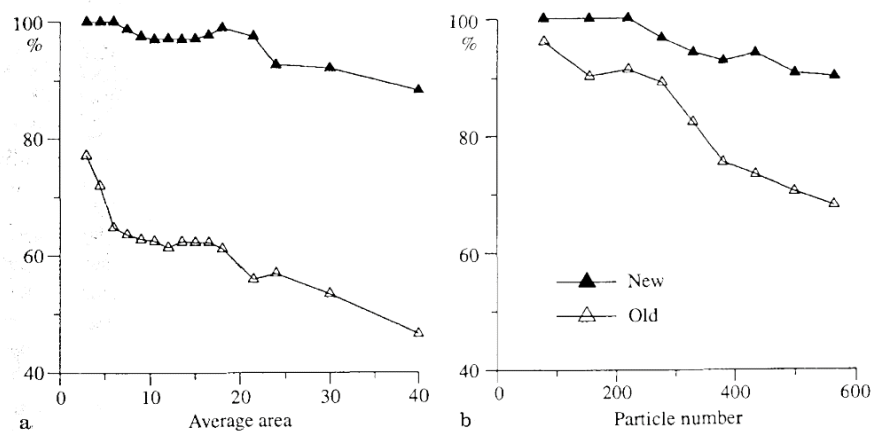


Figure 3.5. Correct barycenter recognition percent vs. (a) average number of pixels per particle and (b) number of particles in a 512x512 pixel image.<sup>68</sup>

Results for the neural network were compared to those of an algorithm that does not separate overlapped particles. The correct barycenter recognition percent was compared for these two algorithms based on synthetic images. Figure 3.5a varies the average number of pixels per particle image with a constant average displacement of 7 pixels and 600 particles (the largest particle concentration studied was 600 particles over  $512^2$  pixel array, or 0.0023 ppp). Figure 3.5b holds the average particle area constant at 12 pixels per particle image (or 4 pixels particle diameter) and varies the particle concentration. The neural recognizer outperforms the previous identification algorithm in all cases, but is not recommended for images with small number of overlaps or average particle areas below 7 pixels (or 3 pixel particle image diameter). Particle images that consist of only a small number of pixels will always be identified as overlapped according to the circularity and convexity measures used. Thus, the algorithm performs boundary erosion on every image and substantially increases computational cost. The increase in performance is minimal

when the number of overlapped particles is small, therefore the algorithm is not recommended due to the computational cost.

### 3.2.3 Particle Mask Correlation

The particle mask correlation algorithm by Takehara and Etoh<sup>69</sup> was introduced as an alternative to threshold binarization. The technique assumes the shape of particle images and searches for areas where the pixel intensity correlates with the assumed shape. Takehara and Etoh suggested that a Gaussian distribution, given in Equation 3.3, best describes particle images, where  $a$  is the peak intensity,  $x_0$  and  $y_0$  give the image centroid coordinates, and  $\sigma$  gives a measure of the size. Equation 3.4 gives the normalized correlation coefficient calculated for an  $m \times n$  pixel window with pixel intensity,  $I$ , mean pixel intensity,  $\hat{I}$ , mask intensity,  $I_m$ , and mean mask intensity,  $\hat{I}_m$ . Normalized correlation coefficients are then be binarized, with an experimentally determined optimal threshold of 0.7. The area is extracted from this binarized image, where after the image's center of gravity and equivalent radius are then calculated.

$$I(x, y) = a * \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}\right) \quad (3.3)$$

$$r(x_0, y_0) = \frac{\sum_{i=x_0-m/2}^{x_0+m/2} \sum_{j=y_0-n/2}^{y_0+n/2} (I(i, j) - \hat{I})(I_m(i, j) - \hat{I}_m)}{\sqrt{\sum_{i=x_0-m/2}^{x_0+m/2} \sum_{j=y_0-n/2}^{y_0+n/2} (I(i, j) - \hat{I})^2} \sqrt{\sum_{i=x_0-m/2}^{x_0+m/2} \sum_{j=y_0-n/2}^{y_0+n/2} (I_m(i, j) - \hat{I}_m)^2}} \quad (3.4)$$

It was determined that the particle mask method is capable of eliminating linear elements, such as tank edges or immersed bodies, within the camera frame. Additionally, for a fixed  $\sigma$  value equal to 3.0 pixels, particle images between  $0.3\sigma$  and  $8\sigma$  were identifiable. For overlapping particles of similar size and brightness, the critical distance,  $L$ , for proper identification was

determined to be  $L = 2\sigma$ . This critical distance increases as difference between the particles' peak brightness increase, up to a maximum of  $L = 4\sigma$  for a brightness ratio of 1/20.

An improvement upon the concept of the particle mask correlation method, called the cascade correlation method, was introduced by Angarita-Jaimes *et al.*<sup>70</sup>. The technique initially performs the same cross-correlation shown in Equation 3.4, labelling this as  $R_1$ . Then the correlation is repeated for  $R_2$  by replacing  $I$  with  $R_1$  in Equation 3.4 and reducing the radius of the mask intensity distribution. Peak pixel intensities in the  $R_2$  coefficient distribution are then found to determine the number and location of each particle. This technique introduces errors in position determination, as locations were calculated from the cross-correlation coefficient distribution rather than the pixel intensity distribution.

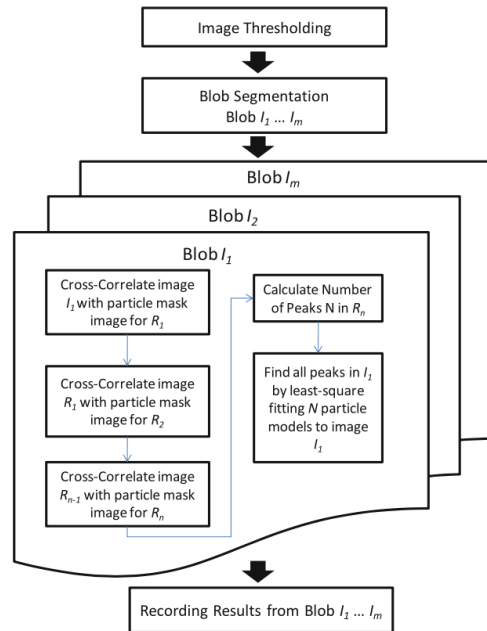


Figure 3.6. Flow chart for the modified cascade correlation particle identification algorithm.<sup>62</sup>

To improve upon the cascade correlation method, Lei *et al.*<sup>62</sup> repeated the correlation process described above, each time with the particle mask radius reduced by one pixel. This iterative process was terminated with the mask's radius reached a user defined minimum value. In addition,

they decided to simply extract the number of peaks,  $N$ , from the cascade correlation method. This information was then be used to perform a least squares fit on the original pixel intensity distribution,

$$C^2 = \sum_{j=1, k=1}^{j=m, k=n} \left[ I(j, k) - \sum_{i=1}^N I_0 e^{\frac{x_i(j, k)^2}{2\sigma_i^2}} \right]^2, \quad (3.5)$$

to determine the particle centroid locations. Figure 3.6 shows the flow diagram for the improved cascade correlation particle identification algorithm.

### 3.2.4 Optical Flow Feature Extraction

Shindler *et al.*<sup>71,72</sup> presented a new algorithm for feature extraction, which would be considered a particle, based in optical flow. The optical flow equation conserves the image intensity of a particle image according to

$$\frac{DI}{Dt} = \frac{\partial I}{\partial t} + u \frac{\partial I}{\partial x} + v \frac{\partial I}{\partial y} = I_t + uI_x + vI_y = \frac{\partial I}{\partial t} + \nabla I^T \cdot \vec{U} = 0; \nabla I(\vec{x}, t) = \begin{bmatrix} I_x \\ I_y \end{bmatrix}, \quad (3.6)$$

where  $\vec{U}$  is the unknown velocity vector. This problem is solved by reformulating it as a minimization in the least-squares sense, leading to the system of equations<sup>73,74</sup>

$$\mathbf{G} \cdot \mathbf{U} + \mathbf{b} = 0; \mathbf{U} = -\mathbf{G}^{-1} \cdot \mathbf{b}, \quad (3.7)$$

where

$$\mathbf{G} = \begin{bmatrix} \int_W I_x^2 dS & \int_W I_x I_y dS \\ \int_W I_x I_y dS & \int_W I_y^2 dS \end{bmatrix}; \mathbf{b} = \begin{bmatrix} \int_W I_x I_t dS \\ \int_W I_y I_t dS \end{bmatrix}. \quad (3.8)$$

The solution requires that the eigenvalues,  $\lambda_1$  and  $\lambda_2$ , of  $\mathbf{G}$  be non-zero, and that they be real and positive. In this regard, a ‘good feature to track’, which would be considered a particle, would need to satisfy

$$\min(\lambda_1, \lambda_2) > \lambda_t, \quad (3.9)$$

where

$$\lambda_t = C \cdot \max(\min(\lambda_1(x, y), \lambda_2(x, y))) \quad (3.10)$$

where  $C$  is a user-defined parameter not larger than unity. The value of  $C$  has been chosen between 0.005 and 0.01 for medium noisy images and should be increased for high levels of noise. A parameter characterizing the minimum distance between features is also given in their algorithms to allow for variations in seeding density, which then define the identified feature is one particle. The barycenter of this particle is then obtained by fitting a Gaussian fit, as given in Equation 3.1, to their feature. In addition to identifying particles, the least squares minimization of Equation 3.7 gives an initial estimate of local velocity.

### 3.3 PARTICLE IDENTIFICATION COMPARISON

#### 3.3.1 *Non-Overlapped Particle Identification Comparison*

Though at present PTV methodologies emphasize the development of algorithms that can detect overlapped particle positions accurately, it is important to evaluate the accuracy of non-overlapped particle position estimators. In doing so, developers will be able to compare the performance of their overlapped particle position estimators with the non-overlapped particle position estimators, as the latter will provide the best estimate to which the former can be measured against.

Marxen *et al.*<sup>65</sup> compared the Gaussian three-point and the Gaussian least-square particle center estimators and quantified the particle positions uncertainties. In Figure 3.7, they show the RMS errors as a function of particle diameter, for a range of Gaussian noise added to their images. They show that the errors are very similar, and that the effect of noise is significant.

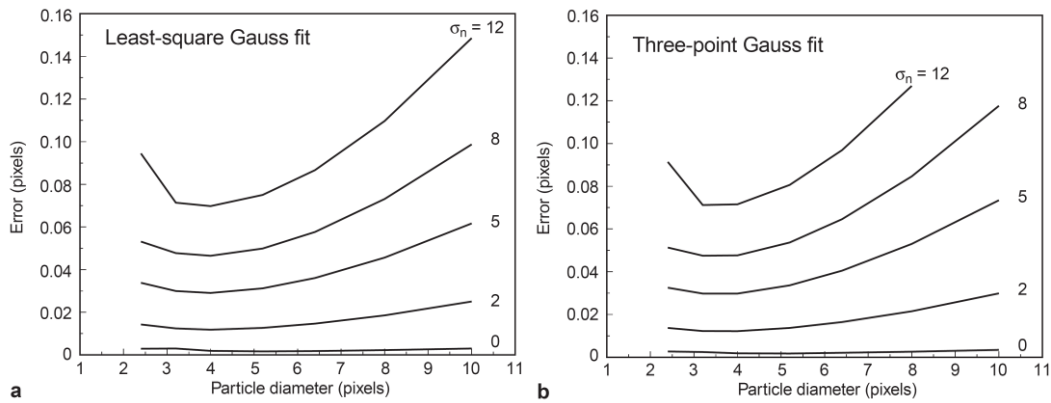


Figure 3.7. RMS error of the center location with respect to the true value of a function of particle diameter and the standard deviation of the Gaussian Noise  $\sigma_n$  (maximum peak intensity =200)<sup>65</sup>

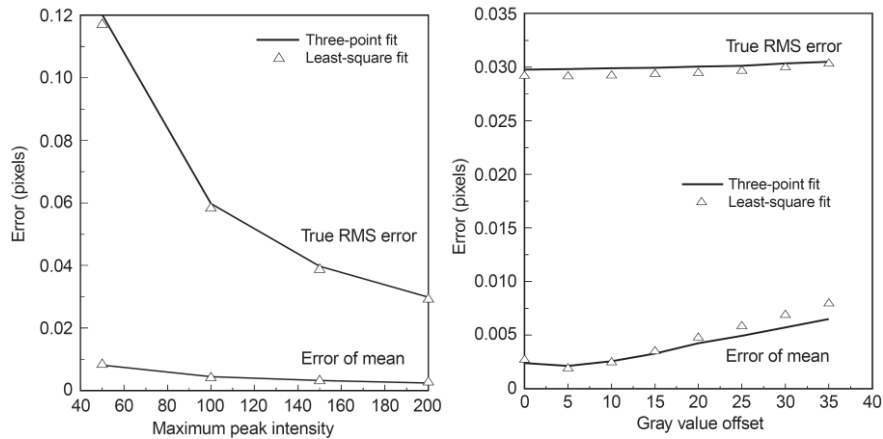


Figure 3.8. (left) RMS error and error of the mean value as a function of the maximum peak intensity ( $\sigma_n = 5$ , particle diameters = 4 pixels); (right) RMS error and the error of the mean value as a function of the maximum peak intensity ( $\sigma_n = 5$ , particle diameters = 4 pixels)<sup>65</sup>

In addition, they also studied the influence of different peak intensities and the offset of the Gaussian gray-value distribution from zero, using images with 4 pixel particle diameters and  $\sigma_n =$

5. Their results, shown in Figure 3.8, show that both the mean and RMS errors have a strong dependence on the peak intensities, while the errors are weakly dependent on the gray offset values. As discussed earlier in Section 3.1.2, the authors conclude that recommended the use of the three-point Gaussian estimator as it was 100 times faster than the least-square Gaussian estimator.

### 3.3.2 *Overlapped Particle Identification Comparison*

Ohmi and Li<sup>61</sup> used the PIV standard image 01 to compare three different particle position estimators, the results of which are shown in Table 3-1. Here, the number of particles identified in the PIV standard image 01 for each method is listed<sup>75</sup>. In these results, it is shown that the dynamic threshold binarization algorithm was able to identify more particles, though uncertainties in particle positions were not reported and compared. In their modified dynamic threshold binarization algorithm, the mean error and standard deviation of the particle positions, though not measured directly, were estimated to be 0.05 and 0.1 pixels, respectively. It should also be noted that in a study performed by Cardwell *et al.*,<sup>76</sup> the authors found that the modified dynamic threshold binarization method did not perform robustly in identifying overlapped particles as compared with the particle mask method.

The improvements of the cascade correlation method over the particle mask correlation method is shown in Figure 3.9, where a reduction of ~70% can be seen at maximum overlap. The improvements to the cascade correlation method made by Lei *et al.*<sup>62</sup> are shown in Figure 3.10 and Figure 3.11.. Here, Figure 3.10 shows the effect of separation distances of two identical particles with a particle diameter of 20 pixels, where the particle diameter profile is Gaussian, and is defined as 4 times the standard deviation of its Gaussian profile. Figure 3.10(a) shows the results for

Poisson noise, where in comparison to the cascade correlation method, their algorithm results in particle location errors of 0.06 and 0.16 pixels for SNR of 10 and 25, respectively, for maximum overlapped particles. Similarly, Figure 3.10(b) shows that their algorithm for data with Gaussian noise, also for maximum overlapped particles, results in errors of 0.07 and 0.25 pixels.

Table 3-1: Comparison of identification algorithms for the PIV standard image 01.<sup>61</sup>

Particle identification algorithm	Number of particles identified/total particles	Identified ppp/total ppp
Single Threshold Binarization	1160/4000	0.018/0.06
Dynamic Threshold Binarization	1269/4000	0.019/0.06
Particle mask correlation	1134/4000	0.017/0.06
Optical Flow Feature Extraction	1330/4000	0.020/0.06

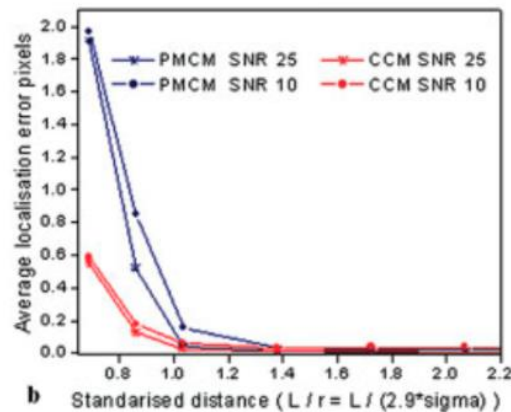


Figure 3.9. Comparison of the cascade correlation method with the particle mask correlation method.<sup>70</sup>

The effect of particle size and Gaussian noise is shown in Figure 3.11.. Here, particle diameters were varied from 4 to 20 pixels. For a Gaussian noise level of 2.5%, Figure 3.11.(a) shows that the algorithm requires the particle diameters to be greater than 10 pixels in order to resolve overlaps up to 50%, since particles with smaller diameters do not provide enough spatial distribution and hence resolution to be able to accurately resolve overlaps of up to 50%. For a Gaussian noise level of 5%, Figure 3.11.(b) shows similar trends, however, the errors roughly

double due to the doubled noise level. What is also seen is that larger particles result in higher particle location errors. For example, 6 pixel diameter particles have location errors of 0.03 and 0.06 for Gaussian noise levels of 2.5% and 5%, respectively, while 12 pixel diameter particles have location errors of 0.06 and 0.13 for the same Gaussian noise levels. For the 6 pixel diameter particles, these are 95% and 90% smaller than the Cascade correlation method for a Gaussian noise level of 2.5%, and 90% and 78% smaller than the Cascade correlation method for a Gaussian noise level of 5%.

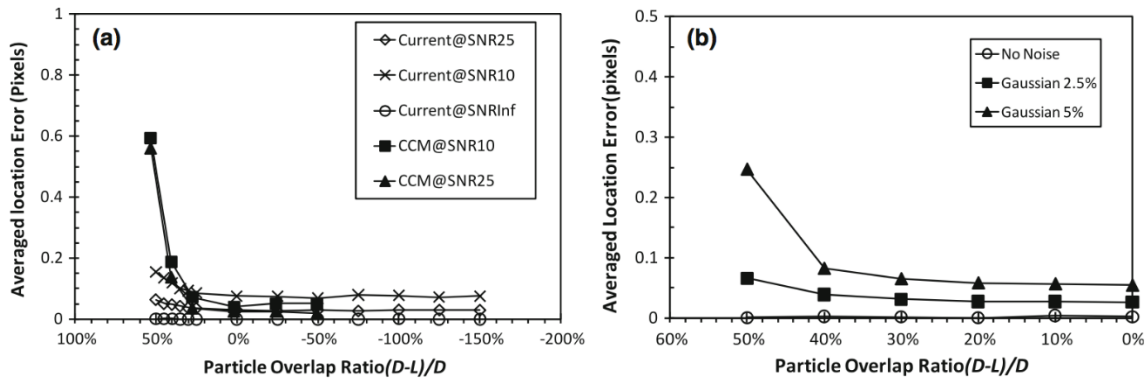


Figure 3.10. Averaged particle location error versus particle overlap ratio for a) Poisson noise levels at infinity, 25, and 10. B) Gaussian noise levels at 2.5% and 5%.<sup>62</sup>

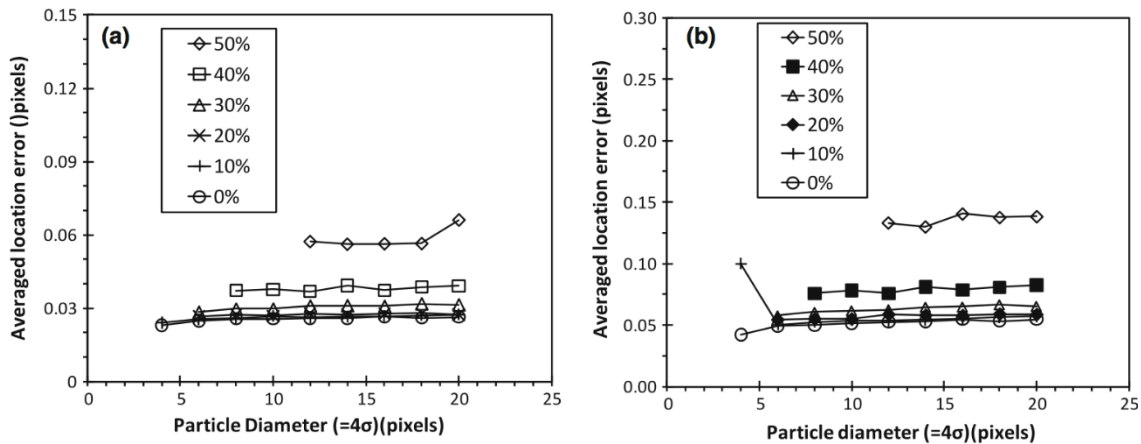


Figure 3.11. Averaged particle location error versus particle diameter for different overlap ratios a) 2.5% Gaussian noise b) 5% Gaussian noise.<sup>62</sup>

Recognizing that in practice, seeding particles will vary in size, Lei *et al.*<sup>62</sup> also did similar studies with overlaps of particles with varying sizes at different Gaussian noise levels (see Figure 3.12). For this study, their particle diameter was defined as the average of the two overlapped particles, where the smaller particle was set to 4 pixels in diameter. The diameter ratio was varied from 1 to 5. For 2.5% Gaussian noise, Figure 3.12 (a) shows that the curves overlap; however, only for diameter ratios of 4 and 5, the curves extend out to resolvable overlaps of up to 60%, with errors under 0.05 pixels. Figure 3.12 (b) shows similar trends for 5% Gaussian noise, but with increased noise that is below 0.1 pixels.

It is important to point out that Lei *et al.*<sup>62</sup> recognized that the VSJ overlapped particle images<sup>75</sup> used only the intensities of the brightest particle when forming the overlapped particle images. They showed that this was not the correct representation of overlapped particles, rather the correct representation of overlapped particles should sum all the intensities for each pixel from each of the overlapped particles, as is shown in Figure 3.13. Here, it can be clearly seen that the experimental data and the superposition of the particles are much closer in profile than the method used to calculate the VSJ images. As a consequence, the VSJ images do not capture overlapped particle occlusions well, and therefore the use of these images will artificially result in lower particle position uncertainty estimates.

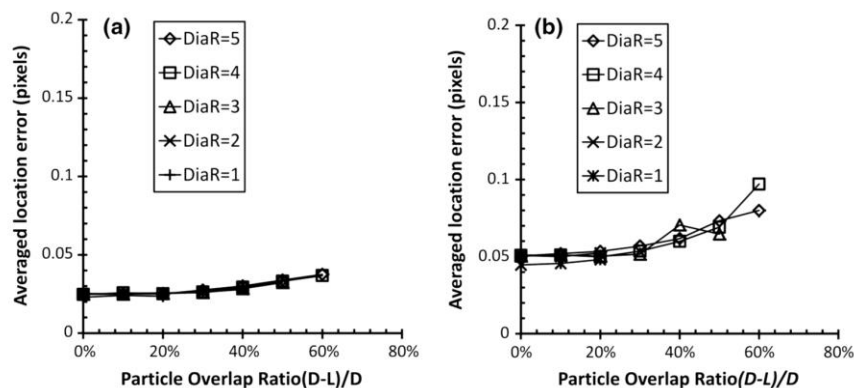


Figure 3.12. Averaged particle location error with various particle diameter ratios and noise levels. (a) 2.5% Gaussian noise versus particle overlaps. (b) 5% Gaussian noise versus particle overlap ratio.<sup>62</sup>

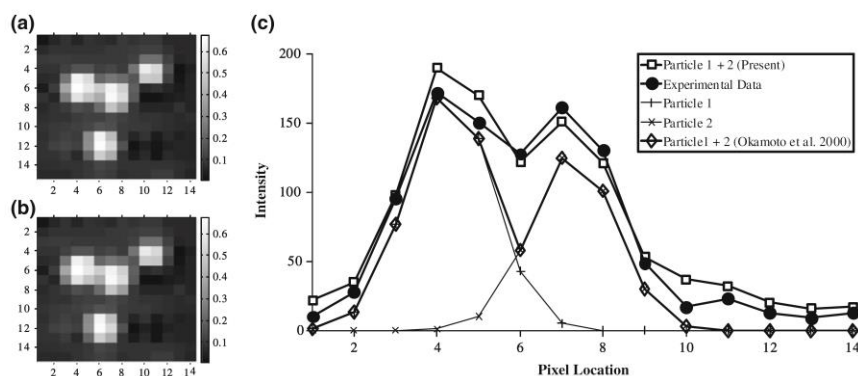


Figure 3.13. Comparison of particle image reconstruction performance based on different assumptions. **a** Original image, **b** reconstructed image by current work, **c** comparison of profiles of the 6th row.<sup>62</sup>

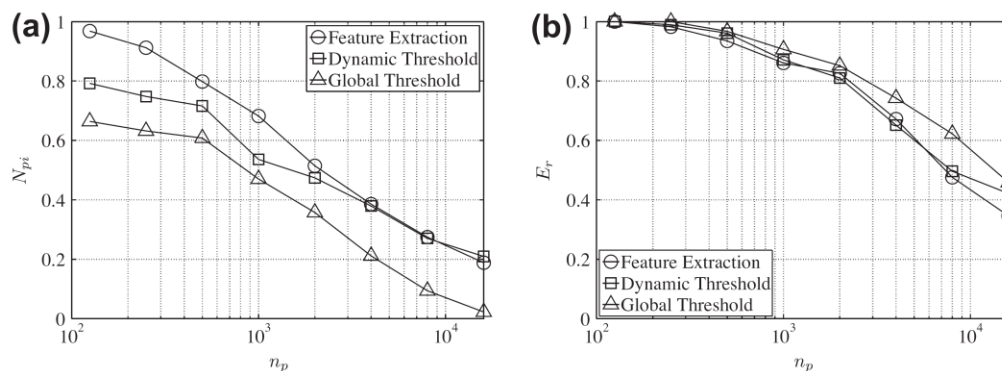


Figure 3.14. Tracer density influence on (a) the normalized number of identified particles,  $N_{pi}$ , and (b) the reliability,  $E_r$ .<sup>72</sup>

The optical flow feature extraction (OFFE) technique as also compared with the dynamic threshold (DT) and global threshold (GT) methods.<sup>72</sup> Figure 3.14 shows the effect of particle density on the normalized particle density (identified particles per total particles,  $N_{pi}$ ) and on the reliability,  $E_r$ , which is defined as the number of correctly identified particles divided by the total number of identified particles. For these tests, the number of particles were varies from 125 to 16,000 (labeled  $n_p$ ) within a 256x256 pixel image size, corresponding to a range of 0.002 to 0.244 ppp, with a 3 pixel mean particle diameter and a background mean and standard deviation Gaussian noise of 10 and 1, respectively. Figure 3.14a shows that the OFFE outperforms the GT by about 0.2  $N_{pi}$  throughout the range of particle densities tested in identifying particles. At 0.002, .0038, 0.0076, 0.015, 0.03 and 0.06 ppp, the  $N_{pi}$  are about 0.975, 0.91, 0.8, 0.68 and 0.51 respectively. Beyond 0.06 ppp, the performances become identical. Figure 3.14b shows that of these identified particles, the reliability of OFFE is comparable to that of DT up to 0.12 ppp.

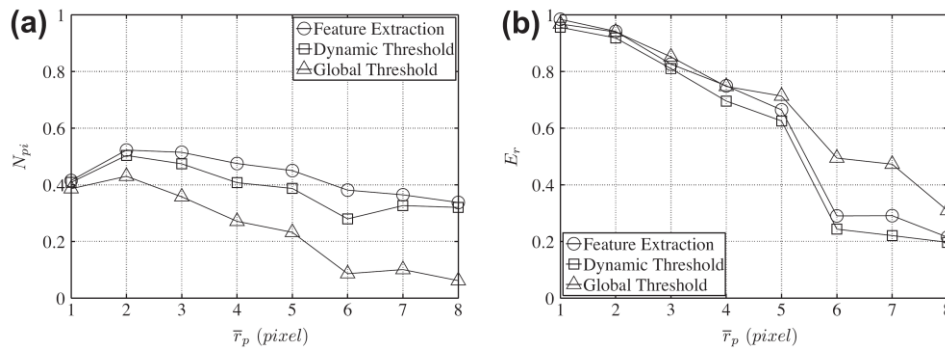


Figure 3.15. Particle radius influence on (a) the normalized number of identified particles,  $N_{pi}$ , and (b) the reliability,  $E_r$ .<sup>72</sup>

Figure 3.15 shows the effect of particle image radius on  $N_{pi}$  and  $E_r$ . For these tests, the mean particle image radius varied from 1 to 8, with 2000 particles per image, or 0.03 ppp, and a mean and standard deviation in the background Gaussian noise of 10 and 1, respectively. Figure 3.15a again shows the OFFE outperforms the DT and GT, and that the OFFE's performance is mostly insensitive to the particle image radius. Of these identified particles, all methods show a reliability

that is mostly similar up to a particle radius of 5 pixels (see Figure 3.15b), after which the GT noticeably outperforms the DT and OFFE.

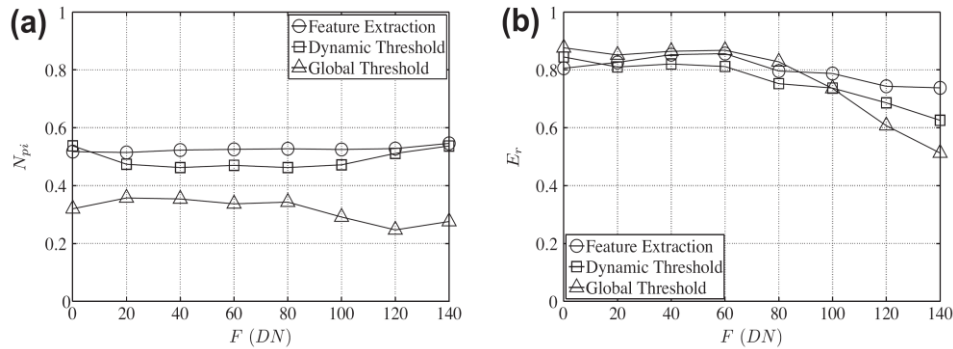


Figure 3.16. Background mean noise influence on (a) the normalized number of identified particles,  $N_{pi}$ , and (b) the reliability,  $E_r$ .<sup>72</sup>

Figure 3.16 shows the effect of background mean noise on  $N_{pi}$  and  $E_r$ . For these tests, the mean particle image radius was set to 3, with 2000 particles per image, or 0.03 ppp. The mean background noise is labelled as  $F$ , and the standard deviation of the noise was set to 1. Figure 3.16a shows that all three algorithms' performance is mostly uniform across the noise span tested, where the OFFE shows the best performance. Figure 3.16b shows that even though the signal-to-noise decreases as the noise level goes up, that all three algorithms' performance degrades slowly, as the reliability gently decreases with increasing noise, where the OFFE outperforms the DT and the GT.

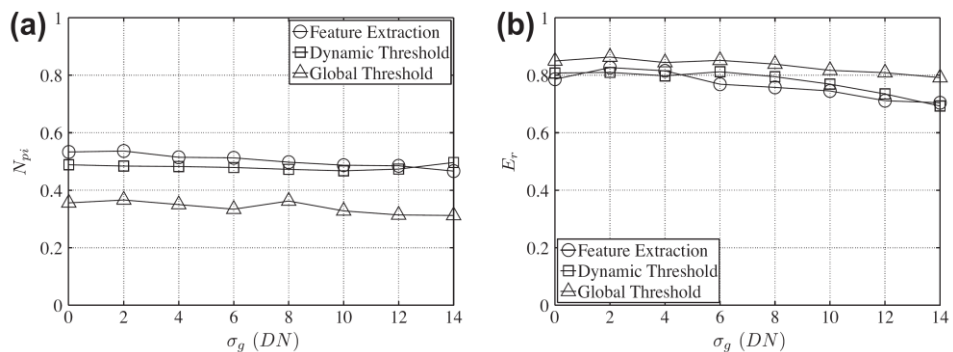


Figure 3.17. Background standard deviation noise influence on (a) the normalized number of identified particles,  $N_{pi}$ , and (b) the reliability,  $E_r$ .<sup>72</sup>

Figure 3.17 shows the effect of background standard deviation noise on  $N_{pi}$  and  $E_r$ . Similar to previous tests, the mean particle image radius was set to 3, with 2000 particles per image, or 0.03 ppp. The background standard deviation noise is labelled as  $\sigma_g$ . Also, the coefficient  $C$  in Equation 3.10 was increased for the noisiest images in order to prevent false detections. Figure 3.17a shows that all three algorithms' performance is mostly uniform across the noise span tested, where the OFFE shows the best performance of the three methods tested. Figure 3.17b shows that for increasing standard deviation of the noise, the reliability of all the tested algorithms is minimally affected, and the GT outperforms OFFE and DT.

## Chapter 4. IDENTIFICATION OF PARTICLES' SPATIAL LOCATIONS

While tomographic (Section 4.2.2), synthetic aperture (Section 4.2.3), plenoptic imaging (Section 4.2.4), and holographic (Section 4.2.5) methods directly identify particle spatial locations from reconstructed volumes, most other PTV algorithms use particle positions within the image space to determine the corresponding positions in space. This then allows for tracking of particle tracers that will result in velocity vector fields in terms of real-world coordinates. The technique for identifying particle spatial locations varies between 2D and 3D methods. Section 4.1 therefore discusses 2D spatial localizations, while Section 4.2 discusses 3D spatial localizations. Most of these transformations from the camera domain to the spatial domain require a calibration as well as methods for handling optical distortions, therefore calibration methods are discussed in Section 4.3.

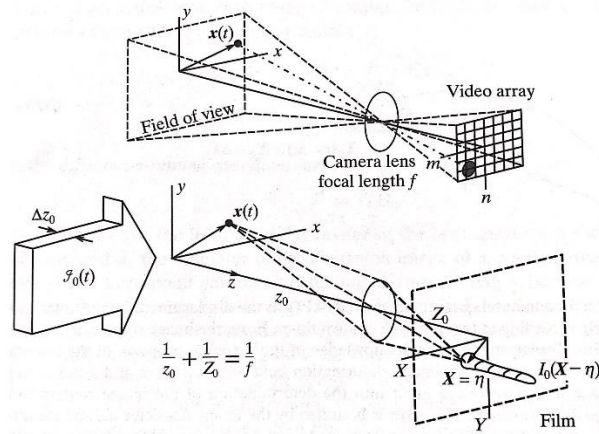


Figure 4.1. The geometry used in the pinhole model for determining 2D spatial coordinates of particles.<sup>4</sup>

#### 4.1 SPATIAL LOCALIZATION IN 2D

When only two-dimensional position and velocity are of interest, geometry can be used to determine the spatial coordinates from the camera plane. Assuming the pinhole camera model<sup>19</sup>, Equation 4.1 can be used to transform the image coordinates,  $X$  and  $Y$ , to the spatial coordinates,  $x$  and  $y$ . In this case,  $Z_0$  is the distance from the image to the effective center of the lens and  $z_0$  is the distance from the object to the effective center of the lens. This geometry is shown in Figure 4.1.

$$\begin{pmatrix} X \\ Y \end{pmatrix} = \frac{Z_0}{z_0} \begin{pmatrix} x \\ y \end{pmatrix} \quad (4.1)$$

The pinhole model works for 2D imaging without any optical distortion or out-of-plane measurements. When out-of-plane motion exists, the following perspective error, given by Prasad,<sup>77</sup> arises in the measurement of  $x$  displacement:

$$\varepsilon_x = \frac{\Delta z}{\Delta x} \sin \phi, \quad (4.2)$$

where  $\Delta z$  is the out-of-plane displacement,  $\Delta x$  is the in-plane displacement, and  $\phi$  is the viewing angle with respect to the optical axis. The uncertainties in 2D localization come only from errors

in the image identification, calibration, and perspective. The calibration steps in section 4.3 can be used to address these uncertainties.

## 4.2 SPATIAL LOCALIZATION IN 3D

The localization of particle coordinates in three dimensions can be a complex task. In order to extract three components of both position and velocity vectors, one of the methods outlined in this section is typically used. Rather than using only one camera, these 3D techniques incorporate multiple cameras and illuminate a volume rather a sheet. Due to this complexity, calibration steps are typically required in order to ensure proper localization.

### 4.2.1 *Photogrammetric PTV*

Photogrammetry is the science of making measurements from photographs. The underlying principle is that distance between two points in an image can be determined if the scale is known. In the context of PTV, this scale depends on the depth of a particle image. This section outlines techniques that have been used to estimate depth and obtain 3D coordinates of imaged particles.

#### 4.2.1.1 Multi-Camera Setup

Most photogrammetric PTV techniques rely on multiple viewing angles in order to determine 3D coordinates. For extremely low particle densities, two cameras sufficiently provide all three spatial dimensions; however, a third or fourth camera can remove ghosts that arise when particle images overlap, as well as reduce uncertainties in the measurements. It should be also noted that stereoscopic flow imaging (two cameras) is most often used to obtain data 3-component velocities within a two-dimensional plane. Figure 4.2 shows an experimental setup with two cameras for stereoscopic PTV and three cameras and volume illumination for 3D PTV.

The chosen viewing angles affect both the presence of ghost particles and the error associated with the 3D coordinate estimate. Graff and Gharib<sup>78</sup> performed a generalized analysis to assess the error in the reconstructed scene (reconstruction precision) as well as the ghost particle generations (reconstruction quality) of various camera arrangements. Table 4-1 shows the error factor,  $E$ , defined as

$$E = \sqrt{\sum_{i=1}^N \left[ \left( \frac{\partial \bar{b}}{\partial x_i} \right)^2 + \left( \frac{\partial \bar{b}}{\partial y_i} \right)^2 \right]}, \quad (4.3)$$

for  $N$  cameras with a mean image separation,  $\bar{b}$ , associated with various setups for in-plane measurements of position, assuming that each camera has the same parameters and uncorrelated error.

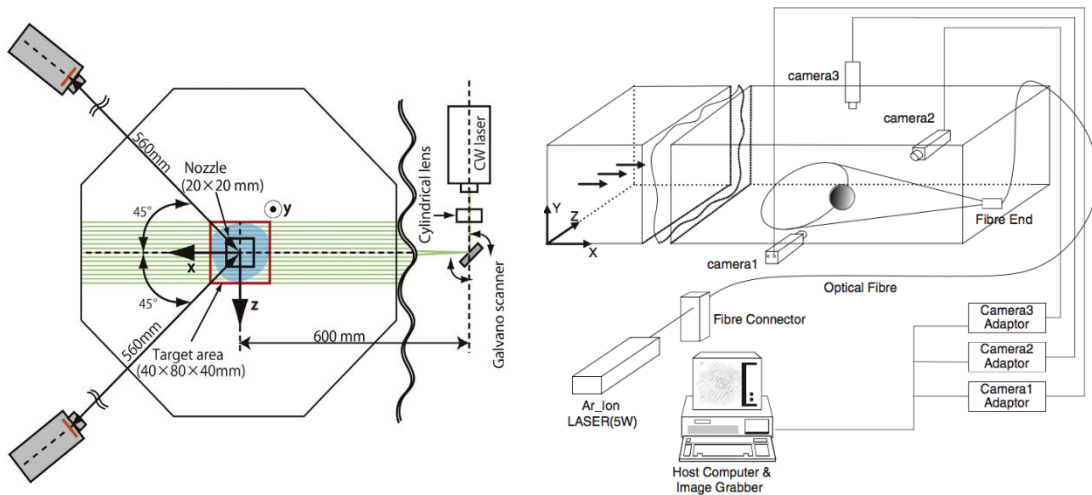
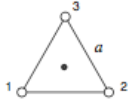
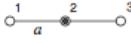
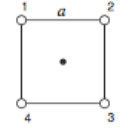
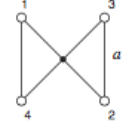
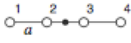
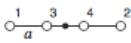


Figure 4.2. Setup for Stereoscopic PIV used to investigate a jet flow by Watanabe et al. (left)<sup>79</sup> 3D PTV setup used to investigate the wake of a sphere by Doh et al. (right)<sup>80</sup>

Table 4-1: Camera setups for 3D PTV, their respective error factors,  $E$ , and sensitivity coefficients,  $B$ , for in-plane measurements.  $\Delta c$  is the horizontal distance between cameras and  $\Delta d$  is the vertical distance between cameras.<sup>78</sup>

Setup	$\Delta c$	$\Delta d$	$\bar{B}/M$	$E$
	$-a, a/2, a/2$	$0, -a\sqrt{3}/2, a\sqrt{3}/2$	$a$	1
	$a, a, 2a$	$0, 0, 0$	$4a/3$	$2\sqrt{2}/3 \approx 0.94$
	$-a, 0, a, 0$	$0, a, 0, -a$	$a$	$1/\sqrt{2} \approx 0.71$
	$-a, 0, a, 0$	$a, -a, a, -a$	$(1 + \sqrt{2})a/2$	$\approx 0.92$
	$-a, -a, -a, 3a$	$0, 0, 0, 0$	$3a/2$	$1/\sqrt{2} \approx 0.71$
	$-3a, 2a, -a, 2a$	$0, 0, 0, 0$	$2a$	1

The lowest  $E$  for a three-camera system is  $\sim 0.94$ , while that of a four-camera system is  $\sim 0.71$ , showing that a proper four-camera arrangement can perform better than a three-camera system. They also show that random ghost particles, which occur due to random chance that the points distribution generates false matches within the matching tolerance, are decreased with increasing cameras. However, clumped ghost particles, which occur when one or more point images are within a matching tolerance of another in a single camera image, increase rapidly with increasing cameras, though they are more readily identifiable and easy to remove.

An alternative, cost effective solution for a 3D PTV experimental setup was developed by Kreizer and Liberzon<sup>81</sup>. The basic concept is to use a single camera with an image splitter and mirrors to obtain four different viewing angles of the observation volume. Figure 4.3 shows an example of this setup. The authors also developed an open source MATLAB toolbox to assist in the development of a similar system based on optical parameters.

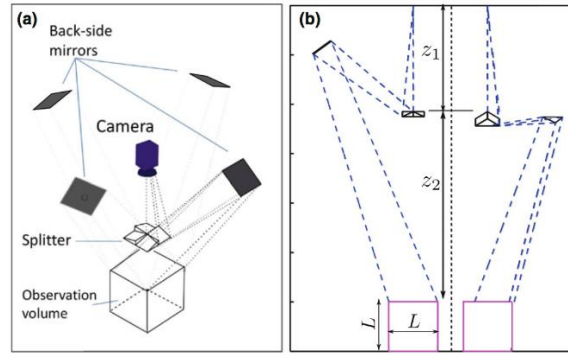


Figure 4.3. Experimental setup of the single camera 3D PTV system.<sup>81</sup>

Accuracy of this experimental setup, when calibrated properly, is the same as multi-camera 3D PTV techniques. The drawback of this technique, however, is that each viewing angle uses one quarter of the camera's resolution.

#### 4.2.1.2 Triangulation Methods

Just as in 2D PTV, particle images are identified in the image plane of each camera individually, as discussed in Chapter 3. From these 2D images, the 3D position of each object is reconstructed using triangulation, by first creating epipolar lines based on a pinhole camera model, then using their intersections to triangulate their 3D particle locations. This approach can be used for both stereoscopic (2 camera), or multi-camera (3 or more cameras) PTV systems. Figure 4.4 shows how triangulation works using epipolar lines for three cameras.  $E_{1-2}$  is the epipolar line created for pixel  $P'$  from image 1 into image 2. In this case, two cameras would not suffice for particle location, as there are three particle images in image 2 that lie on  $E_{1-2}$ . Thus the third camera is used, which would allow for generating epipolar lines,  $E_{(2-3)i}$ , associated with the 3 identified particles in image 2. It would also allow for generating an epipolar of  $P'$  from image 1 to image 3,  $E_{1-3}$ , the intersection of which would identify the correct triplets from which the particle's 3D spatial location can be determined. In reality, due to measurement uncertainties, a tolerance must be added about the epipolar lines to identify triplets within these uncertainties.

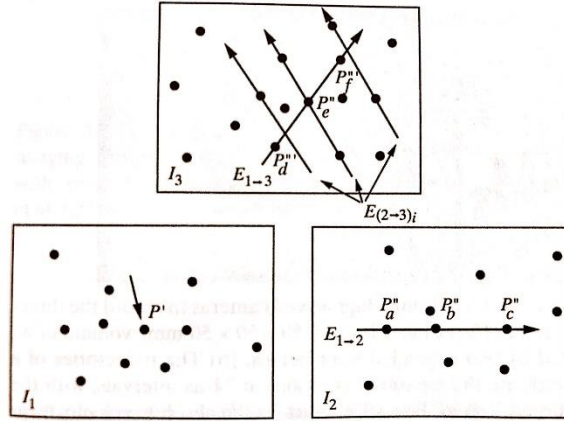


Figure 4.4. Principle of intersection of epipolar lines for photogrammetric PTV systems.<sup>24</sup>

The mathematical formulation of photogrammetric PTV without any distortion is

$$x'_i = x_h - c \frac{r_{11}(X_i - X_0) + r_{21}(Y_i - Y_0) + r_{31}(Z_i - Z_0)}{r_{13}(X_i - X_0) + r_{23}(Y_i - Y_0) + r_{33}(Z_i - Z_0)} \quad (4.4)$$

$$y'_i = y_h - c \frac{r_{12}(X_i - X_0) + r_{22}(Y_i - Y_0) + r_{32}(Z_i - Z_0)}{r_{13}(X_i - X_0) + r_{23}(Y_i - Y_0) + r_{33}(Z_i - Z_0)} \quad (4.5)$$

In this case  $x'_i$  and  $y'_i$  are the image plane coordinates of a particle image and  $x_h$  and  $y_h$  are the coordinates of the center point in the image plane. The spatial coordinates are  $X_i$ ,  $Y_i$ , and  $Z_i$ .  $X_0$ ,  $Y_0$ , and  $Z_0$  are the coordinates of the camera focal point and  $r$  is a 3x3 rotation matrix that captures the camera's external orientation.

To correct for lens distortion, Murai *et al.*<sup>82</sup>, Nishina *et al.*<sup>83</sup> have suggested radial distortion corrections, however, Maas<sup>24</sup> has also included tangential distortion<sup>84</sup> corrections by modifying Equations 4.4 and 4.5 as

$$\bar{x}_i = x'_i + dx_i, \bar{y}_i = y'_i + dy_i, \quad (4.6)$$

$$dx_i = x'_i (k_1 r_i'^2 + k_2 r_i'^4 + k_3 r_i'^6) + p_1 (r_i'^2 + 2x_i'^2) + 2p_2 x'_i y'_i, \quad (4.7)$$

$$dy_i = y'_i (k_1 r_i'^2 + k_2 r_i'^4 + k_3 r_i'^6) + p_1 (r_i'^2 + 2y_i'^2) + 2p_2 x'_i y'_i, \quad (4.8)$$

$$r_i'^2 = x_i'^2 + y_i'^2, \quad (4.9)$$

where  $k_1, k_2, k_3$  are the radial distortions, and  $p_1, p_2$  are the tangential distortions. Maas also proposed compensating for the effects of image digitization and storage using additional affine transformations as proposed by El-Hakim<sup>85</sup> and Albertz and Kreiling.<sup>86</sup> Calibration can be used to obtain these coefficients are further discussed in Section 4.3.3.

Maas *et al.*<sup>24</sup> found that their implementation of Photogrammetric PTV tracked on the order of  $10^3$  tracer particles before ghost particles begin to appear. These particles were imaged within a  $200 \times 160 \times 50$  mm<sup>3</sup> measurement volume, resulting in 0.0038 ppp. Particle positions were estimated to have errors of 60  $\mu$ m in plane and 180  $\mu$ m out-of-plane. The rms random errors for displacement were significantly smaller than for position, as the systematic errors in position estimates do not affect accuracy of displacement estimates.

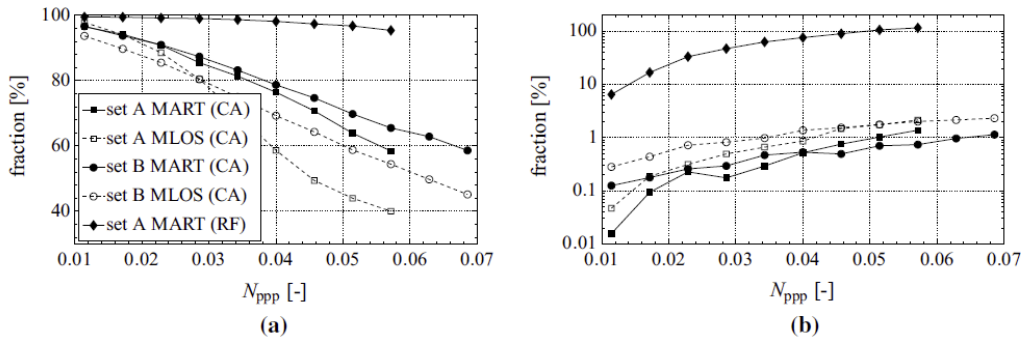


Figure 4.5. (a) The fraction of correctly estimated particle locations (within one voxel radius from the true particle location) and (b) the fraction of ghost particles compared to the number of true particles<sup>87</sup>.

One method for increasing the particle seeding density, previously discussed in Section 3.2.3, was developed by Lei *et al.*<sup>62</sup>, who developed a method to identify overlapped particles. Another algorithm was developed by Fuchs *et al.*<sup>87</sup> The first step of the algorithm reconstructs the volume using the tomographic multiplicative algebraic reconstruction technique (MART), described in

Section 4.2.2. The reconstructed volume is then binarized and the image particle locations are estimated from their center of mass. These images are then mapped back onto each camera sensor. They are then compared to the actual particle images. The true particle images first must be identified and located within each camera sensor using a 2D Gaussian fit. Matching is done between the reconstructed and true particle images. If a particle image is associated with multiple reconstructed images, it is disregarded. This reduces the effect of ghost particles. Triangulation is then performed for particle location if uniquely matching particle images exist on at least two sensors.

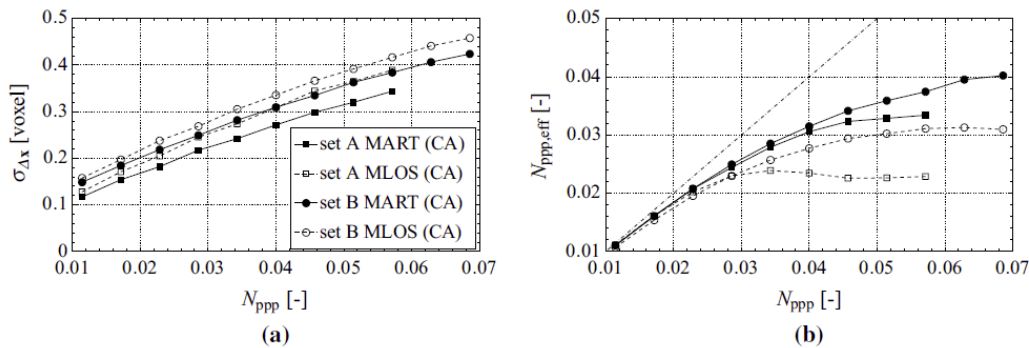


Figure 4.6. (a) Spatial particle location uncertainty and (b) the effective particles per pixel.<sup>87</sup>

The results of this reconstruction technique (CA) are compared to a standard tomographic reconstruction (RF) for two volume thicknesses (see Figure 4.5). Set A has a volume size of  $800 \times 800 \times 300$  voxel<sup>3</sup> and set B has a volume size of  $800 \times 800 \times 50$  voxel<sup>3</sup>. For example, at 0.05 ppp, the tomographic reconstruction (RF) shows near 100% correct particle identification, but also 100% generation of ghost particles. At 0.05 ppp, the proposed algorithm (CA) reduces the total number of correct particle locations to 70% (resulting in an effective 0.035 ppp), while also substantially reducing the fraction of ghost particles present in the reconstruction.

In addition, Fuchs *et al.*<sup>88</sup> performed an uncertainty analysis of triangulation PTV using two cameras oriented 60° with respect to one another (see Figure 4.6). Two experimental setups were tested, the first used cameras with a SNR of 106 and the second used cameras with a SNR of 478. The test with the lower SNR achieved in-plane and out-of-plane standard deviations around 0.1 and 0.15 pixels, respectively. By increasing the SNR, these errors decreased to standard deviations of 0.05 and 0.09 pixels.

Most recently, Wieneke<sup>89</sup> introduced an iterative particle reconstruction (IPR) technique for performing triangulation that works well for particle seeding densities up to 0.05 ppp. The technique begins just the same as other triangulation methods: identification of particle images in 2D (though which method was used was not specified) and triangulation to determine 3D coordinates. The algorithm reconstructs particles in 3D and removes the images from each camera, allowing for further iterations of triangulation. After the initial 3D triangulation position and intensity of particle  $p$ ,  $I_{part}^i(x_i, y_i, p)$ , are calculated, a projection onto camera  $i$ ,  $I_{proj}^i$ , is calculated as

$$I_{proj}^i(x_i, y_i) = \sum_p I_{part}^i(x_i, y_i, p) \quad (4.10)$$

The residual is then calculated as the difference of the projection and the original image. A further step calculates the residual for a particle image center shifted  $\pm 0.1$  voxels in each direction. The particle location with the smallest residual is then selected. An intensity correction is then applied and the residual image is then recalculated. Images are removed from an image if the intensity falls below a threshold or if it is located within one pixel of another image. Due to the removal of nearby particle images, IPR only requires three of four cameras to have corresponding intensity peaks to locate a particle in 3D.

Figure 4.7 shows the flow diagram for iterative particle reconstruction. There is a large loop that triangulates new particle images and their projections. There is also an inner loop that corrects particle locations and calculates the residual images. This technique is highly iterative and the computational time was on the same order of magnitude as MART reconstruction (4.2.2). Despite the high computational cost, the algorithm is an improvement over 3D PTV for high-density images. Figure 4.8 shows that IPR can identify as many particles as MART and more than 3D PTV, which they say is the same as a single-pass IPR, for seeding densities up to 0.05 ppp. Beyond this limit, the algorithm's performance drops off. Not only does the algorithm provide high yield of true particle coordinates, but it also gives better accuracy than the other algorithms up to 0.05 ppp, as shown in Figure 4.9.

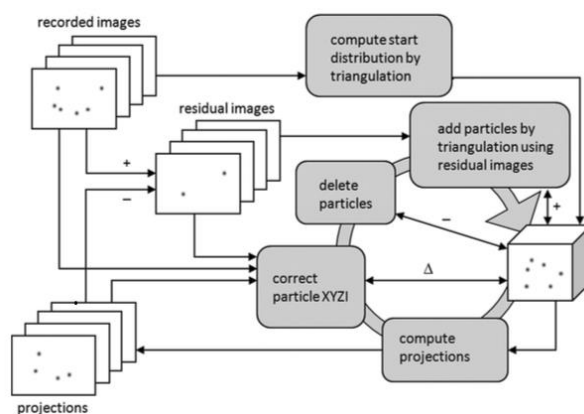


Figure 4.7. Flow diagram for iterative particle reconstruction.<sup>89</sup>

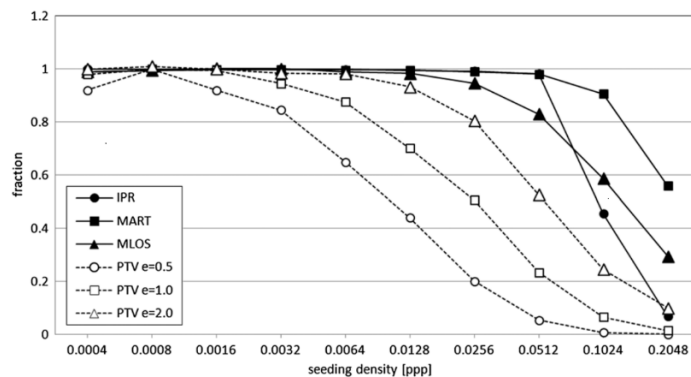


Figure 4.8. Fraction of detected true particles vs seeding density for IPR and other 3D reconstruction algorithms.<sup>89</sup>

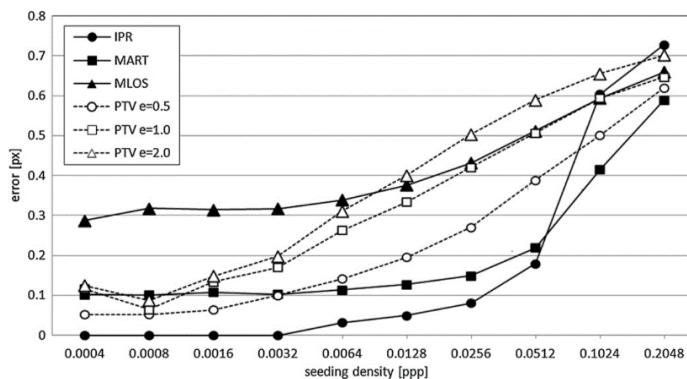


Figure 4.9. Mean 3D localization error vs particle seeding density for IPR and other 3D reconstruction algorithms.<sup>89</sup>

#### 4.2.1.3 Stereoscopic Methods

Stereoscopic PTV (stereo-PTV) uses only two cameras to determine three velocity components. While stereo-PIV is widely used for planar measurements, PTV a two-camera PTV setup is not particularly common. This is due to the fact that PTV requires accurate and reliable determination of particle coordinates, which is difficult using only two cameras. Applications of stereo-PTV have therefore been few.

The first use of stereo-PTV was in a micro-scale experiment using a two cameras and a stereomicroscope. Bown *et al.*<sup>90</sup> then Yu *et al.*<sup>91</sup> developed micro stereo-PTV algorithms using different tracking techniques. Stereomicroscopes are designed to allow for two cameras to image a single volume at microscales. There are two configurations that can be used: a Greenough type and a common main objective type (CMO). These configurations are shown in Figure 4.10, where the CMO type gives a translational configuration of stereo imaging and the Greenough type gives an angular displacement configuration. The CMO microscope offers the advantage of sharpness throughout the entire field of view and smaller bias and rms errors than the Greenough configuration (see Figure 4.11); however, the image generated using a CMO configuration is

axially asymmetrical because the beams do not pass through the center of the objective lens. Therefore, the optical aberrations are easier to correct for in the Greenough type microscope.

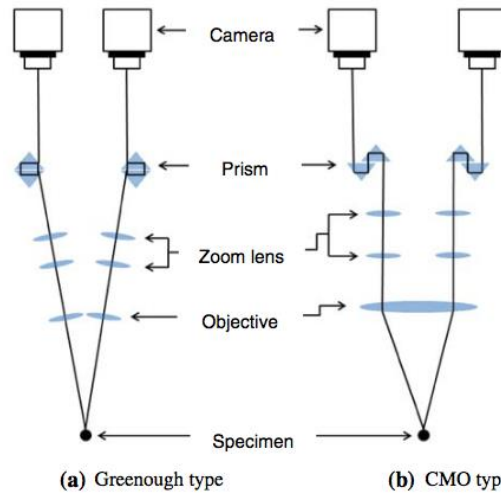


Figure 4.10. Schematics of stereoscope configurations with CMO (translation) and Greenough (angular displacement) type microscope systems.<sup>91</sup>

In order to determine 3D coordinates using two cameras, a calibration procedure is performed to determine a direct linear transform (DLT) for conversion between image and object spaces. This is a general mapping function

$$X = F(x), \quad (4.11)$$

where  $F$  is a third-order polynomial that relates image coordinates  $X$  to the object coordinates  $x$ . In order to determine object coordinates, stereoscopic correspondences must be determined. Bown *et al.*<sup>90</sup> suggested that a cross correlation should be performed on the left and right images to estimate the image shift between the two image spaces. For each particle in the first image space, a correspondence is considered valid if exactly one particle in the second image space lies within a specified tolerance (set to 10 pixels) of the estimated image shift. Stereoscopic pairs are then transformed to object coordinates using the  $F$  matrix.

Bown *et al.*<sup>90</sup> performed an experiment using a backward-facing step flow with a field of view of 900 x 720  $\mu\text{m}$  and a displacement on the order of 10 pixels, the particle tracks were averaged over 100 images. 2D averaging was done on a 45x45  $\mu\text{m}^2$  grid to be compared to stereo-PIV and 3D averaging was done across 10 x 10 x 10  $\mu\text{m}^3$  with a depth of 45  $\mu\text{m}$ . Comparing these average values to DNS solutions, the 2D PTV results had uncertainties of 2% of mean velocity for in-plane flow and 3% of mean velocity for out-of-plane flow. This is an improvement compared to 3% in-plane and 7% out-of-plane uncertainties for PIV. 3D averaging results gave experimental uncertainties of 2% of the mean velocity for in-plane components and 5% of the mean velocity for out-of-plane components. The 3D averaging gives a full three-dimensional, three-component mapping of the flow field. This averaging method is only applicable for steady flows.

Yu *et al.*<sup>91</sup> took a different approach for low seeding densities that finds particle tracks in each image individually using a relaxation method (see Section 5.3). The particle positions are then transformed into the object plane in physical coordinates. Particle tracks having the shortest distance in the projections between left and right images are then selected as stereoscopic matches. Using the two corresponding 2D displacements in the left and right images, the stereoscopic approach for determining 3D displacements is then used, where a transformation matrix,  $F$ , is approximated for each individual particle pair as  $\nabla F$ .

$$\begin{bmatrix} \Delta X_l \\ \Delta Y_l \\ \Delta X_r \\ \Delta Y_r \end{bmatrix} = \begin{bmatrix} F_{xl,X} & F_{xl,Y} & F_{xl,Z} \\ F_{yl,X} & F_{yl,Y} & F_{yl,Z} \\ F_{xr,X} & F_{xr,Y} & F_{xr,Z} \\ F_{vr,X} & F_{vr,Y} & F_{vr,Z} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \quad (4.12)$$

This gives three components of velocity; however, it does not assign a depth, thus this technique is a two-dimensional three-component (2D3C) PTV technique.

In order to test the accuracy of this stereoscopic tracking algorithm, an experiment was performed by translating a known calibration target in the  $x$ - and  $z$ -directions. This was done using both cross-correlation stereo-PIV and the stereo-PTV algorithm used by Yu *et al.*,<sup>91</sup> with both a Greenough type and a CMO type stereomicroscope. The results are presented in Figure 4.11, where the bias errors associated with PIV techniques were reduced for PTV both in-plane ( $x$ -direction) and out-of-plane ( $z$ -direction). Both the  $x$ - and  $z$ -direction bias errors were less than 1% of the true displacement for PTV, although the RMS errors ranged from 4% to 8% in the  $x$ -direction and 2% to 7% in the  $z$ -direction. In the majority of tested cases, the stereo-PTV algorithm using the CMO type stereomicroscope had the lowest errors.

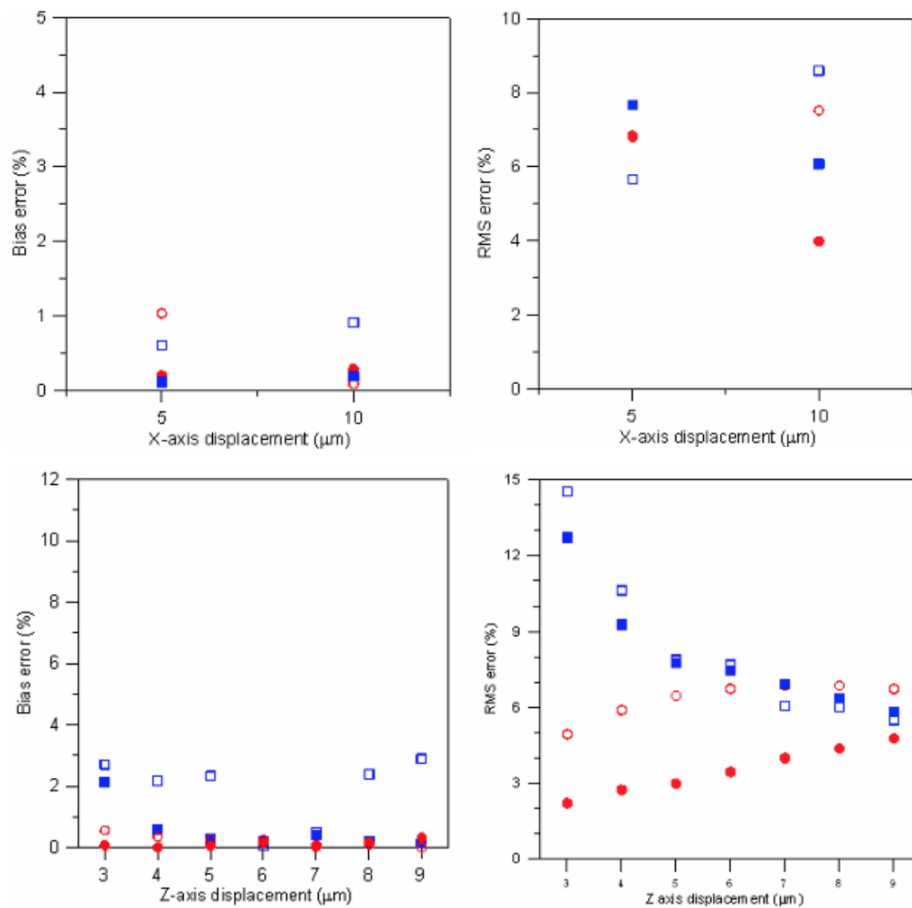


Figure 4.11. Bias and RMS errors in the  $x$  and  $z$ -directions using stereo-PTV (circles) and stereo-PIV (squares) with Greenough type (outlined) and CMO type (filled).<sup>91</sup>

Peterson *et al.*<sup>92</sup> made a modification to the microscopic stereo-PTV algorithms presented in order to use a single camera. Figure 4.12 shows how a camera's view was split in two in order to record two images in the same frame. Both views were reoriented by 90 degrees by beam twister mirror setups, which made more efficient use of the image sensor space.

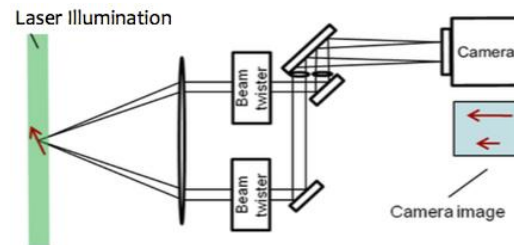


Figure 4.12. A single camera using two off-center optical paths through a single lens.<sup>92</sup>

Peterson *et al.*<sup>92</sup> used the concept of epipolar lines, as described in section 4.2.1.2 with an additional constraint on image-similarity. Stereoscopic pairs are determined by minimizing the difference between epipolar lines in 3D space. If a pair consists of only particles that are not claimed by another pair, then they are considered a match. Otherwise, the pair with the highest degree of similarity is chosen as valid. The seven proposed similarity criteria are: peak intensity, summed intensity, total number of pixels, image width, image height, maximum value of a cross-correlation between raw particle images, and maximum value of a cross-correlation between binarized particle images. No single similarity criteria was capable of separating valid pairs, so thresholds were determined for all seven criteria such that each allowed for 90% of valid sets to be recognized. The proposed algorithm was applied to a flow exiting a hose and compared to PIV results. The two algorithms agreed within 9% of the velocity measurements.

Bao and Li<sup>93</sup> proposed a method for stereoscopic particle image pairing that utilizes a combination of photogrammetric and binocular vision principles. In binocular vision, a point

source of light that is imaged outside of the focus plane will appear as a circle. The depth of the point source from the camera lens can be estimated from the radius of this blur circle  $r$  as

$$D = \frac{Fv_0}{v_0 - F - rkf}, \quad (4.13)$$

where  $F$  is the focal length of a thin convex lens,  $v_0$  is the distance from the lens to the image plane, and  $f$  is the  $f$ -number of the camera lens. The parameter  $k$  needs to be determined from a calibration procedure. Figure 4.13 shows the geometry of this binocular vision setup and shows how an out-of-focus particle will appear as a circle in the image plane.

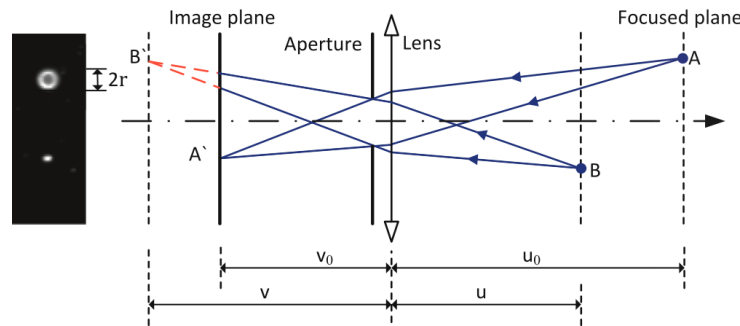


Figure 4.13. Geometry of defocused image formation.<sup>93</sup>

The procedure for stereoscopic pairing proposed by Bao and Li<sup>93</sup> began by selecting an unpaired reference particle,  $p_0$ , in the left image. The epipolar line segment with a depth range obtained from the blur circle radius and its uncertainty is then calculated. The candidate particle,  $p_{min}'$ , in the right image that is closest to this line segment is then selected. If this particle is within the depth estimate of  $p_0$ , then it is considered the optimal pair particle. If it is not, then  $p_{min}'$  is labeled as an outlier and the next closest particle is considered as a candidate. After a valid particle is selected, then this procedure is repeated for each reference particle in the left image. This technique was shown to increase the correct pairing rate for a simulation with 2200 particle pairs (0.0011 ppp) from 56.1% to 74.3% when compared to an epipolar line nearest neighbor analysis.

This technique showed that stereoscopic correspondences can be determined in a 40 mm thick volume using only two cameras.

Panday *et al.*<sup>94</sup> introduced a technique for 3D localization with two cameras that uses an epipolar line search and ant colony optimization. Ant colony optimization (ACO) is based on the methodology of an ant colony when searching the surrounding area for food. If a particular path is considered good by the ant, it will leave behind some amount of pheromone for other ants. In the context of stereo-PTV, a path chosen by an ant is from one particle image from camera one to another particle image in camera two. The path between particle images is considered good if the distance between epipolar lines is short. Ants make decisions about which path to follow based on pheromone amounts and local information, characterized by Equation 4.14, where  $p^k$  is the probability an ant takes the path from particle  $i$  to particle  $j$ ,  $\tau$  is the pheromone measure between the two points,  $\eta$  is the reciprocal of the distance between particles  $i$  and  $j$ , and  $\alpha$  and  $\beta$  are the parameters that determine the importance of local and global information,

$$p^k(i, j) = \frac{[\tau(i, j)]^\alpha [\eta(i, j)]^\beta}{\sum_{l \in N^k} [\tau(i, l)]^\alpha [\eta(i, l)]^\beta}. \quad (4.14)$$

Ants visit every particle,  $i$ , in the left frame and decide which particle in the right frame to visit. After every ant visits every particle in its itinerary, pheromone amounts are updated. Equation 4.15 gives the update procedure for the pheromone left by an ant between points  $i$  and  $j$ , and  $\rho$  is the pheromone decay rate, which ensures there is no unlimited accumulation of pheromone.

$$\tau(i, j) \leftarrow (1 - \rho)\tau(i, j) + \sum_{k=1}^m \Delta\tau^k(i, j), \quad (4.15)$$

where

$$\Delta\tau^k(i, j) = \begin{cases} 1/L^k & \text{if } (i, j) \in T^k \\ 0 & \text{otherwise} \end{cases}, \quad (4.16)$$

where  $T_k$  is the set of portions of route included in the itinerary of ant  $k$  and  $L^k$  is the total distance of the itinerary route. Thus, the change in pheromone amount is determined by how well an ant has chosen its route.

Two modifications to this ant system (AS) method were proposed. The first modification was named the ant colony system (ACS) method. It differed from AS in three aspects: the selection probability of an ant agent to go from one particle in the left image to the other particle in the right image, the global updating rule was only applied to routes belonging to the best ant tour, and local pheromone updating is applied as ants construct a solution. The new selection probability for an ant agent is

$$p^{*k}(i, j) = \frac{[\tau(i, j)] [\eta(i, j)]^\beta}{\sum_{l \in N^k} [\tau(i, l)] [\eta(i, l)]^\beta}, \quad (4.17)$$

$$p^k(i, j) = \begin{cases} 1 & j = \arg \max p^{*k}(i, l) \\ 0 & \text{otherwise} \\ p^k(i, j) = p^{*k}(i, j) & q > q_0 \end{cases} \quad q \leq q_0, \quad (4.18)$$

where  $q$  is a random variable uniformly distributed over  $[0,1]$  and  $q_0$  is a parameter between zero and one that determines the relative importance of exploitation versus exploration. If exploration is chosen ( $q < q_0$ ), then the ant makes a decision based on available information. If exploitation is chosen ( $q > q_0$ ), then biased exploration is performed as in the AS method.

The pheromone update in ACS is done in two steps. The first step takes place after each ant travels from a single path from particle  $i$  to  $j$ . This updated is

$$\tau(i, j) \leftarrow (1 - \psi)\tau(i, j) + \psi\tau_0, \quad (4.19)$$

where  $\psi$  is the local evaporation rate of pheromone. This update causes paths that are selected to have a reduced pheromone so that other ants are persuaded to choose different path in an iteration. This keeps ants from converging to a common path. The second step of the pheromone update takes place after all ant agents complete their itinerary

$$\tau(i, j) \leftarrow (1 - \rho)\tau(i, j) + \rho\Delta\tau(i, j), \quad (4.20)$$

where

$$\Delta\tau(i, j) = \begin{cases} \frac{1}{L^+} & \text{if } (i, j) \in T^+ \\ 0 & \text{otherwise} \end{cases}, \quad (4.21)$$

where  $T^+$  is the best particle link route ever traveled by all the agents and  $L^+$  is the total distance of that best route. Thus only the best route is used to update pheromones.

The second modification of the AS algorithm is based on Q-learning properties. This algorithm is called Ant-Q and differs from ACS only in the first step of the local pheromone updating rule, which is given by

$$\tau(i, j) \leftarrow (1 - \psi)\tau(i, j) + \psi\gamma * \max_{l \in N^k} \tau(j, l), \quad (4.22)$$

where  $\gamma$  is a new parameter between zero and one. The idea behind this modification is to update local pheromone trails with a value, which was a prediction of the value of the next state.

Panday *et al.*<sup>94</sup> tested these algorithms using stereoscopic PIV standard images with a resolution of 256 x 256 pixels. The parameters used in this study were:  $\alpha$  and  $\beta$  were 1.0 and 5.0, respectively,  $\rho$  was 0.5,  $\tau_0$  was 1.0,  $\psi$  was 0.5, and  $\gamma$  was 1. Five iterations were done using 20 ant agents. The correct matching rates for each ant colony algorithm compared to a nearest neighbor epipolar line search is shown for standard images with a range of seeding densities from 0.0024-0.024 ppp in Table 4-2. Ant colony pairing improved the performance of stereo matching over the

entire range of particle densities tested. The limit of the algorithm was a correct pairing rate of 95% for Ant-Q with a seeding density of 0.0024 ppp. The authors suggested that increased accuracy would require more cameras.

Table 4-2: Ant Colony optimization results for Stereo image pairing.<sup>94</sup>

Series #/Frame #	Existing Pairs	Correct Pair Rate (%)			
		Conventional	AS	ACS	Ant-Q
#351/000	1546	68.69	72.06	71.6	72.12
#351/001	1526	70.77	71.75	72.02	72.08
#352/000	283	88.69	91.52	93.64	92.93
#352/001	277	88.09	92.77	92.42	92.78
#371/000	157	89.17	91.08	93.63	93.63
#371/001	160	89.38	95	93.75	95
#377/000	352	75	77.56	76.42	75.85
#377/001	352	76.42	80.11	82.1	81.82

#### 4.2.1.4 Scanning 3D PTV

An alternative technique for capturing 3D velocity fields is the use of scanning multiple planes separately, introduced by Hoyer *et al.*<sup>95</sup>. The technique typically uses one or two cameras as opposed to the four to six used in other 3D methods. Figure 4.14 gives an example experimental setup. Scanning PTV records multiple frames to reconstruct a single volumetric measurement; therefore, in order to compare to other 3D PTV techniques, scanning PTV (SPTV) must be able to scan an entire volume in the same time that a single frame is recorded otherwise. For this reason the frame rate of SPTV must be increased by a factor equal to the number of slices of flow measured over the frame rate of 3D PTV, which is limiting in the context of time-resolved measurement. The benefit of scanning is that a single camera frame only contains particle images from a thin cross-section of the observation volume, removing ghosts from the triangulation process.

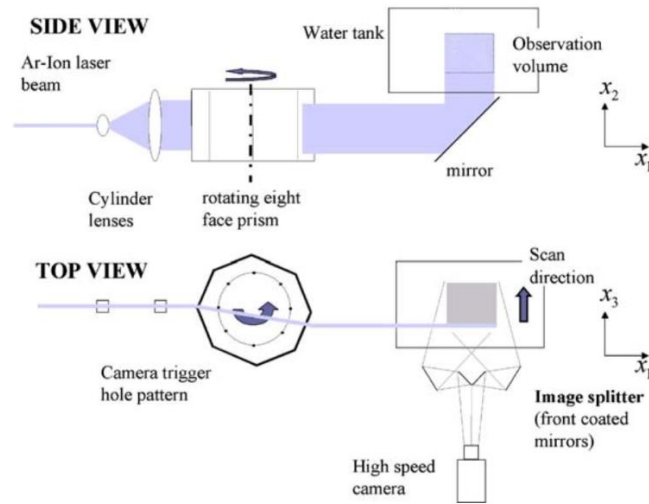


Figure 4.14. Experimental setup with a laser beam being expanded to a sheet and scanned through a volume using a rotating eight face prism.<sup>95</sup>

Hoyer *et al.*<sup>95</sup> used single camera with an image splitter to obtain four views of the observation volume. The four views were used for triangulation, as done by Maas *et al.*<sup>24</sup>, to determine precise 3D particle coordinates within each plane. When scanning a volume in SPTV, two phenomena need to be accounted for: frames are captured at skewed points in time and particles can move between slices between volume scans. When searching for a particle's track, Hoyer *et al.*<sup>95</sup> introduced the constraint that a particle can be expected to be found in either the same plane or an adjacent plane between two volume scans. This requires an estimate for the maximum out-of-plane flow velocity as well as an appropriate frame rate. When the issues of SPTV are handled properly, Hoyer *et al.*<sup>95</sup> found that they were able to double both the observation volume and the spatial resolution compared to photogrammetric 3D PTV. The SPTV identified on average 3900 particles per volume scan, which is equivalent to 0.0037 ppp for the single 1024 x 1024 pixel camera that was used. Due to their hardware limitations, the camera memory could only record data for 4 seconds at a time. Thereby limiting the ability of their SPTV system for statistical studies. With hardware improvements, however, this can be overcome.

#### 4.2.1.5 Defocusing Methods

Willert and Gharib<sup>96</sup> suggested a photogrammetric method based on the same principles but only using one camera. Called the Defocusing DPIV (DDPIV) technique, it uses a three-hole aperture, where the holes are placed at the vertices of an equilateral triangle centered about the lens' optical axis, to produce three images for each particle. Based on the separation of the images, the distance from a reference plane can be determined. This concept is shown in Figure 4.15, where imaging with a single on-axis pinhole is compared with imaging with two off-axis pinholes.

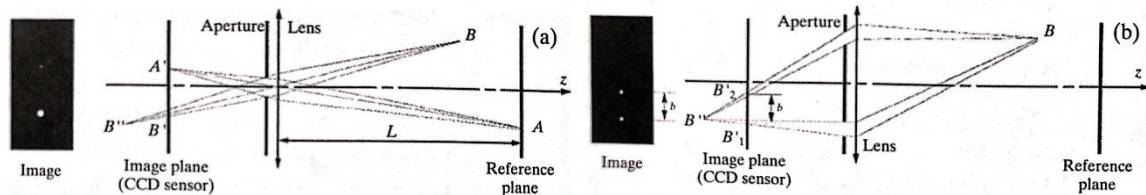


Figure 4.15. The principle of defocused imaging.<sup>97</sup>

In both cases, it can be seen that when a particle is on the reference plane, particle *A* images sharply onto the CCD sensor. In the former case, however, particle *B*, focuses behind the CCD, resulting in a slightly blurred image on the CCD sensor. In the latter, case, particle *B* still focuses behind the CCD sensor. However, because it images through the two off-axis pinholes, it results in two slightly blurred images onto the CCD sensor. In this manner, the separation between the particle's images, *b*, correlates with the particle's depth location, while the pair's lateral location correlates to the particle's planar position. Using this two-pinhole arrangement, Willert and Gharib<sup>96</sup> derived the relations expressing a particle's spatial location as

$$X = \frac{-x_0 Z}{ML}, \quad (4.23)$$

$$Y = \frac{-y_0 Z}{ML}, \quad (4.24)$$

$$Z = \frac{1}{\frac{1}{L} + Kb}, \quad (4.25)$$

$$K = \frac{1}{MLd}, \quad b = \frac{MD}{Z}(L-Z), \quad M = \frac{f}{L-f}, \quad (4.26)$$

where  $(x_o, y_o)$  are the geometric center of the particle images' locations,  $d$  is the pinhole separation,  $L$  is the distance from the aperture to the reference plane, and  $M$  is the geometric magnification.

The sensitivity of the system to detect depth changes is given as

$$\frac{\partial b}{\partial Z} = -\frac{1}{KZ^2}. \quad (4.27)$$

Once particle images were identified in the CCD images, these equations were used to obtain the particles' 3D spatial positions.

Yoon and Kim<sup>98</sup> noted that in the single lens microscopic applications, the distance between the pinhole mask and the lens plane is not negligibly small, and that the misalignment error of the pinhole mask with the lens is also not trivially small, thereby introducing noticeable errors. Towards this end, they proposed an  $x$ - $y$  compensation calibration procedure using polynomial calibration functions to correct for the  $x$  and  $y$  positions of the centers of the particles on the image plane. Using 17 particles on their target for this calibration procedure, they obtained uncertainties of 0.090/0.098/0.26  $\mu\text{m}$  for the  $x/y/z$  positions, respectively, across a 50  $\mu\text{m}$  volume depth. With this approach, they were able to obtain velocity measurements over a micro backward-facing step with dimensions 768 $\mu\text{m}$  long by 388  $\mu\text{m}$  wide by 50  $\mu\text{m}$  deep, though no particle image density was reported. Pereira *et al.*<sup>99</sup> suggesting using a calibration procedure to obtain camera parameters, which entailed placing 2  $\mu\text{m}$  fluorescent particles onto a transparent plate, and sequentially moving

it through known depth locations and acquiring images. These were then used with equations 4.23-4.26 to identify 3D particle positions.

Because each particle is triply exposed onto the CCD, this technique is limited to low seeding. Tien *et al.*<sup>100,101</sup> suggested placing three different color filters over each pinhole and to use a 3-CCD color camera. Using the color filters and a white backlight, particle images appear as cyan (green and blue), yellow (red and green), and magenta (blue and red) on a white background (see Figure 4.16). Furthermore, Tien *et al.*<sup>101</sup> recognized that color aberration results in different peak locations for different wavelengths (see Figure 4.17) caused particle location errors, and resolved this with a new configuration by aligning red, green, and blue LED sources with each of the pinholes for illumination (see Figure 4.18), as well as eliminating any crosstalk between color channels. The volume imaged was cubic and the mean and variance of error in out-of-plane location are plotted in Figure 4.19, where it can be seen that the mean errors are within 1.5%, the  $x$  and  $y$  standard deviation errors are within 0.04%, and the  $z$  standard deviation errors are within 0.25%. To further improve this approach, Tien<sup>102</sup> suggests using two dichroic beam splitters to separate the incoming light into three spectral ranges, and to image each of these onto monochrome cameras (see Figure 4.20).

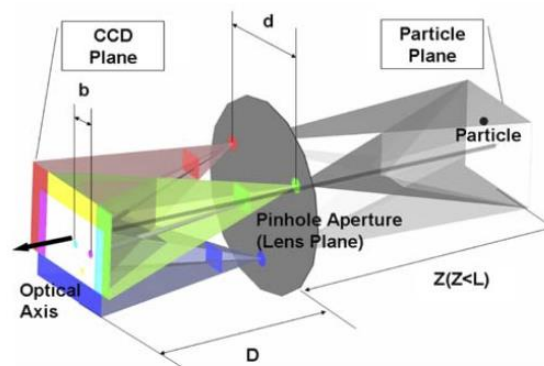


Figure 4.16. Defocused images obtained from a black particle illuminated with a white backlight and three color filters.<sup>100,101</sup>

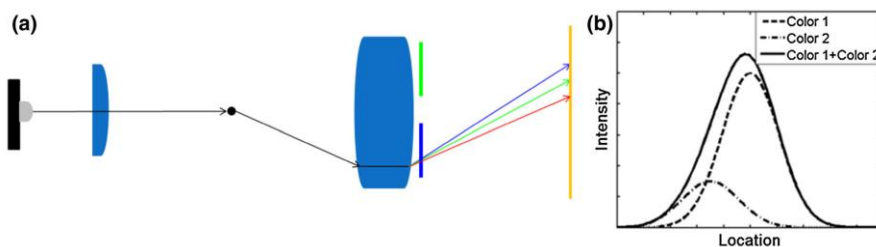


Figure 4.17. Particle image peak shifts due to color aberration: a color aberration due to white light illumination, b the skewed Gaussian distribution

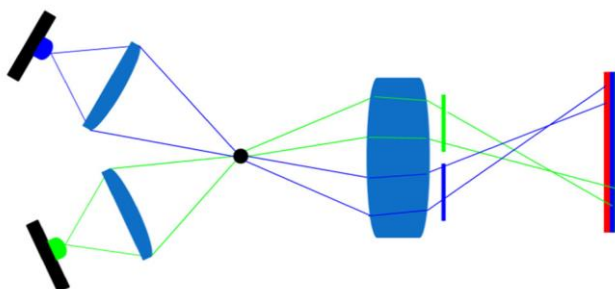


Figure 4.18. Improved illumination configuration<sup>94</sup>

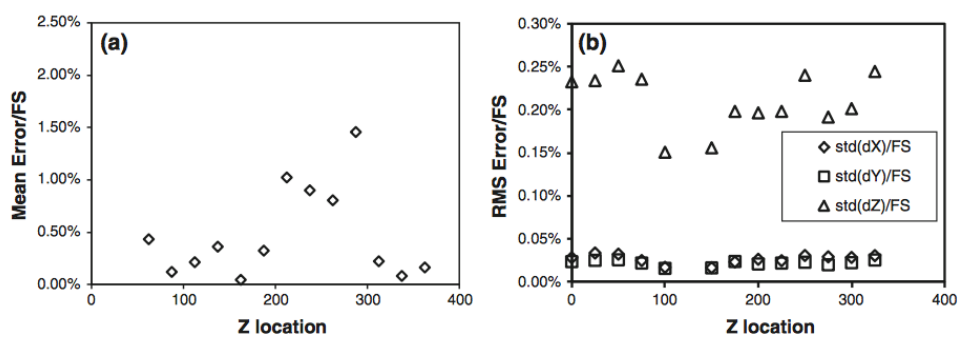


Figure 4.19. Out-of-plane location error as a percentage of the calibration range<sup>101</sup>.

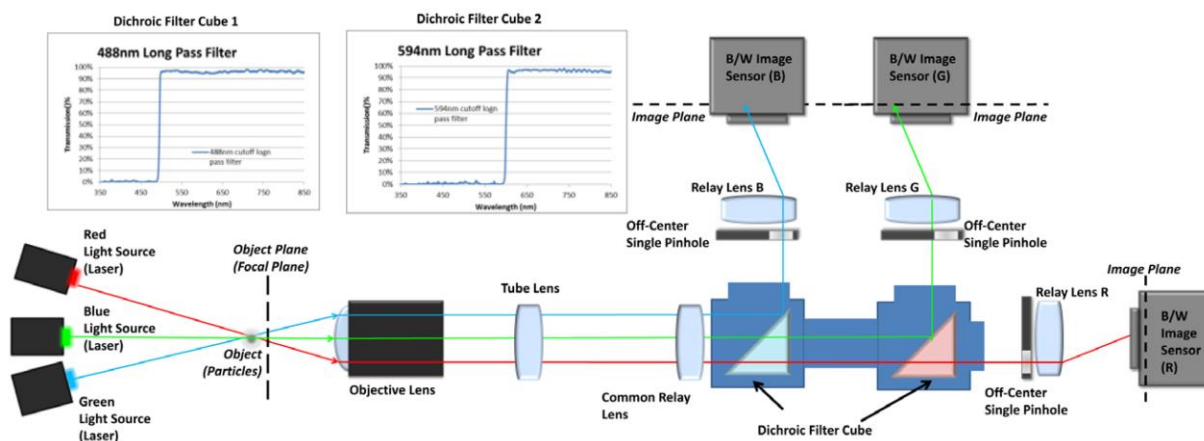


Figure 4.20. Concept of the proposed multi-spectral 3D $\mu$ PTV system.<sup>102</sup>

An alternative implementation of DDPIV uses three separate cameras. In this setup, each CCD is arranged around a common system axis, such that the axis of each camera and lens is radially tilted towards the common system axis so that they all intersect at the same point on the system axis.<sup>103,104</sup> This multi-camera arrangement, shown in Figure 4.21<sup>103</sup>, is equivalent to the single camera setup shown in Figure 4.13 aside from the fact that each particle will only create one image on each CCD, thereby eliminating in an alternative approach the problem of image overcrowding associated with the initial DDPIV system.

Grothe and Dabiri<sup>104</sup> defined an improved volume of interest by taking into account the dimensions of the CCDs. The new definition changes the imaging volume from a tetrahedron into a truncated rectangular pyramid. This was done by shifting the optical axis of the system to the half-height and half-width of the triangle, rather than its centroid. This allows for full exposure of the CCDs and a more practical rectangular, symmetric imaging volume. The comparison of camera setups and imaging volume are shown in Figure 4.22 and Figure 4.23, respectively. The performance of the modified setup was shown to be similar to that of Kajitani and Dabiri. The absolute error ratio for the rectangular domain was shown to be 0.68% lower than that of the tetrahedral domain.

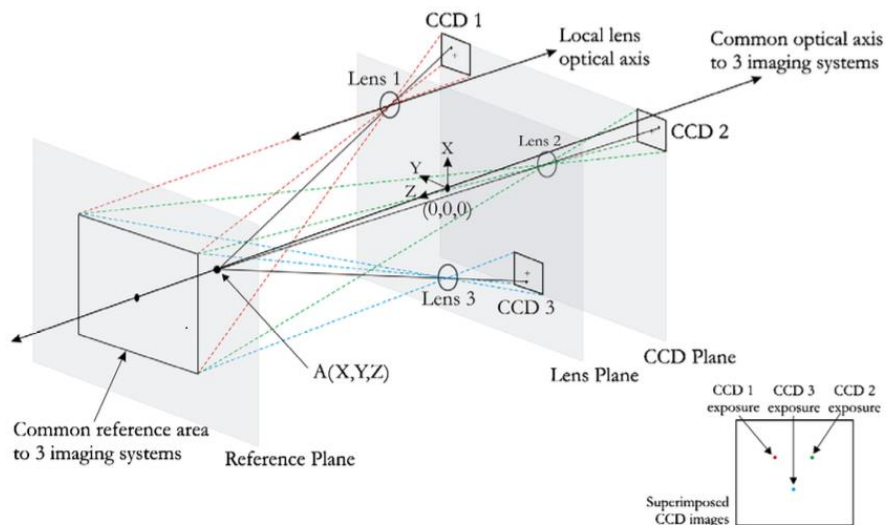


Figure 4.21. Implementation of 3D defocusing PIV.<sup>103</sup>

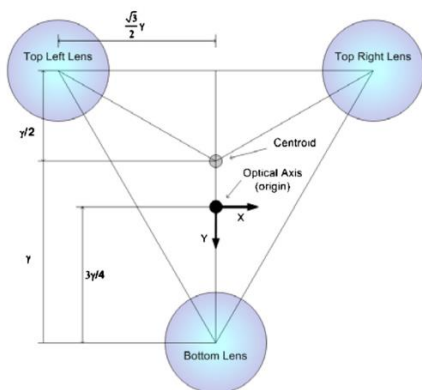


Figure 4.22. Location of the optical axis for a rectangular imaging domain.<sup>104</sup>

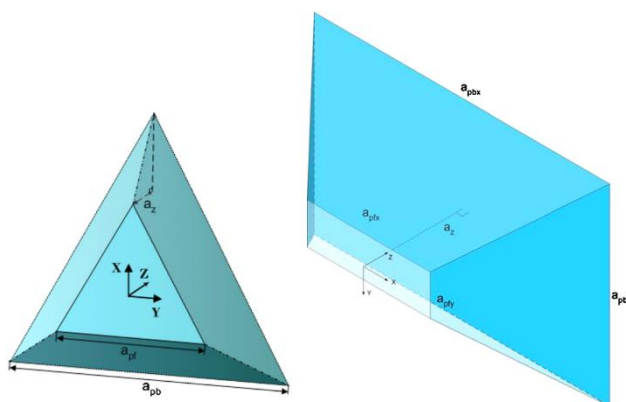


Figure 4.23. Tetrahedral (left) and rectangular (right) imaging domains for defocusing PIV.<sup>104</sup>

Table 4-3: Comparison between two-dimensional, Kajitani & Dabiri, and Grothe & Dabiri system uncertainties.<sup>104</sup>

	2D Derivation	Kajitani & Dabiri	Grothe & Dabiri
$\delta(dX)$	$K Z d\Delta x\sqrt{1+\frac{2X^2}{d^2}}$	$K Z \gamma\Delta x\sqrt{1+\frac{X^2}{\gamma^2}}$	$K Z \gamma\Delta x\sqrt{1+\frac{X^2}{\gamma^2}}$
$\delta(dY)$	$K Z d\frac{\Delta x}{2}\sqrt{1+\frac{2Y^2}{d^2}}$	$K Z \gamma\Delta x\sqrt{1+\frac{Y^2}{\gamma^2}}$	$K Z \gamma\Delta x\sqrt{17+8\frac{X}{\gamma}+16\frac{X^2}{\gamma^2}}$
$\delta(dZ)$	$\sqrt{2}KZ^2\Delta x$	$KZ^2\Delta x$	$KZ^2\Delta x$
$\delta(db)$	$\sqrt{2}\Delta x$	$\Delta x$	
$\frac{\delta(dZ)}{\delta(dY)}$	$\frac{2 Z }{d\sqrt{1+\frac{2Y^2}{d^2}}}$	$\frac{ Z /\gamma}{\sqrt{1+\frac{2Y^2}{\gamma^2}}}$	$\frac{4 Z /\gamma}{\sqrt{17+8\frac{X}{\gamma}+16\frac{X^2}{\gamma^2}}}$

Pereira and Gharib<sup>105</sup> applied the technique for particle densities on the order of 0.024 ppp. For a three-pinhole system, Kajitani and Dabiri<sup>103</sup> defined  $\gamma$  as the distance from the pinholes' geometric center (system's optical axis) to each of the pinholes, and  $\zeta$  as the distance from the particle image's triplet geometric center to each of the particle's images, and derived a full three-dimensional characterization of the three-pinhole system proposed by Willert & Gharib.<sup>96</sup> Grothe and Dabiri<sup>104</sup> further improved on the full characterization of this system. Uncertainty analyses were performed for the three systems and are shown in Table 4-3.

The error for  $\delta(dY)$  shown in Table 4-3 shows that this error is minimum on the system optical axis, however it grows with lateral locations. Grothe and Dabiri<sup>104</sup> plotted their results in Figure 4.24, showing the error ratio of in-plane and out-of-plane measurements. Here, it can be seen that errors are dependent on the non-dimensional depth and off-axis position. The peak error ratio occurs at the system optical axis, and drops with lateral location per given depth location. This is due to increasing errors in  $\delta(dY)$  for given  $Z$  locations and hence fixed errors in  $\delta(dZ)$ .

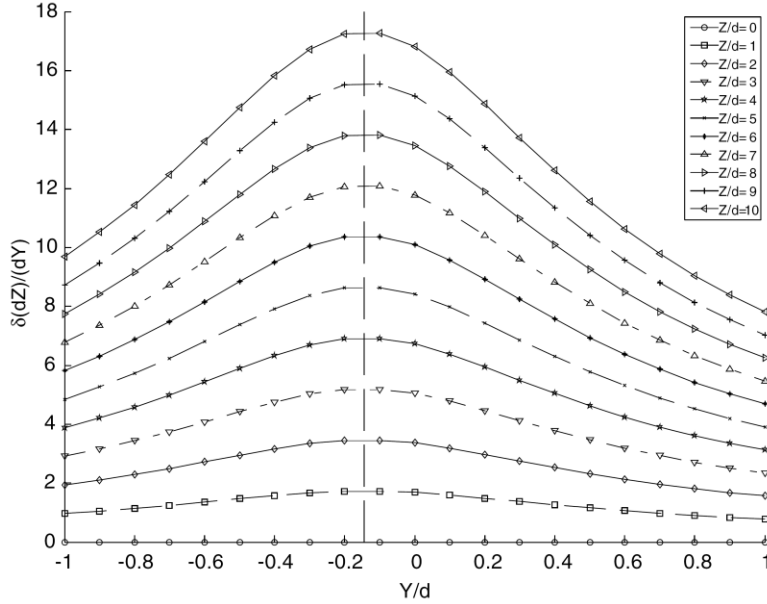


Figure 4.24. Error ratio of the out-of-plane to in plane measurements.  $Z$  and  $X$  are both normalized with respect to  $d$ , the length of a side of the equilateral triangle aperture arrangement.<sup>106</sup>

#### 4.2.1.6 Anamorphic Methods

The astigmatism PTV (APTV) technique<sup>106-115</sup> implements an anamorphic approach using a single camera and a cylindrical lens to stretch the particle images from circular to elliptical, such that depth information can be extracted from this deformation. While this method has been used in macroscopic scale experiments,<sup>107,108,113</sup> it was first used in microscale experiments by Kao & Verkman,<sup>106</sup> and has since predominantly been used in microscale experiments. Using the lens equations

$$\frac{1}{u_x} + \frac{1}{v_x} = \frac{1}{f} \quad \& \quad \frac{1}{u} + \frac{1}{v_y} = \frac{1}{f'}, \quad (4.28)$$

Chen *et al.*<sup>109</sup> showed that the measurable range,  $\Delta u$ , of the system is

$$\Delta u \approx \left( \frac{f-d}{f \times f_x} + \frac{p_2}{(v_x + p_2) \times v_x} \right) u_m^2 - p_1, \quad (4.29)$$

where  $u_m$  is the distance from the center of the measurable range to the first principal plane of the combined lens, where  $u_x$  and  $u_y$  are the distances between the in-focus planes and the first principal plane of the combined lens in the  $x$  and  $y$  directions, respectively,  $v_x$  and  $v_y$  are the image distances in  $x$  and  $y$  directions, respectively;  $f$  and  $f_c$  are the focal lengths of the field and cylindrical lenses, respectively;  $d$  is the distance between the two lenses;  $f'$  is the effective focal length of these two;  $p_1$  and  $p_2$  are the distances from the first and second principal planes of the combination of these two lenses to the field lens, and

$$p_1 = \frac{f \times d}{f + f_c - d}, \quad p_2 = \frac{(f - d)d}{f + f_c - d}. \quad (4.30)$$

If the cylindrical lens is close to the field lens, where  $d=0$ , and  $p_1 = p_2 = 0$ , then the measurement field reduces to

$$\Delta u \approx \frac{u_m^2}{f_c}, \quad (4.31)$$

showing that the measurable range is inversely proportional to the focal length of the cylindrical lens.

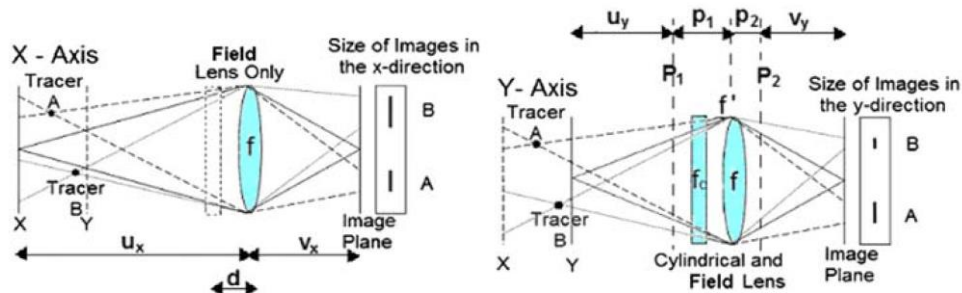


Figure 4.25. Schematics of the anamorphic technique using cylindrical optics when applied to particle imaging. The left part of this picture illustrates how the imaging system works in the  $x$  direction, and the right part shows the system operation in the  $y$  direction.<sup>109</sup>

Figure 4.25 shows an example of the anamorphic images obtained with an APTV system, where it can be seen that the particle shape transitions from a horizontal elliptical profile to a vertical elliptical profile as the particle transitions through the depth of the imaged volume.

The uncertainty of the depth location is given as<sup>109</sup>

$$\Delta z = \frac{\delta d}{2(NA)M} = \frac{\delta s}{2(NA)^2}, \quad (4.32)$$

where  $\delta d$  is the uncertainty of the particle image diameter within the CCD plane,  $\delta s$  is the uncertainty in the wavefront sag measurement,  $M$  is the magnification, and  $NA$  is the numerical aperture. Furthermore, a compensation factor was developed to adjust for  $x$  and  $y$  particle location perspective errors. With calibration, the resulting depth position errors are  $2.8\mu\text{m}$  over a  $500\mu\text{m}$  depth range. Further studies by Cierpka *et al.*<sup>110</sup> (see Figure 4.27) show the sensitivity of astigmatism algorithms with a single camera at various noise levels using simulated data to the estimated to true axis ratio,  $\varepsilon_a$ , (left), and to the subpixel position error (right), where  $NL$  is the ratio between the peak noise intensity levels and the peak image gray value. Here, particle image diameters were greater than three pixels. The plots show that the axis ratio and subpixel position error is about 0.5% and  $\pm 0.05$  pixels for a 5% noise level, and that the subpixel position error increases to  $\pm 0.1$  pixels for 20% noise levels.

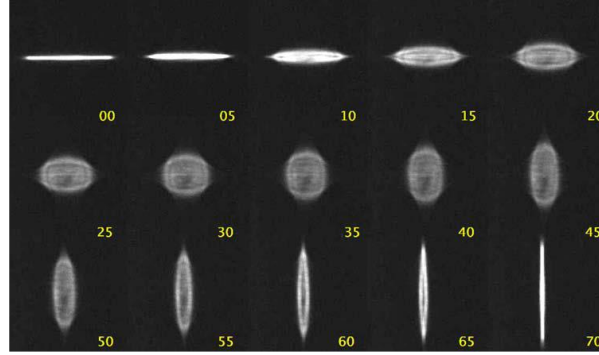


Figure 4.26. Anamorphic images of a fluorescent micron-sized particle traversed along the optical axis (the yellow numbers indicate the particle displacement in microns)<sup>109</sup>

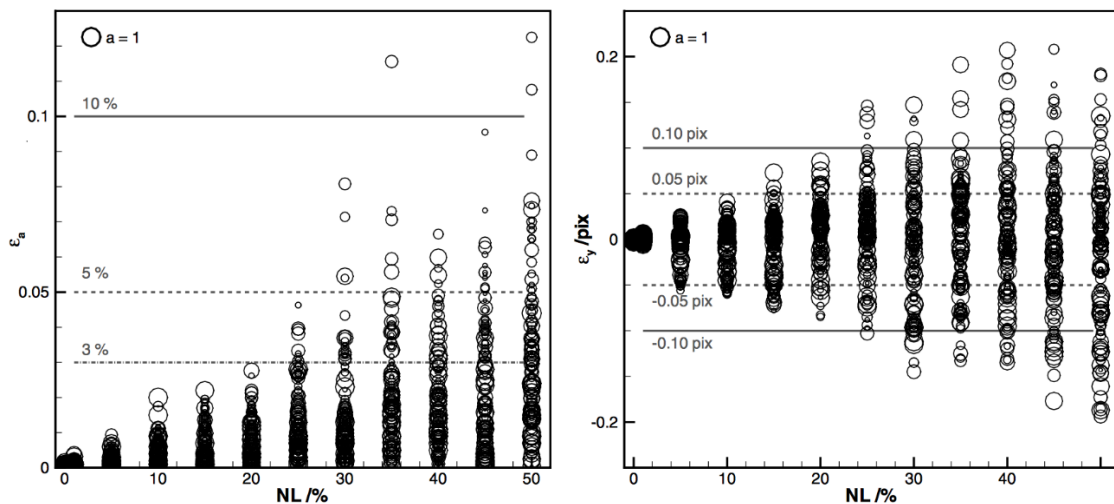


Figure 4.27. (left) Error in axis ratio and (right) in-plane displacement error with respect to noise level for single camera astigmatism particle localization.<sup>110</sup>

Cierpka *et al.*<sup>111</sup> proposed an intrinsic calibration method in order to extend the measurement volume's depth and to account for all image aberrations. In this work, they were able to image Poiseuille flow within a straight rectangular micro-channel with a cross-sectional areas  $200 \times 500 \mu\text{m}^2$ , and in-plane and out-of-plane velocity uncertainties of 0.9% and 3.7% of the centerline velocity, respectively. Seeding densities per image were 0.00006 ppp, and 2000 images were averaged to obtain these results. In a later study, Rossi & Kähler,<sup>112</sup> through an analytical study, identified three non-dimensional parameters,  $k_1$ ,  $k_2$ , and  $k_3$ , that were most important in optimizing APTV performance.  $k_1$  represents the ratio between the magnifications in the  $x$  and  $y$

directions,  $k_2$  represents the ratio between the main focal planes' distance and the measurement volume depth, and  $k_3$  represents the ratio of the diameter of the in-focus to the out-of-focus particles at the edges of the measurement volume. For optimized performance, they recommend that  $k_3$  ( $k_3 < 0.15$ ) should be kept as low as possible, which states that the depth-of-focus should be small with respect to the measurement volume height;  $k_2$  should be chosen depending on the calibration procedure used.

Fuchs *et al.*<sup>114</sup> suggested adding a second astigmatism camera and using a stereoscopic PTV approach using triangulation when out-of-plane measurements of position and velocity are of the utmost importance. Figure 4.28 shows how the uncertainty is reduced with the addition of a second camera in that the stereoscopic uncertainty is the intersection of the uncertainty regions of each of the cameras. With 0.0005 ppp seeding densities, the statistical measurement error from stereoscopic astigmatism PTV was found to be in the range of 3-5  $\mu\text{m}$  for  $x$  and  $y$  locations and 12-13  $\mu\text{m}$  for  $z$  locations for an observation volume of  $40 \times 40 \times 20 \text{mm}^3$ , which is equal to an error in the  $z$  position estimate of 0.03% of  $d_z$ .

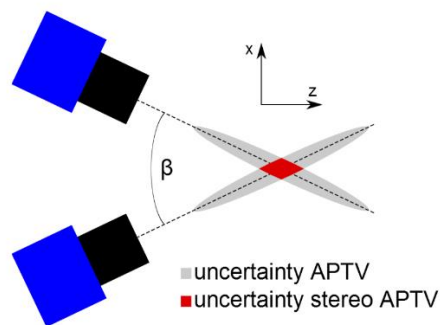
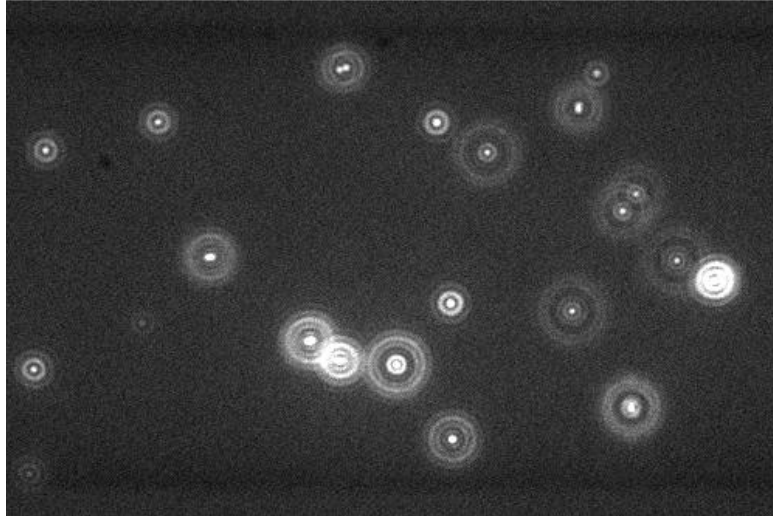


Figure 4.28. Qualitative uncertainty between astigmatism PTV techniques.<sup>114</sup>

## 4.2.1.7 Particle Diffraction Pattern Methods

Figure 4.29. Flow image at 0.225s.<sup>124</sup>

Particle diffraction patterns can only be generated with clear interference fringes if the reflected or emitted light from the particle is coherent or partially coherent, and is then superposed constructively and destructively. This diffraction pattern is three-dimensional, extending both in depth through the optical axis, and radially about the depth axis. In the absence of aberrations, this pattern is symmetric, and asymmetric in the presence of aberrations.<sup>116</sup> The point spread function (PSF) is the analytical expression of this diffraction pattern, and its square corresponds to the intensity distribution seen within an image. Gibson and Lanni<sup>117</sup> derive a prediction for the square of the PSF that includes aberration, and based on Kirchhoff's scalar diffraction integral formula, as

$$I(x_d, y_d, \Delta z) = \left| C \int_0^1 J_0 \left[ k \frac{NA}{\sqrt{M^2 - NA^2}} \rho \sqrt{x_d^2 + y_d^2} \right] \exp[jW(\Delta z, \rho)] \rho d\rho \right|^2. \quad (4.33)$$

Here,  $NA$  is the numerical aperture,  $M$  is the magnification,  $(x_d, y_d)$  is the non-dimensional lateral particle locations,  $\Delta z$  is the depth location from the focal plane,  $k$  is the wave number,  $\rho$  is the normalized radius in the exit pupil of the objective, and  $W(\Delta z, \rho)$  is the phase aberration function.

The phase aberration function is a product of the wave number and the optical path difference (OPD), which is an important element in addressing optical aberrations. Detailed derivations and expressions for the OPD can be found in Gibson and Lanni<sup>117</sup>, and Park.<sup>118</sup>

An example of particle diffraction patterns is shown in Figure 4.29. Speidel *et al.*<sup>119</sup> showed particle diffraction patterns can be used to map 3D flow fields. Using a single nanoparticle within an epifluorescent microscopic imaging system, they demonstrated the concept by moving a 216 nm fluorescent microbead away from the focal plane, showing that the particle first blurs, then multiple rings appear, where the outer most ring is the brightest. In this regard, they called this approach the optical serial sectioning microscopy (OSSM). They then calibrated the outermost ring to the depth location. Wu *et al.*<sup>120</sup> demonstrated this approach also works for micrometer scale particles, as well as for imaging multiple microbeads. In addition, through simulation, they showed that the outermost ring is dominated by spherical aberrations.

Park and Kihm<sup>121</sup> calibrated the outermost diffraction ring to the particle's depth location with respect to the focal plane. An image processing algorithm was developed to identify the particles' center locations as well as the diameter of the outermost diffraction rings. An analytical model was formulated to create a calibration curve to calibrate the outermost diffraction rings to the depth location. Figure 4.30 shows the analytical evaluations of the diffraction patterns at different depth locations Figure 4.30b,c and the depth the outermost diffraction ring size relation in Figure 4.30d. Figure 4.31 illustrates the image processing algorithm they developed image flow around a sphere. Figure 4.31a shows a raw CCD image on which this algorithm is applied. Figure 4.31b illustrates the resulting image after a multi-step binarization process. Figure 4.31c illustrates the circle-fitting process. Conceptually, a sufficiently large circle centered around each pixel is started, and is then incrementally shrunk, whereupon each iteration, the brightness of the circumference is examined.

Then the outermost rings of each particle are identified, and its center is assigned as the  $x$ - $y$  location of the particle, and its depth is determined from the calibration results obtained as shown in Figure 4.30d. Figure 4.31d shows examples of particle positions that have been identified using this procedure.

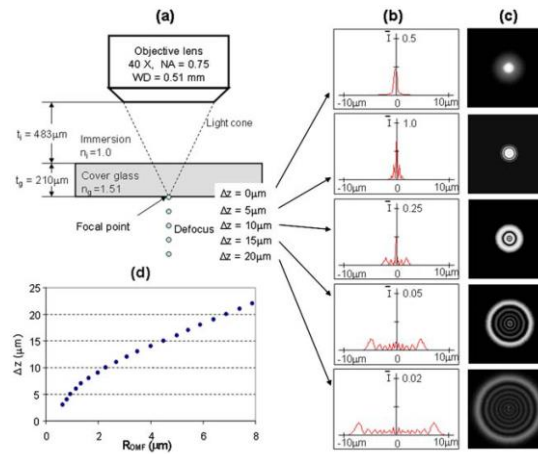


Figure 4.30. The concept of deconvolution microscopy application for three-dimensional micro-PTV technique; **a** Critical parameters of experimental setup such as specifications of objective lens, defocus distance between particles and lens's focus, thicknesses, and refractive indices of cover glass and immersion medium, **b** PSF of diffracted particle image in each indicated defocus distance, **c** Graphical diffraction images, which are drawn based on the corresponding PSF, and **d** A relation between a defocus distance ( $\Delta z$ ) and a radius of the outermost fringe ( $R_{OMF}$ )<sup>121</sup>

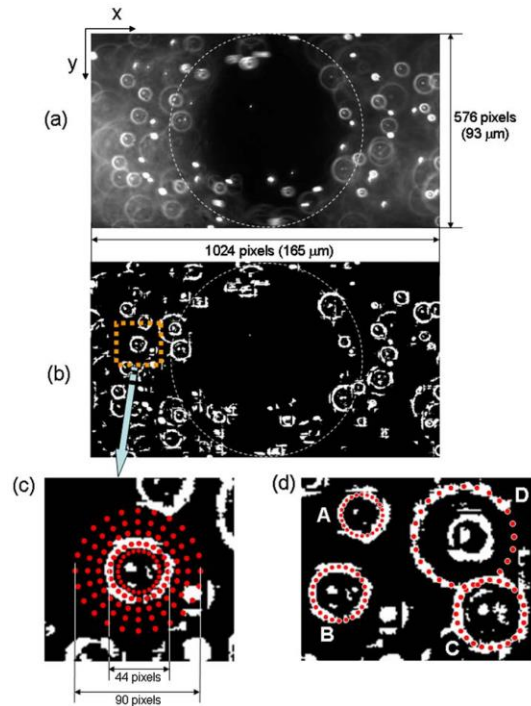


Figure 4.31. **a** A raw diffraction image captured with an intensity resolution of 8-bit and a shutter speed of 30-ms. The dashed line indicates an interrupting micro-sphere. **b** After an image processing the 8-bit grayscale image is converted to a binary image that contains only 0's and 1's. **c** An identification of three-dimensional particle position is conducted by using axisymmetric 24-point data. **d** A few identified three-dimensional particle positions:  $A(x-, y-, R_{OMF}$ -pixel numbers: 158, 178,16),  $B(140, 245, 22)$ ,  $C(240, 200,40)$  and  $D(264, 261, 29)$ <sup>121</sup>

Luo *et al.*<sup>122</sup> also used the Gibson and Lanni<sup>117</sup> Kirchhoff's scalar diffraction model but with an OPD based on Haebelé<sup>123</sup> to predict the diffraction pattern details through a glass cover, which introduced a mis-matched refractive index, within an air immersed microscope objective. Rather than using only the outermost diffraction ring, they used a combination of first dark and first bright fringe diameters. In a follow-up study, Luo *et al.*<sup>124</sup> studied the effects of cover glass thicknesses and the use of partially coherent light emanating from fluorescent particles on the diffraction pattern sizes and their effect of particle depth identification. For partially coherent light, they found that the wavelength distribution of the fluorescence significantly affected particle diffraction pattern at large depth locations. Here, they were able to show the predictions worked well for

small depths (<30 $\mu\text{m}$  for 1100 $\mu\text{m}$  cover glass and <10 $\mu\text{m}$  for 160 $\mu\text{m}$  coverslip) for monochromatic light, though they also reported that the scalar diffraction model over-estimated the diffraction pattern sizes. In addition, for the smaller depths, the central interference fringes were used to estimate particle depth locations, while for larger depths, the outermost fringes were used to make this estimation. In addition, they found that smaller thicknesses of the cover glass produce larger changes in the diffraction pattern through depth, resulting in shorter depths where particles can be tracked. In further work, Luo and Sun<sup>125</sup> noted that though diffraction patterns change with depth, their central region are nearly Airy disk intensity distributions, which can be approximated with a Gaussian function. With this in mind, they developed an algorithm that iteratively cross-correlated a Gaussian intensity distribution template

$$g(r) = g_0 \exp\left(\frac{-r^2}{2\Delta^2}\right), \quad (4.34)$$

where  $g_0$  is the maximum intensity at template's center,  $r$  is the radial distance from the center, and  $\Delta$  is the Gaussian peak width, through their image using

$$c(i, j) = \sum_{m=1}^M \sum_{n=1}^N g(m, n) \cdot f(i + m, j + n), \quad (4.35)$$

where  $c$  is the cross-correlation matrix,  $(\cdot)$  denotes a cross-correlation operation, and  $(M, N)$  denote the size of the Gaussian template, to identify diffraction centers. Once particle centers were identified, their earlier approaches were used to identify diffraction patterns, which were then cross-correlated with a calculated Gibson-Lanni distribution, using a normalized cross-correlation algorithm, to identify the particles' depth locations. Using this approach, the particles' centers were found with uncertainties within 0.02-0.04  $\mu\text{m}$  (0.1-0.2 pixels). For depth locations greater than 13 $\mu\text{m}$ , the diffraction patterns contained at least three fringe rings, and using a 1.1 mm cover

glass slide the accuracy was recorded to be better than  $0.1\ \mu\text{m}$ , or an order of magnitude larger than the particles' center accuracies.

Peterson *et al.*<sup>126</sup> proposed capturing calibration images of a particle at various known depth locations to create a calibration curve. From these images, the particle's center was determined, and the outermost ring was identified, which was used to calibrate against the known depth locations. To generate the calibration curve, the particle diameters,  $0.3$ ,  $1$ , and  $3\ \mu\text{m}$  were tested with 4 different microscopic lenses with magnifications of 20x, 40x, and 50x and effective numerical apertures of 0.6, 0.8, and 0.73, respectively. A quadratic was used to fit the data with determination coefficients that were better than 99.8%. Uncertainties in particle center locations averaged to 0.6 pixels; uncertainties on depth are directly related to determination of the outermost ring, which was 0.08 pixels for the larger rings, and 3.4 pixels for the smaller rings.

#### 4.2.2 Tomographic PTV

Both the purpose and the setup of a tomographic PTV (tomo-PTV) system are similar to that of a photogrammetric PTV system. Figure 4.32 shows an example setup with four cameras. A Tomographic PIV algorithm implements the steps outlined in Figure 4.33 for image reconstruction and cross-correlation analysis. The key step in this algorithm that translate exactly to tomo-PTV is the tomographic reconstruction. This is done using CCD pixel intensity data before any particle identification. Unlike other methods, the particle identification step is done in the three-dimensional space as a single step. In addition, a particle mask correlation algorithm (Section 3.2.3) has been extended into 3D for use in the voxel space for tomo-PTV.<sup>127</sup> On the other hand, voxel clusters can also be identified and the centroid can be estimated from three 1D Gaussian fits (Section 3.1.2).<sup>130</sup>

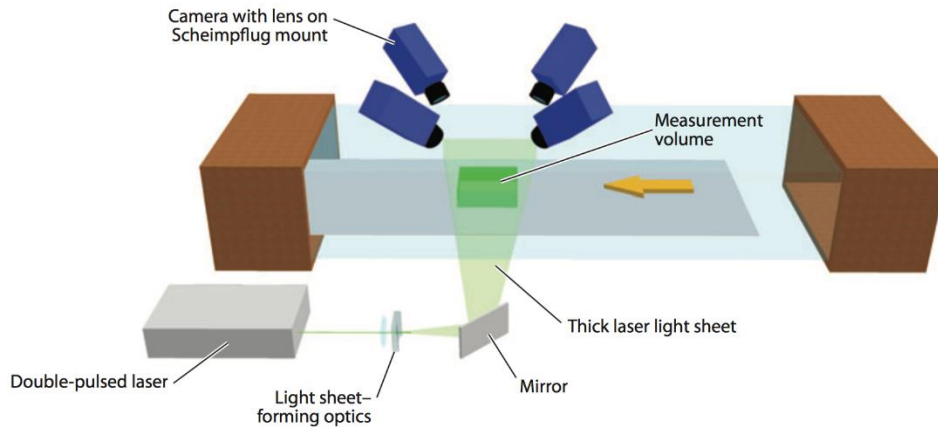


Figure 4.32. Four camera tomographic PIV setup.<sup>48</sup>

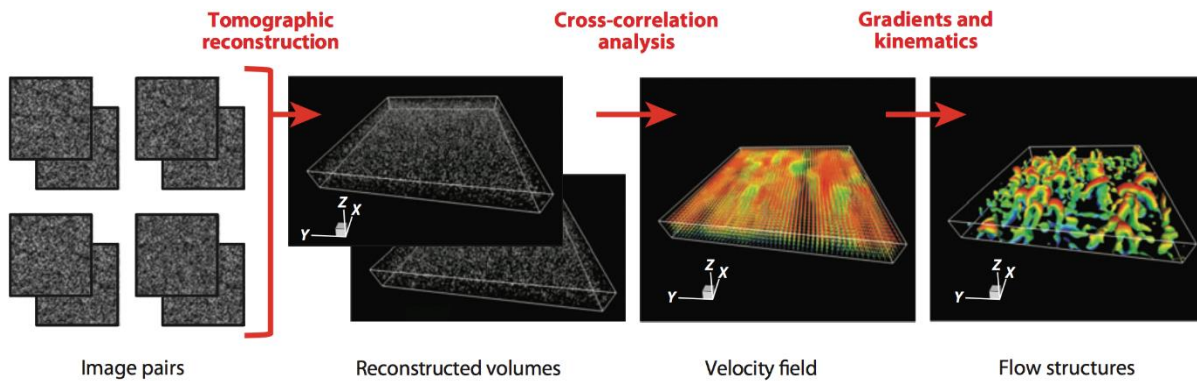


Figure 4.33. Steps for handling tomographic PIV data.<sup>48</sup>

The tomographic reconstruction algorithms for PIV and PTV are generally based on the multiplicative algebraic reconstruction technique (MART)<sup>127</sup>. The MART technique is based on image particle intensities captured by each camera and the correlation with the actual illuminated particle intensity. This relationship is given in equation 4.36 where  $E_i$  is the intensity of pixel  $i$  and  $I_{x,y,z}$  is the light intensity at position  $x, y, z$ .

$$E_i = \int I_{x,y,z} dl \quad (4.36)$$

In order to discretize and relate the  $x, y, z$  coordinates to the camera's pixels, a discretized 3D domain is created. These discrete elements are called voxels, and make up the measurement

volume. In this discretized form, Equation 4.36 is modified as shown below in Equation 4.37. A weighting factor,  $w_{ij}$  is introduced to relate each pixel intensity to each voxel intensity. These weighting factors rely heavily on an accurate calibration procedure in three dimensions. The 2D grid of markers is positioned at various depths within the measurement volume to obtain the proper mapping coefficients.

$$E_i = \sum_j w_{ij} I_j \quad (4.37)$$

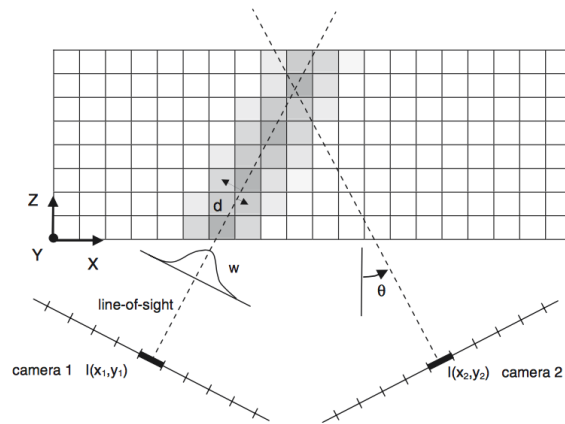


Figure 4.34. 2D representation of pixel and voxel interaction in tomo-PTV.<sup>128</sup>

Iterations of the MART algorithm are calculated using this concept. Initially, the voxel field is set to a uniform, nonzero value. Values of  $I_j$  are then iteratively updated using

$$I_j^{k+1} = I_j^k \left( \frac{E_i}{\sum_j w_{ij} I_j^k} \right)^{\mu w_{ij}} \quad (4.38)$$

An alternative algorithm for tomographic reconstruction is the multiple line-of-sight algorithm (MLOS), which generates voxel intensities from a single pass. MLOS is less accurate than MART, although using it as a first guess to allow MART to only operate on non-zero voxels can decrease the processing time by a factor of 5-10.<sup>89</sup> Simultaneous multiplicative algebraic reconstruction (SMART) is another technique that closely resembles MART, and can further decrease

computation time by performing some simultaneous calculations. The MLOS-SMART algorithm was shown to maintain the same accuracy as MART while decreasing computation time by a factor of 15.<sup>129</sup>

Once converged, the tomographic reconstruction will yield a 3D voxel space containing particle images. Using this voxel space, the particle identification algorithms explained in Section 3 can be used. Both the image segmentation and Gaussian centroid estimation steps can be extended for use in three dimensions.

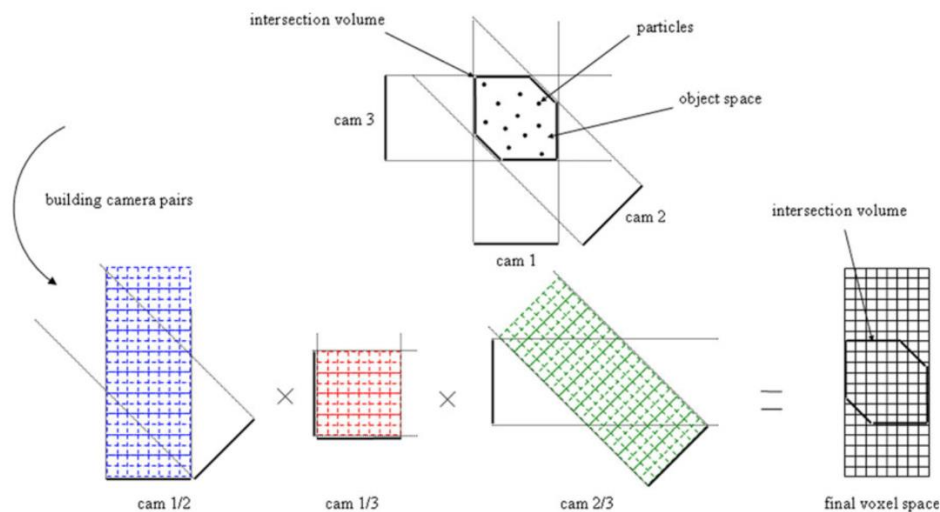


Figure 4.35. Volume definition with pixel interaction for telecentric imaging.<sup>130</sup>

Kitzhofer and Brücker<sup>130</sup> suggested that for small interrogation volumes, tomographic reconstruction can be simplified by using telecentric imaging. Telecentric lenses have constant magnification and parallel projection, making the relationship between pixels and voxels simpler, as shown in Figure 4.35. This simplification reduces computation time and performs similarly to MART reconstruction for particle seeding densities up to 0.1 ppp, although for a volume with depth equal to its height and width, the authors suggested using the algorithm for tracking with 0.01 ppp. The use of telecentric imaging limits the imaging volume dimensions to those of the

lenses used. In the case of Kitzhofer and Brücker, the lenses limited the volume to 90 x 90 x 90 mm<sup>3</sup>.

Tomographic reconstruction can be ambiguous, leading to what are called ghost particles in the reconstruction volume. High seeding particle density and too few cameras increase the probability of ghost particles appearing within the reconstruction. Elsinga *et al.*<sup>128</sup> defined a quality factor  $Q$

$$Q = \frac{\sum_{X,Y,Z} E_1(X,Y,Z)E_0(X,Y,Z)}{\sqrt{\sum_{X,Y,Z} E_1^2(X,Y,Z)E_0^2(X,Y,Z)}}, \quad (4.39)$$

to quantify reconstruction volume accuracy, which is a normalized correlation coefficient between the reconstructed image,  $E_1(X,Y,Z)$  and the exact distribution of light intensity,  $E_0(X,Y,Z)$ . The exact distribution is generated by a Gaussian intensity distribution of three voxel diameter centered at each known particle location. The quality factor as it relates to the number of cameras, particle density, calibration error, and viewing angle (angle between the outer cameras and the  $z$ -axis) can be seen in Figure 4.36. Unless stated otherwise, the following baseline parameters were used: three cameras at  $\theta = \{-20, 0, 20\}$  degrees, 0.05 ppp, five reconstruction iterations, and no calibration errors or image noise. A two-camera system is largely insufficient, and adding cameras increases  $Q$  rapidly as it approaches unity with five cameras. The optimal angle between cameras is 30 degrees for the three-camera setup, as larger angles increase the intercepted length of the line-of-sight between cameras, causing a larger number of ghost particles to appear. Increasing the particle seeding density also increases the number of ghost particles, although increasing the number of cameras allows for higher seeding densities. Experimental calibration error also reduces  $Q$ . Reconstruction quality greater than 0.75 was achievable for errors below 0.4 pixel.

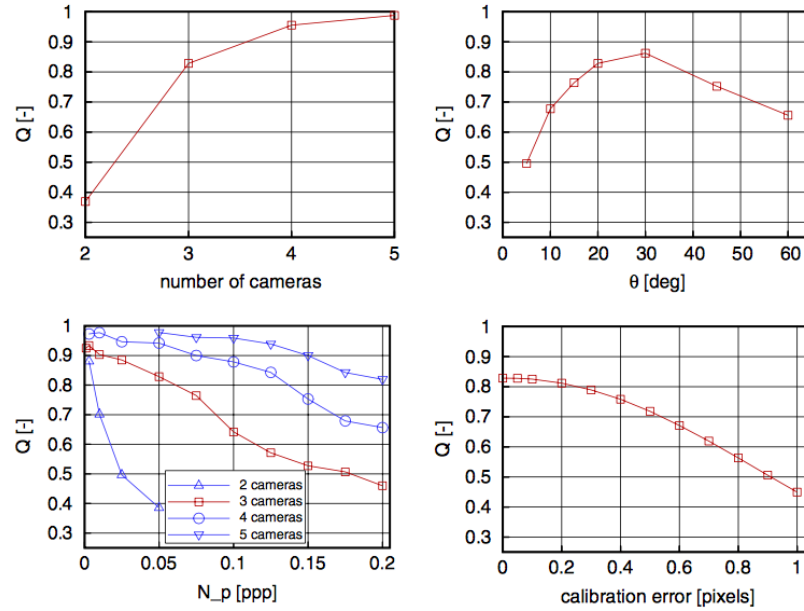


Figure 4.36. Tomographic reconstruction quality compared to exact intensity distribution as a function of setup and calibration.<sup>128</sup>

#### 4.2.3 Synthetic Aperture PTV

The synthetic aperture method, first introduced for PIV by Belden *et al.*<sup>131</sup> and extended to Lagrangian particle tracking by Bajpayee and Techet,<sup>132</sup> utilizes five or more cameras (nine were used in this study) to simulate a lens with an arbitrarily large aperture. By backprojecting particle images from each camera to a specific depth and merging the images, particles present at the chosen depth will be sharply focused. Particles located at a different depth will be blurred and dim. Performing this refocusing at depths throughout the volume of interest gives a 3D reconstruction of particle locations.

A reference plane within the observation volume is defined by the location of the planar calibration grid. A homography transform from the  $i$ th camera frame to the reference plane is defined based on a calibration:

$$\begin{Bmatrix} bx' \\ by' \\ b \end{Bmatrix}_i = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}_i \begin{Bmatrix} x \\ y \\ 1 \end{Bmatrix}_i, \quad (4.40)$$

where  $b$  is a constant,  $\mathbf{x}$  and  $\mathbf{x}'$  are the image and reference plane coordinates, and  $h_{jk}$  are coefficients obtained from camera calibration. The reference plane particle images are then refocused to new focal planes in the observation volume using

$$\begin{Bmatrix} x'' \\ y'' \\ 1 \end{Bmatrix}_i = \begin{bmatrix} 1 & 0 & \mu_k \Delta X_{Ci} \\ 0 & 1 & \mu_k \Delta Y_{Ci} \\ 0 & 0 & 1 \end{bmatrix}_i \begin{Bmatrix} x' \\ y' \\ 1 \end{Bmatrix}_i, \quad (4.41)$$

where  $\mathbf{x}''$  is the new focal plane coordinate,  $\Delta X_{Ci}$  and  $\Delta Y_{Ci}$  are the relative camera locations of the  $i$ th camera, and  $\mu_k$  is a constant that determines the location of the  $k$ th focal plane. The camera locations can be determined using a calibration given by Vaish *et al.*,<sup>133</sup> although a calibration image taken at each focal plane can be used to determine the  $h$  and  $\mu$  coefficients for each camera (see Section 4.3.3). The refocused images from all the cameras are then averaged to create a single image for each focal plane.

Figure 4.37 shows how two cameras capture the images of a point in a secondary plane,  $B$ , and in a reference plane,  $A$ . When the two images are remapped to the reference plane, the images of particle  $A$  will align and appear as a bright coherent point while the images of particle  $B$  will appear as two separate, dim points. This is due to the parallax between cameras and allows in-plane points to be differentiated from out-of-plane ones.

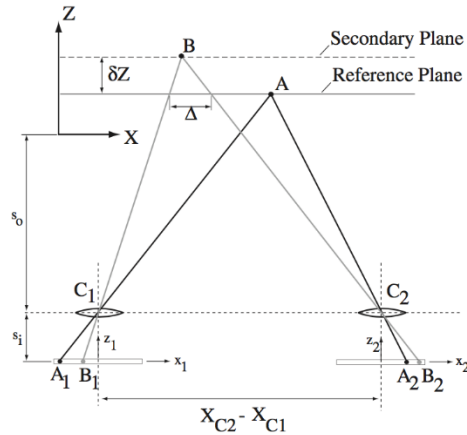


Figure 4.37. Schematic showing the reference plane and focal plane as well as illustrating the concept of parallax.<sup>131</sup>

Modifications to the refocusing transformation in the presence of refractive interfaces were introduced by Belden.<sup>134</sup> The computational cost of 3D reconstruction is high when each pixel is backprojected individually through refractive interfaces, therefore Bajpayee and Techet<sup>135</sup> suggested a homography transform to approximate the distortion when the interfaces are planar. This homography transform,  $H$ , is calculated by projecting the four corner points of each camera through the refractive interfaces:

$$P_j^z = HR_j^z. \tag{4.42}$$

$P_j^z$  are backprojected corner points from the  $j$ th camera onto the  $z$  plane and  $R_j^z$  is the projection of these points onto the camera. This transformation is shown in Figure 4.38. The homography fit approximates refraction through planar surfaces only.

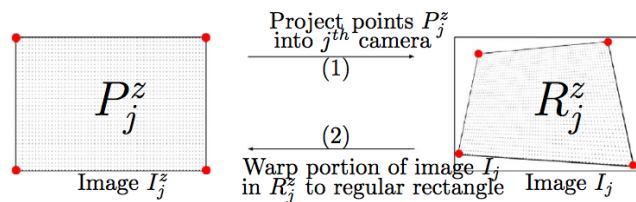


Figure 4.38. Projection and backprojection using the homography fit method for synthetic aperture refocusing.<sup>135</sup>

In the SAPTV algorithm used with synthetic data, the average localization error was found to be 0.2177 mm for a reference depth of 5 mm. In an experiment with a vortex ring, between 420 and 480 particles were detected in each frame, and 366 particles were visible for 1 second at 30 Hz.<sup>132</sup>

Belden *et al.*<sup>131</sup> used the synthetic aperture method to reconstruct 0.125 ppp in a simulation and 0.015 ppp in an experiment. They also showed that the reconstruction quality, defined by Elsinga,<sup>128</sup> depends on the number of cameras used and the mapping error onto the reference plane. Figure 4.39 shows that the synthetic aperture method performs best with 13 cameras and less than 0.2 pixels mapping error in the reference plane.

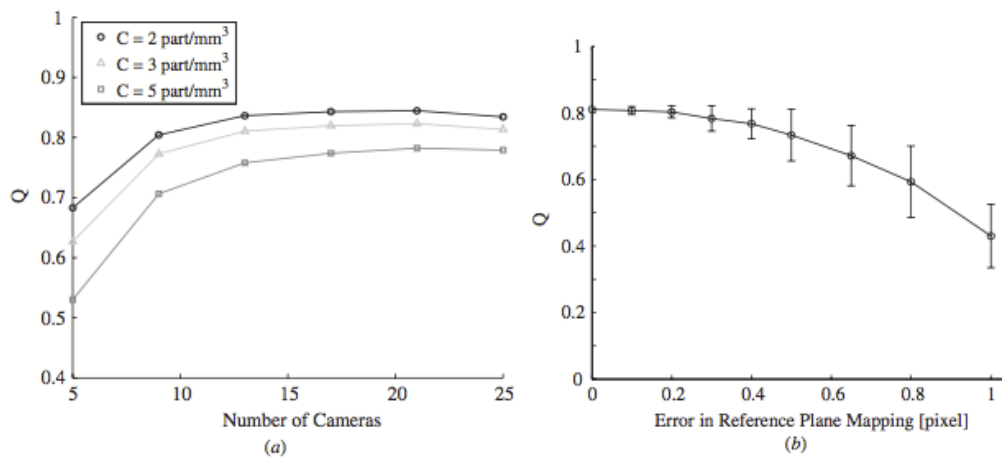


Figure 4.39. Reconstruction quality,  $Q$ , as a function of number of cameras (a) and reference plane mapping error (b) in a  $50 \times 50 \times 10 \text{ mm}^3$  volume.  $C$  is the particle seeding density.<sup>131</sup>

When compared to tomo-PTV, the homography fit synthetic aperture reconstruction increased the computational efficiency by 270 times when used for a  $512 \times 512 \times 128$  voxel volume and 0.1 ppp.<sup>135</sup>

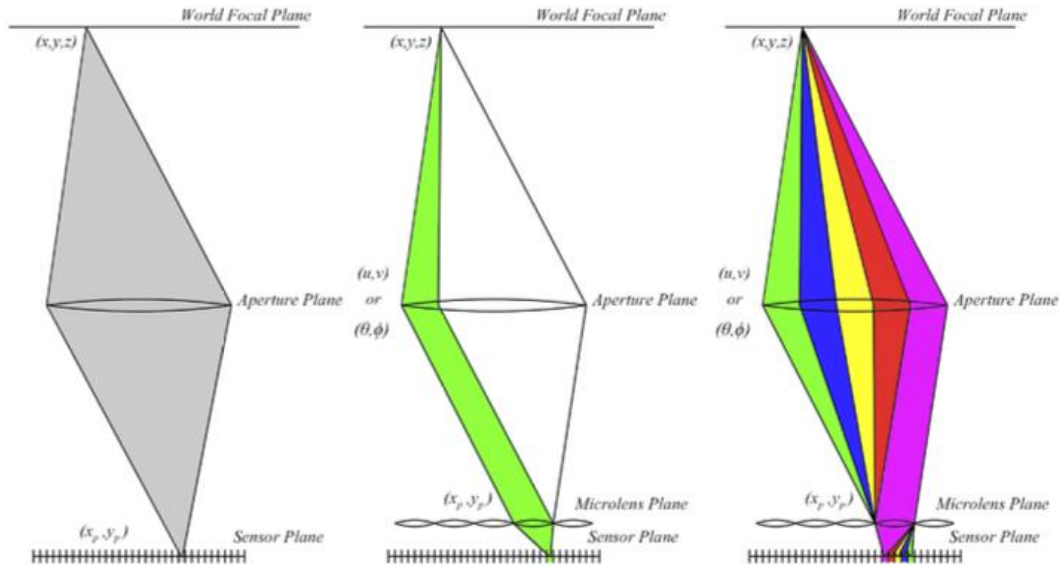


Figure 4.40. Schematic showing the application of micro-lenses for capturing the angle of incoming light in a plenoptic camera.<sup>140</sup>

#### 4.2.4 Plenoptic Imaging

The concept of using many many viewing angles refocusing to generate a volumetric reconstruction of particle images was made possible using a single camera with the introduction of the Lytro Camera. The Lytro Camera uses micro-lenses between the aperture and sensor to measure both the position and the angle of incident light rays. This setup is shown in Figure 4.40 and demonstrates the use of one camera to effectively view an object from multiple angles using adjacent pixels. The use of these micro-lenses, however, reduces the effective resolution of the camera being used.

Refocusing methods, such as those used in SAPTV, have been used to reconstruct 3D volumes from plenoptic data.<sup>136-139</sup> The refocusing technique uses the following transformation to transform image plane coordinates,  $x'$ , into a new refocused plane coordinates,  $x''$ :

$$\begin{Bmatrix} x'' \\ y'' \\ 1 \end{Bmatrix}_i = \begin{bmatrix} 1 & 0 & \mu_k \Delta X_{Ci} \\ 0 & 1 & \mu_k \Delta Y_{Ci} \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x' \\ y' \\ 1 \end{Bmatrix}_i, \quad (4.43)$$

where  $\Delta X_{Ci}$  and  $\Delta Y_{Ci}$  are the locations of the  $i$ th pixel relative to the corresponding micro-lens, and  $\mu_k$  is a constant that determines the location of the  $k$ th focal plane.

Tomographic methods for volume reconstruction of plenoptic imaging data have been used to get rid of out-of-plane images.<sup>140-142</sup> These algorithms use the MART algorithm described in 4.2.2. The 3D reconstruction is discretized into voxels, just as in tomo-PTV; however, the size of these voxels is chosen to be similar to that of a micro-lens rather than that of a pixel. The technique used in the refocusing algorithm is used here to find a weighting function, which defines the relationship between the image space and the voxel space. In order to define this weighting function, the plenoptic camera is mapped to each plane of voxel elements and a linear interpolation is performed to determine the intersection of each pixel with each voxel. The weighting coefficients are then normalized such that the sum of weights for each voxel is unity. The technique for determining this weighting function is described in detail by Farhinger *et al.*<sup>140</sup>

The MART plenoptic 3D reconstruction was tested using synthetic data in a 37.8 x 25.0 x 25.0 mm volume, which was discretized into 300 x 200 x 200 voxels. The errors in the algorithm were less than one voxel in-plane and 3 voxels in depth. Particle densities between 0.0001 and 0.01 ppp were tested. The reconstruction quality,  $Q$ , was above 0.9 for densities up to 0.0032 ppp, and dropped to 0.23 at 0.01 ppp.

Farhinger and Thurow<sup>142</sup> later suggested using a filtered refocusing algorithm, which calculates refocused intensities throughout the observation volume then filters based on the

fraction of projections that exceed a SNR threshold. Each pixel that contributes to a voxel's intensity is then checked against this threshold. The ratio of these valid projection and the total projections is then compared to a second user-selected threshold. If the threshold is exceeded, the reconstructed voxel intensity is used as normal; if the threshold is not met, the voxel intensity is set to zero.

Figure 4.41 shows a comparison of the errors of this filtered refocusing algorithm and the MART based algorithm. The standard deviation of in-plane position errors were 0.519 voxels and 0.746 voxels for filtered refocusing and MART, respectively. The errors in depth position estimates were 1.571 and 1.753 voxels.

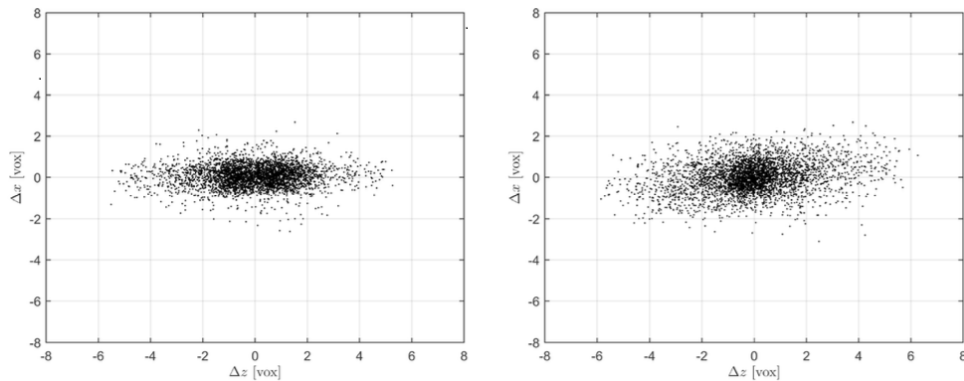


Figure 4.41. Errors in plenoptic imaging reconstructions using filtered refocusing (left) and MART (right).<sup>142</sup>

Anglin *et al.*<sup>143</sup> developed a deconvolution method for volume reconstruction of plenoptic data in order to reduce the computational cost of volume reconstruction. The refocusing algorithm is inherently shift-variant, thus for the purpose of deconvolution, a shift-invariant form was developed. A further benefit of this approach is that the deconvolution algorithm was able to reconstruct a volume in seconds, compared to the hours required for MART based reconstruction.

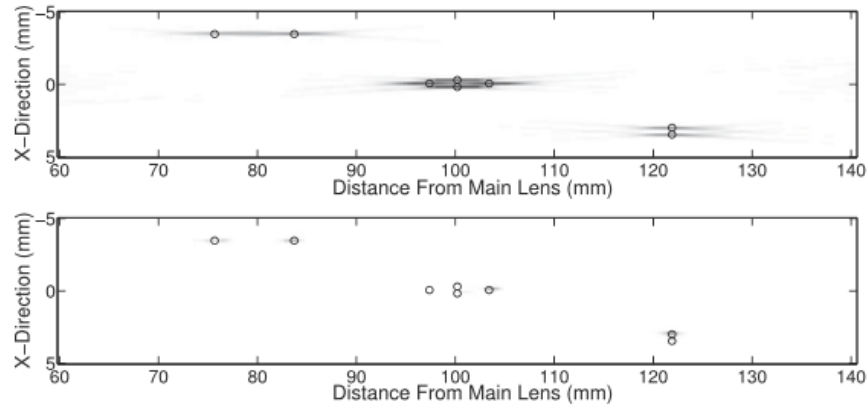


Figure 4.42. Reconstructed 3D volume using deconvolution (top), and MART (bottom) algorithms. “o” indicates point locations and inverted grayscale is used.<sup>143</sup>

Figure 4.42 demonstrates the differences between the MART reconstruction and the deconvolution algorithm. Deconvolution introduces image blur in the depth direction, causing roughly double the localization error compared to MART. The MART algorithm, on the other hand, failed to recover all the points in the reconstruction.

Chen and Sick<sup>144</sup> performed an experimental analysis of a plenoptic 2.0 camera, which is designed such that the micro-lens array is placed farther than one focal length away from the camera sensor. In doing this, the angular information is compromised for spatial resolution. The camera used was a 29-megapixel color plenoptic camera. The effective resolution of this camera was 7 megapixels. In this application of Plenoptic PTV, the particle image positions were identified in each micro-lens using a particle mask correlation algorithm (Section 3.2.3). Each particle had 14 corresponding images on different micro-lenses, allowing for robust determination of 3D position with only one camera.

In the context of the increased effective resolution of plenoptic 2.0 cameras, Chen and Sick showed that seeding densities should be kept below 0.25 particles per micro-lens in order to achieve good tracking results. They showed that error in displacement calculations was minimal

even at these high seeding densities. The impact of out-of-plane particle motion on particle displacement was tested; it was found that out-of-plane displacements greater than 2% of the measurement depth were overestimated by 20%, and an out-of-plane displacement of 16% of the measurement depth caused a 25% overestimate of in-plane displacement. It was suggested that the middle 40% (20 mm) of the examined depth (50 mm) gave accurate 3D reconstruction results for displacements less than 8 pixels (0.4 mm).

Generally, plenoptic cameras are effective for volume reconstruction when optical access is limited, as only one camera is used with one viewing angle. Accurate reconstructions are achievable using either computationally expensive MART algorithms, or faster deconvolution. The achievable seeding densities of these algorithms is limited due to the effective reduction in spatial resolution.

#### 4.2.5 *Holographic PTV*

This section discusses the basics of digital PTV holography as well as its state-of-the-art techniques for 3D reconstruction. For a more thorough discussion, the reader is referred to one of several review papers.<sup>145-147</sup>

Holographic imaging is a technique for capturing 3D information by using a film or digital camera to record a reference light source superimposed with the light scattered by an object. The illumination must be a coherent laser light source so that the object and reference interfere constructively and destructively. The holographic recording,  $I_h$ , captures the interference between a complex reference wave,  $\tilde{R}$ , and an object wave,  $\tilde{O}$ ,

$$I_h(x_h, y_h) = (\tilde{R} + \tilde{O})(\tilde{R}^* + \tilde{O}^*) = RR^* + \tilde{R}\tilde{O}^* + \tilde{R}^*\tilde{O} + \tilde{O}\tilde{O}^*, \quad (4.32)$$

where the asterisk indicates the complex conjugate and  $(x_h, y_h)$  represent the hologram image coordinates. The first term of this equation represents the mean intensity of the reference beam. The second and third terms generate the interference patterns used to determine depth. The final term represents the speckle pattern effect.<sup>145</sup> The interference patterns have a fringe spacing given by

$$d_f = \frac{\lambda}{2 \sin \alpha}, \quad (4.44)$$

where  $\lambda$  is the wavelength of the coherent light source and  $\alpha$  is the half-angle between the directions of light propagation of the two interfering waves. The fringe spacing is a maximum when the object wave and reference wave are aligned. Holographic imaging for PTV is done using either an in-line or an off-axis setup, referring to the alignment of the reference and object light sources. An in-line system uses a single laser for object illumination and as a reference wave, whereas off-axis systems will use two separate lasers. The low resolution of digital cameras limits experiments to in-line holography, as the fringe spacing in off-axis holography requires high resolution film.<sup>19</sup>

The greatest limitation of holographic imaging is the low spatial resolution in the depth direction. This is a result of the finite aperture and it causes reconstructed particle images to be elongated in the depth direction. In-line holographic systems typically elongate particles to be in the range of several millimeters up to tens of millimeters while off-axis systems only elongate particles to 0.2-0.5 mm.<sup>19</sup>

Typical in-line and off-axis holographic systems are depicted in Figure 4.43 and Figure 4.44, respectively. Note that the in-line setup is simple and does not require holographic film. In-line holography also benefits from forward scattering, requiring only 0.1% of the laser power of

standard 3D PTV methods.<sup>148</sup> Off-axis systems have better depth resolution than in-line systems and also do not suffer from the speckle pattern effect. Relative ease of use has caused digital in-line systems to be the most common for PTV applications.<sup>19</sup>

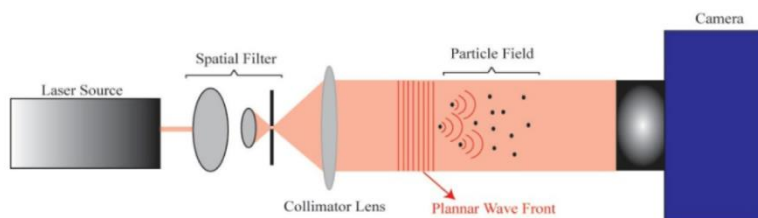


Figure 4.43. Typical lens-less digital in-line holographic PTV setup.<sup>149</sup>

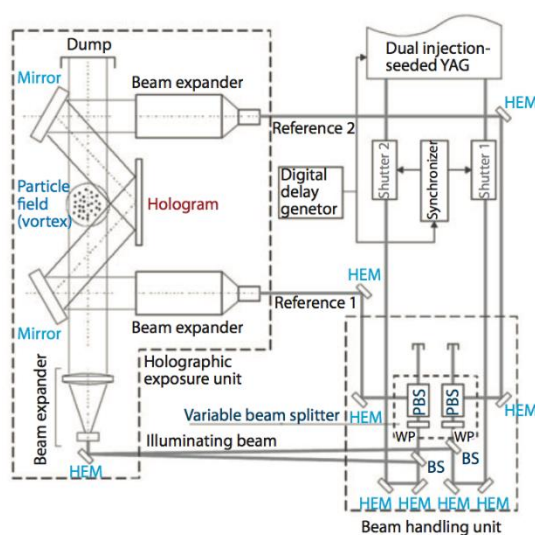


Figure 4.44. Double-exposure off-axis holographic PTV setup using two separate reference beams.<sup>145</sup>

Digital in-line holography is particularly well-suited for micro-scale experiments, as the depth of field can be extended to almost 1000 times that of a conventional microscope. The effect of particle elongation is also reduced to only 2-10 times the particle diameter, decreasing with increasing magnification.<sup>145</sup> The only required modification to the experimental setup in Figure 4.43 is the addition of a microscope objective lens, as shown in Figure 4.45.

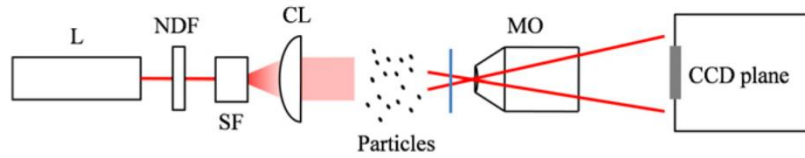


Figure 4.45. Microscopic digital in-line holographic PTV setup.<sup>150</sup>

A PTV hologram captured using in-line recording is shown in Figure 4.46a. The interference patterns that are generated by particles appear as fringe patterns with spacing that varies depending on a particle's depth position. The fringe patterns overlap and interact with those of surrounding particles, causing a speckle pattern effect. This is the cause of the granular appearance of holograms.<sup>19</sup> Volume reconstruction algorithms refocus the raw hologram to different depth planes. One such reconstructed plane is shown in Figure 4.46b. The fringe patterns of in-plane particles become dark, focused points.

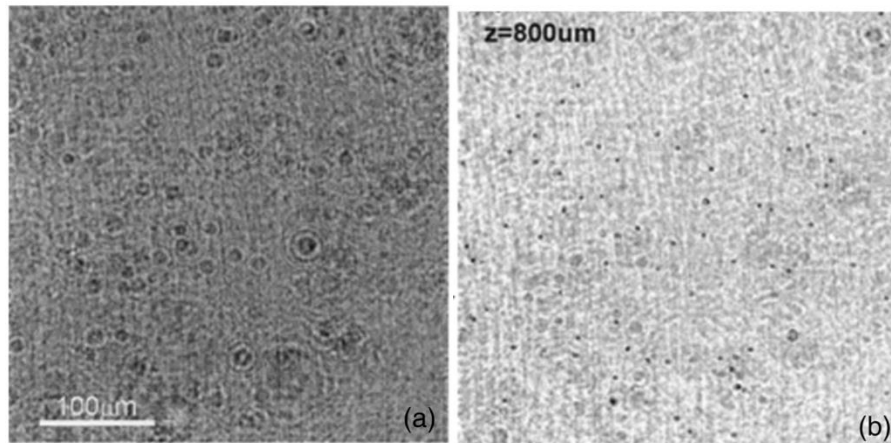


Figure 4.46. Part of a raw holographic recording (a) and a reconstruction of the plane located 800  $\mu\text{m}$  from the hologram plane. The in-focus particles appear as dark spots on the bright background.<sup>146</sup>

Digital reconstruction is performed by approximating the optical field as a diffraction pattern that propagates from the hologram plane. The 3D field reconstruction is generated by

$$\tilde{U}_r(x, y, z) = \iint_{x_h, y_h} \tilde{U}_r(\xi, \eta, z=0) \left[ -\frac{\partial G}{\partial n}(x-\xi, y-\eta, z) \right] d\xi d\eta, \quad (4.45)$$

where  $\frac{\partial G}{\partial n}$  represents the normal derivative of the Green's function of the wave equation propagating in a homogeneous medium evaluated at the aperture plane,  $z = 0$ . This term represents the diffraction of the tracer particles and is used for the 2D convolution integral with the hologram. The most common approximation of this diffraction term is the Rayleigh-Sommerfeld formula

$$-\frac{\partial G}{\partial n} = (x, y, z) = \frac{1}{\lambda} \frac{\exp\left(-jk\sqrt{x^2 + y^2 + z^2}\right)}{\sqrt{x^2 + y^2 + z^2}} \cos \theta, \quad (4.46)$$

where  $\cos \theta = z/\sqrt{x^2 + y^2 + z^2}$  and is generally assumed to be roughly unity for simplicity,  $\lambda$  is the laser wavelength,  $j$  is the unit imaginary number, and  $k = 2\pi/\lambda$ . This equation can be simplified using binomial expansion and a paraxial approximation, which gives the Kirchhoff-Fresnel expression

$$-\frac{\partial G}{\partial n} = (x, y, z) = \frac{\exp(-jkz)}{j\lambda z} \exp\left[j\frac{k}{2z}(x^2 + y^2)\right]. \quad (4.47)$$

The 3D reconstruction is most commonly simplified by analysis in the frequency domain, such that

$$\tilde{U}_r(x, y, z) = \mathcal{F}^{-1} \left[ \mathcal{F} \left\{ \tilde{U}_r(\xi, \eta, z=0) \right\} \mathcal{F} \left\{ \frac{\partial G}{\partial n} \right\} \right], \quad (4.48)$$

where  $\mathcal{F}$  represents the Fourier transform. The reconstruction is performed for each  $z$ -plane throughout the imaged volume. The image at a certain point is given by the irradiance field,  $\tilde{U}_r \tilde{U}_r^*(x, y, z)$ .<sup>145</sup>

After volumetric reconstruction, voxel elements contain images of particles, which are elongated in the depth direction. This effect is commonly called the depth of field problem and is

caused by the effective aperture angle, which can be limited by camera resolution, the angle of light diffraction, or the camera size. The elongation,  $\tau$ , of a particle image is

$$\tau = \frac{\lambda}{\theta^2}, \quad (4.49)$$

where  $\lambda$  is the wavelength of the coherent light source and  $\theta$  is the aperture half-angle. When using no magnification, the large pixel size of a digital camera can cause uncertainty in the location of interference fringes. The effective aperture angle of a resolution-limited hologram is

$$\theta = \tan^{-1} \left( \frac{\lambda}{2\Delta x} \right), \quad (4.50)$$

where  $\Delta x$  is the sensor pixel size. Magnification causes this resolution-limit to decrease, causing a hologram to be limited by light diffraction. Assuming Mie scattering, the effective aperture angle is given by

$$\theta = \frac{\lambda}{d}, \quad (4.51)$$

where  $d$  is the particle diameter. This aperture angle is unaffected by the magnification, thus magnification should be chosen such that a hologram is diffraction-limited to minimize the depth-of-field. If magnification is further increased, a hologram can become size-limited, meaning that the field-of-view of a camera limits the depth-resolution. The size-limited aperture angle is

$$\theta = \tan^{-1} \left( \frac{N\Delta x}{2MZ} \right), \quad (4.52)$$

where  $N$  is the number of pixels of size  $\Delta x$ ,  $M$  is the magnification, and  $Z$  is the distance from the object plane to the hologram.<sup>151</sup>

A simple technique for extracting the  $z$ -coordinate of an elongated particle image is by finding the depth plane containing the maximum voxel intensity value for each particle image cluster. In the case of two local maxima, the mean depth of the two planes provides a best estimate.<sup>152</sup> The in-plane particle center can then be determined using standard particle identification algorithms (discussed in Chapter 3) in the identified depth plane. Satake *et al.*<sup>153</sup> used this method of 3D particle location determination and performed an uncertainty analysis for single-camera micro holographic PTV using spherical nylon particles fixed on a glass plate. The glass plate was translated and the measured displacements were compared to the known displacement. The in-plane and out-of-plane displacement errors were 1.8% and 11.0% of the mean velocity, respectively. These values are equivalent to displacement errors of 0.400 pixels in-plane and 2.48 pixels out-of-plane.

Due to the inaccuracy involved in depth determination techniques, many experimentalists use a stereoscopic arrangement in order to give the same accuracy for in-plane and out-of-plane position measurements.<sup>152,154</sup> Stereoscopic pairing of the particle images in the two cameras is done using a nearest neighbor approach using the calculated 3D coordinates. Stereoscopic techniques require a careful calibration between the two cameras in order to achieve good results. Calibration can be done using targets with known locations or using an experimental image itself. Lu *et al.*<sup>152</sup> suggested using the relaxation method (discussed in Section 5.3) to determine correspondences to be used for calibration. Using these particle pairs, an affine transform from the coordinates of camera 2 to the coordinates of camera 1 can be determined. Thus all analysis can be done using the coordinate system of camera 1.

Buchmann *et al.*<sup>151</sup> proposed using a stereoscopic holographic setup and using tomographic reconstruction. This was done by independently reconstructing the volume imaged by both

holograms, thresholding to remove background noise, then calculating the product of voxel intensities from both reconstructions. This causes any voxel with zero intensity from either hologram to become zero in the product. Thus, the uncertainty in the depth coordinate is reduced to that of the in-plane uncertainty. The error estimate for this algorithm was 0.23 pixels.

One of the more commonly used techniques of extracting 3D particle positions using a single-camera setup is the deconvolution method, which can be applied instantaneously or iteratively to refocus a hologram to reconstruct a volume.<sup>155-160</sup> The instant 3D deconvolution is appropriate for the reconstruction of small tracer particles, and thus for PTV. The reconstructed object,  $O$ , is obtained from the deconvolution

$$O = \mathcal{F}^{-1} \left[ \frac{\mathcal{F} \left( |\tilde{U}_r(x, y, z)|^2 \right)}{\mathcal{F} \left( |U_p|^2 + \beta \right)} \right], \quad (4.53)$$

where  $|\tilde{U}_r(x, y, z)|^2$  is the reconstructed intensity of the object wave from convolution,  $|U_p|^2$  is the reconstructed intensity of an arbitrary single point scatter,  $\beta$  is a small constant to avoid dividing by infinity, and  $\mathcal{F}$  is the 3D Fourier transform.  $\beta$  was set to unity in its implementation by Latychevskaia *et al.*<sup>159</sup> This deconvolution yields a set of sharp localizations, or ideally delta functions, that make up the particle field that was imaged. Latychevskaia and Fink<sup>160</sup> showed that reconstructed particle length is reduced to only 2 to 3 pixels after deconvolution.

Focus functions, which calculate the variance and Laplacian of reconstructed voxel intensities in each plane, were introduced as a method for determining the plane in which a particle is in focus.<sup>161,162</sup> These techniques were still limited to selecting the reconstructed plane with the most in-focus particle image as a depth coordinate, thus Seo and Lee<sup>163</sup> proposed that the cross

correlation of the focus functions of a tracked particle in two different frames can be used to refine the out-of-plane displacement. The cross correlation function is defined as

$$F_1 \circ F_2 = \int F_1(z)F_2(z + \Delta z)dz, \quad (4.54)$$

where  $F_1$  and  $F_2$  are the focus functions in the first and second frames, respectively. Applying this algorithm to two holograms with a constant displacement of 100  $\mu\text{m}$ , the standard application of the focus function was compared to the cross-correlation method. The standard deviation of measured displacement was reduced from 4.8  $\mu\text{m}$  to 1.6  $\mu\text{m}$  by applying the proposed cross-correlation method.

An examination of the effect of particle concentration on holographic reconstruction was performed by Kim and Lee.<sup>164</sup> A turbulent jet with  $Re=1,200$  was imaged using a PCO SensiCam CCD with 1,280 x 1,024 pixels in a measurement volume of 8.5 x 6.8 x 9  $\text{mm}^3$ . The reconstruction efficiency,  $\varphi_c$ , was defined as the ratio of reconstructed particles divided by the number of seeded particles and the spatial resolution was defined as the mean particles spacing. These two values were compared for tracer concentrations varying from 2 to 25 particles/ $\text{mm}^3$  (0.0008 to 0.0099 ppp) in Figure 4.47. The reconstruction efficiency between 13 and 17 particles/ $\text{mm}^3$  was roughly constant at 76%. Thus the particle density corresponding to the highest spatial resolution was 17 particles/ $\text{mm}^3$  (0.0067 ppp). This seeding density resulted in 0.0051 reconstructed ppp.

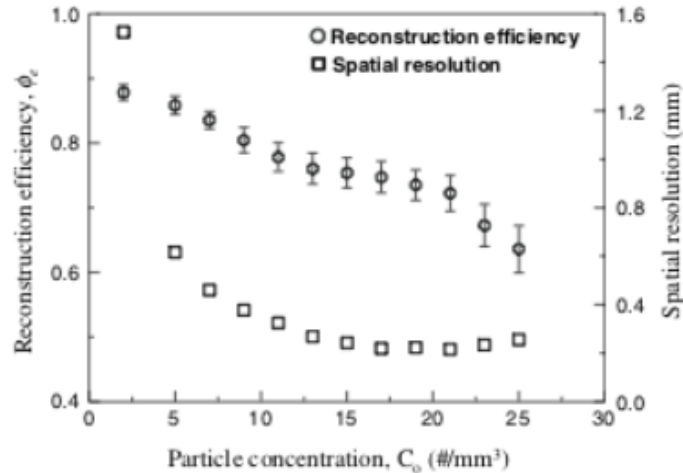


Figure 4.47. Variation of reconstruction efficiency and spatial resolution for a range of particle concentrations.<sup>164</sup>

Toloui and Hong<sup>149</sup> proposed an iterative reconstruction technique for identifying particles in holograms called inverse iterative particle extraction (IIPE). A deconvolution is performed and particle centroids are determined as in other algorithms, then the identified particles are removed from the hologram through back-propagation. This allows particles that were hidden by the fringe patterns of identified particles to be identified. After hologram modification, the volume reconstruction, deconvolution, and centroid identification are recalculated as normal. The iterative algorithm was compared to non-iterative algorithms using synthetic data generated from 0.0008 to 0.0024 ppp. While both algorithms reconstructed more than 90% of the correct particles for the lowest seeding density tested, the iterative algorithm and non-iterative algorithms correctly reconstructed 97% and 40% of particles, respectively, for 0.0024 ppp. IIPE was further tested, successfully reconstructing at least 90% of particles up to 0.003 ppp.

Toloui *et al.*<sup>165</sup> further improved the IIPE algorithm by using variable window local thresholding, longitudinal wall position detection, multi-pass tracking, and continuity-based displacement correction. The local thresholding algorithm initially uses a window that is six times the size of the particle diameter, reducing the window size by one particle diameter after each

iteration of particle extraction. By using this technique with IIPE, the number of extractable tracers was increased by 25% for micro scales and 80% for large volumes. Wall-detection was done by identifying the depth with the greatest particle concentration gradient. These improvements were applied to turbulent channel flow experiment with a volume of  $14.7 \times 50.0 \times 14.4 \text{ mm}^3$ . The extracted particle density was as high as 0.0037 ppp. A hologram that was processed using IIPE is shown in Figure 4.48. Seven iterations of particle extraction left no clear particle images in the residual hologram.

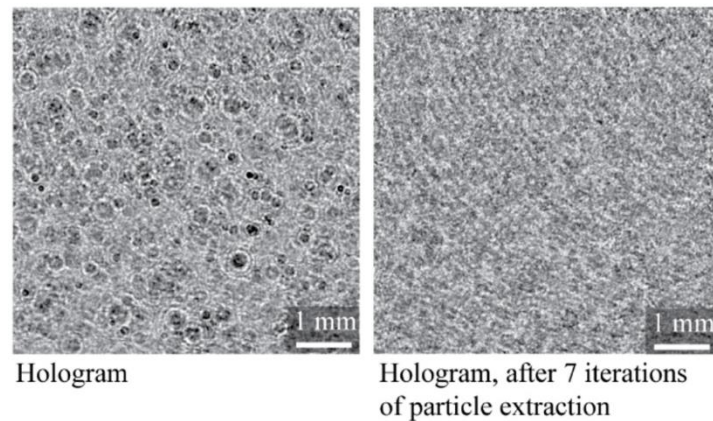


Figure 4.48. Original hologram (left) and residual hologram after using the IIPE algorithm (right).<sup>165</sup>

### 4.3 OPTICAL DISTORTIONS AND CALIBRATION IN PTV SYSTEMS

In PTV systems, optical distortions exist between the image and object spaces. This causes imaging to deviate substantially from the pinhole model that is often used. Distortions can come from camera misalignment, lens imperfections and fluid-window interfaces. This section discusses how to model and compensate for these optical distortions using calibration for a single-camera 2D experiment, a planar 3D experiment, and a volumetric 3D experiment.

### 4.3.1 2D Calibration

In order to obtain meaningful physical results from PTV, an accurate relationship between the image coordinates and the object coordinates is necessary. A standard 2D experiment uses a single camera to image particles within an illuminated sheet, which is typically one to two mm thick. In this case, a simplification can be made, assuming that all imaged particles have zero out-of-plane displacement from the laser light sheet. Therefore, the map from image to spatial coordinates is

$$\mathbf{X} = \mathbf{F}(\mathbf{x}), \quad (4.55)$$

where  $\mathbf{X}$  and  $\mathbf{x}$  are the image and object coordinates, respectively. This map cannot, however, be known exactly, thus an estimate must be made. Soloff *et al.*<sup>166</sup> suggest that polynomial fit is sufficient in most cases and can be fitted in a least squares sense. The authors also suggest that a third-order polynomial in each of the in-plane dimensions is sufficient for all but the most severe cases of distortion. Thus, the 2D calibration polynomial is

$$\hat{\mathbf{F}}(\mathbf{x}) = a_0 + a_1x + a_2y + a_3x^2 + a_4xy + a_5y^2 + a_6x^3 + a_7x^2y + a_8xy^2 + a_9y^3, \quad (4.56)$$

where  $a_i$  are two-component vector-valued coefficients that are determined via calibration.

Calibration is done using a target which has regularly spaced characteristic markings that cover the entire view of the camera. The characteristic markings are most often circular and must be evenly spaced on a Cartesian grid. Alternatively, Kent *et al.*<sup>167</sup> used a single target on the end of an arm attached to a stepper motor, which was able to be moved precisely to many different locations. Before beginning the calibration procedure, coordinate systems for the image and object planes must be defined. The origin within the object coordinates is specified as the location of one specific characteristic point. All other characteristic marks are measured with respect to this

position. The image plane origin is generally defined as the (0,0) pixel located in the lower-left corner of the CCD.

The first step in calibration is to acquire a calibration image. For 2D PTV, a single image recorded with the calibration target aligned precisely at the center of the light sheet. Using the acquired image, characteristic markings are then identified using correlation-based template-matching. The correlation coefficient is determined and a sub-pixel peak is identified by fitting a Gaussian or polynomial curve to the peak. Using the relative positions of these characteristic marks and their corresponding image coordinates, the coefficients of the mapping polynomial can be determined using a least-squares procedure.

Soloff *et al.* applied this calibration procedure using a 12 x 12 Cartesian grid of black circles on a white background. The circles had a diameter of 1mm and the center-to-center spacing was 5.07 mm. The RMS calibration error was 0.54 pixels, or 0.069 mm, showing that the centroid estimation error was 7% of the particle image diameter. Prasad *et al.*<sup>168</sup> suggested that the centroid estimation error is typically 5-10% of the particle image diameter, demonstrating that this calibration procedure was comparable to the error contribution of centroid estimation.

#### 4.3.2 *Stereoscopic Calibration*

Stereo-PTV experiments generally utilize two viewing angles to determine out-of-plane displacements in a thin volume. It is important to note that some two-camera PTV setups make measurements in a thicker observation volume, the calibration of which should be executed using techniques discussed in Section 4.3.3. In order to estimate this third component of velocity, the calibration must be performed using multiple images of a calibration target at different depths.

Soloff *et al.*<sup>166</sup> extended their proposed 2D calibration algorithm to work for planar stereoscopic measurements.

In order to calibrate a stereoscopic system, the mapping function needs to be extended to include three-dimensions. The authors proposed that for a small depth, it is sufficient to only have quadratic dependence in the  $z$ -direction in the calibration polynomial, therefore suggesting the polynomial

$$\hat{\mathbf{F}}(\mathbf{x}) = a_0 + a_1x + a_2y + a_3z + a_4x^2 + a_5xy + a_6y^2 + a_7xz + a_8yz + a_9z^2 + a_{10}x^3 + a_{11}x^2y + a_{12}xy^2 + a_{13}y^3 + a_{14}x^2z + a_{15}xyz + a_{16}y^2z + a_{17}xz^2 + a_{18}yz^2. \quad (4.57)$$

The quadratic dependence of  $z$  requires a minimum of three calibration images at different depths, although more calibration images can be used to reduce the mean squared error of the estimate. After image coordinates of all of the characteristic marks in the calibration images have been determined, the calibration polynomial coefficients can be determined for each of the two cameras separately.

Soloff *et al.*<sup>166</sup> tested their calibration technique using a target made of a rectangular block of aluminum with 0.5 mm holes drilled in a 9 x 9 Cartesian grid. The center-to-center spacing was 9 mm between holes. The calibration images were taken of the backlit plate positioned at the center of the light sheet and with a 0.5 mm offset in both directions. The RMS errors for the mapping function were 1.1 and 1.2 pixels, or 0.045 and 0.051 mm, for cameras one and two, respectively. The out-of-plane displacement estimates based on a PIV interrogation technique were less than 0.02 mm for a measurement depth of 1 mm.

Wieneke<sup>169</sup> introduced a calibration technique for stereoscopic imaging that use ray tracing corrected for distortions (see Section 4.2.1.2) and does not require the calibration plate to be

aligned with the laser sheet. This allows for calibration to be performed even if the observation volume cannot be easily accessed. In addition, this approach uses tracer particle images rather than images of a calibration target, and is therefore called the self-calibration technique. The first step of the self-calibration technique is to determine a 3D mapping between image and object coordinates, either using the technique suggested by Soloff *et al.*<sup>166</sup> or the ray-tracing procedure, for example, the distortion-corrected ray-tracing proposed by Wieneke (Section 4.2.1.2). The second step is to compute the disparity vector map of the two dewarped images collected by each of the two cameras. This is done using standard cross-correlation. In order to average out the effect of depth, between 5 and 50 image pairs should be used in an ensemble-averaging algorithm. A multi-pass algorithm can use this initial disparity vector map to distort the cross-correlation interrogation windows, which can then be used for another cross-correlation to further refine the disparity estimate. Once this disparity map has been calculated, corresponding object coordinates are determined for each vector using triangulation. The disparity vectors mean that the two reprojected lines from each camera do not intersect at a single point in the object plane. Therefore, triangulation must be used to determine the point in space that minimizes the disparity between what was measured in camera one and camera two. This triangulation using disparity vectors is shown in Figure 4.49.

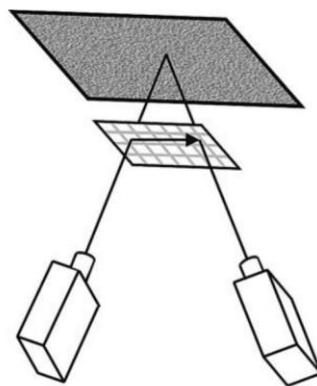


Figure 4.49. The object points of the light sheet determined using a disparity vector.<sup>169</sup>

Once the triangulation has been performed for each disparity vector, a plane is then fitted to the world points in 3D space and the mapping functions of both cameras are corrected by a corresponding transformation such that the new measurement plane becomes the  $z = 0$  plane. The translation and rotation of the new measurement plane are finally applied to the original mapping functions for both cameras. The whole process can be repeated again to arrive at better fits, and Wieneke suggested that using single-pass cross-correlation and repeating the entire correction procedure two or three times converges to gives good results.

It was experimentally shown that this self-calibration procedure was able to place the  $z = 0$  plane within 0.1 pixels of the center of the light sheet. It is advisable to use this procedure to check for the proper alignment of the calibration plate. It was also proposed that calibration could be performed for a closed measurement volume using self-calibration. The cameras must be placed on a traverse such that they can move away from the measurement volume. They are focused on the measurement plane and traversed backward until a calibration target mounted on the front side of the measurement volume is in focus. The initial calibration is done in air and the ray-tracing technique is then used to correct for the distorting media. For this step, the distance between the calibration target and the laser sheet must be known. Self-calibration then corrects further disparities when the cameras have been moved back to observe the measurement plane. Using stereo-PIV, out-of-plane displacements were measured to within 2% for externally calibrated systems using self-calibration. This approach has not been extended into PTV analyses, therefore it is unknown how this uncertainty carries into PTV results.

The *in-situ* calibration technique was used by Choi and Guezennec<sup>170</sup> to calibrate stereo-PTV systems that use a wide-angle separation between cameras to image a deep volume. It was proposed that four polynomials should be fitted to fully describe the relationship between image

and object coordinates for two cameras. For camera two, a particle image located at  $(x_2, y_2)$  corresponds to world coordinates

$$x = f_2(x_2, y_2, y), \quad (4.58)$$

$$z = g_2(x_2, y_2, y), \quad (4.59)$$

where  $y$  is a free variable and  $f_2$  and  $g_2$  are polynomial functions. The corresponding relations for camera one are

$$y = f_1(x_1, y_1, x), \quad (4.60)$$

$$z = g_1(x_1, y_1, x), \quad (4.61)$$

where  $x$  is a free variable and  $f_1$  and  $g_1$  are polynomial functions. The coefficients of the polynomial functions can be obtained using calibration points and least-squares curve-fitting. The authors suggested that a second-order fit in all three variables was sufficient to achieve good model accuracy. This calibration technique was similarly shown to outperform ray-tracing when geometrical-optical parameters were not known exactly.

### 4.3.3 3D Calibration

In the context of 3D PTV, an accurate calibration procedure plays an important role not only for minimizing errors, but also for reconstructing true particles without the presence of ghosts. 3D PTV techniques require a mapping function between a camera's image space and the object space. While photogrammetric techniques define the object space in terms of physical coordinates, tomographic, plenoptic, and synthetic aperture systems define the object space in discrete voxels.

The volumetric calibration procedure is largely the same as that of a thin sheet, except that the calibration target needs to be imaged throughout the entire volume for the most accurate results.

While the standard calibration target is either a metal sheet with machined holes or dark dots on a white background, Schosser *et al.*<sup>171</sup> showed that a non-intrusive calibration can be performed by creating calibration points using a laser. The proposed technique uses a continuous wave laser with a focused beam, perpendicularly aligned to the optical access window to illuminate the measurement volume at known  $x$  and  $y$  locations. Whenever the light impacts a surface, it induces light reflections, which can be imaged and used as calibration points. The reflections from the front and rear boundaries of the measurement volume can be used for the calibration, as shown in Figure 4.50. This non-intrusive method of obtaining calibration images within a Tesla turbine rotor demonstrates the usefulness of this technique for performing 3D velocimetry when measurement volume access is not feasible, such as in turbomachinery, cavity flows, and combustion.

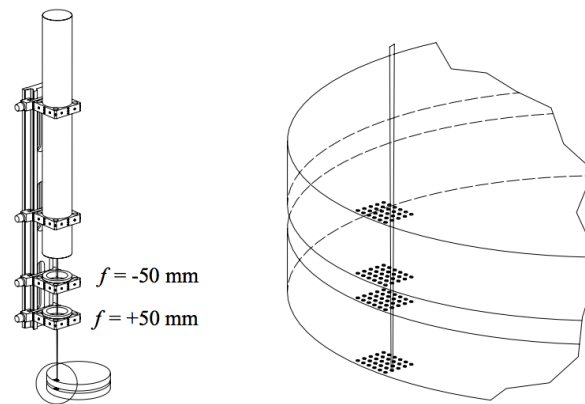


Figure 4.50. Calibration points obtained using laser reflection from measurement volume boundaries.<sup>171</sup>

The calibration points are imaged on each camera and used to calculate the parameters of the mapping function. There are generally two different types of mapping function that can be selected: one that models the refractive interfaces for a given experimental setup, and a general one that aims to account for all potential sources of distortion. A common way to model an experimental setup is to use the pinhole model and correct it for distortions. The equations

describing the transformation from image to epipolar line are given in Section 4.2.1.2. The unknowns can be found by performing a least squares fit to the calibration images.<sup>24</sup>

Maas<sup>172</sup> suggested using a correction to model for refractive index changes in the line of sight for particle tracking experiments. More recently Mulsow<sup>173</sup> incorporated a camera system model with refraction using ray tracing. This approach was applied to planar interfaces and cylindrical vessels by Belden.<sup>134</sup> Generally speaking, ray tracing states that a point in space in the direction of ray  $\mathbf{r}$  is given by

$$\mathbf{X}(t) = \mathbf{X}_0 + t\hat{\mathbf{r}}, \quad (4.62)$$

where  $\mathbf{X}_0$  is a known point on the line and  $t$  is a parameter that determines the location along the line. The unit vector that defines the direction of  $\mathbf{r}$ , the refracted ray, is given by

$$\hat{\mathbf{r}}_t = n_{it}\hat{\mathbf{r}}_i + \left[ -n_{it}\hat{\mathbf{N}} \cdot \hat{\mathbf{r}}_i - \sqrt{1 - n_{it}^2 \left[ 1 - (\hat{\mathbf{N}} \cdot \hat{\mathbf{r}}_i)^2 \right]} \right] \hat{\mathbf{N}} \quad (4.63)$$

where  $n_{it}$  is the ratio of refractive indices of the incident and transmitted media,  $\hat{\mathbf{N}}$  is the unit normal of the refractive interface, and  $\hat{\mathbf{r}}_i$  is the initial ray direction. Using these refraction equations, the alternating forward ray tracing (AFRT) procedure, shown in Figure 4.51, begins by assuming a straight ray from the camera to the object and iteratively determining the correspondence between object and image coordinates. Due to the direct modeling of the refraction, the standard deviation of position estimates in a simulation was found to be a length equivalent to 0.1 pixels.

A model-free calibration method, first suggested by Kent and Trigui,<sup>174</sup> suggests using an optical transform with coefficients determined by *in situ* calibration instead of ray tracing techniques, which require accurate knowledge of the geometry of the experimental setup. In their

setup, Kent and Trigui show that their calibration technique outperforms ray tracing. To show this, an experimental setup with 2 cameras oriented at 90 degrees to one another was simulated. Camera 1 was given various azimuthal offsets as well as a 5 degree offset in elevation. The comparison of Kent and Trigui's *in situ* calibration technique and traditional ray tracing for this camera misalignment and zero 2D error is shown in Figure 4.52. The errors for *in situ* calibration are two orders of magnitude less than those of ray tracing, suggesting that camera misalignment is not a major source of error for the calibration technique. As a result, errors from 2D localization cause the most substantial uncertainty for this method.

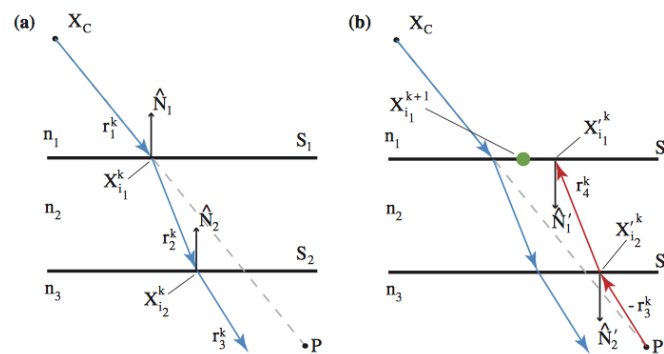


Figure 4.51. Schematic of the AFRT procedure used by Belden.  $P$  is the point of interest and  $X_C$  is the image coordinate.<sup>134</sup>

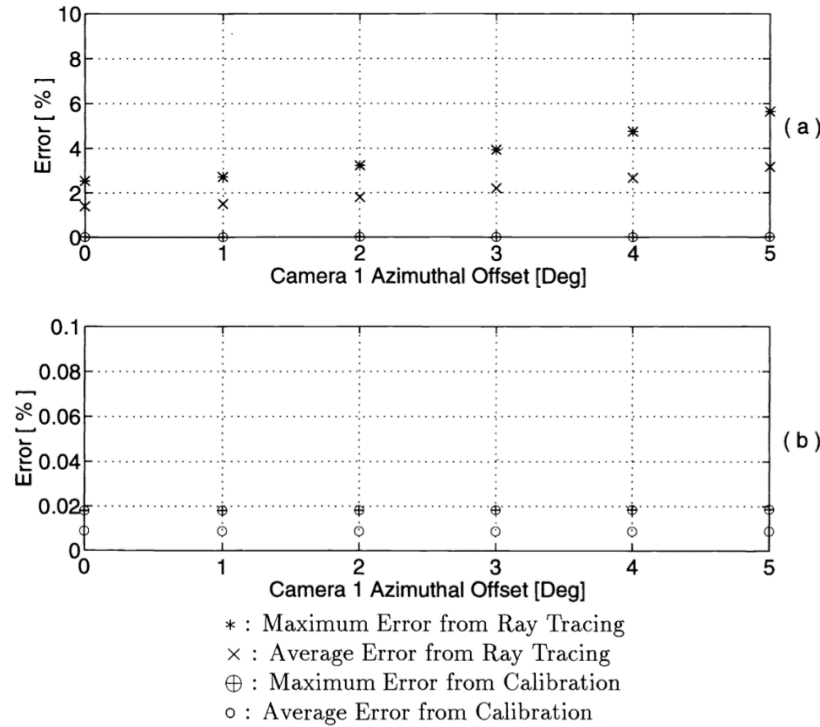


Figure 4.52. 3-D positioning errors from camera azimuthal misalignment.<sup>174</sup>

This model-free calibration technique was also used by Tien *et al.*<sup>175</sup> for calibration in the presence of nonhomogeneous media and optical imperfections. In the calibration process, a transformation matrix between image coordinates and object coordinates is found for each  $z$  coordinate used. Then for each pixel in the camera, the spatial  $x$  and  $y$  coordinates for each  $z$  coordinate are calculated using this transformation. The technique uses radial basis functions to approximate the mapping function. The calibration error estimates from these experiments in the context of defocusing micro-PIV are below 4 microns (1.5% of calibration range) throughout the measurement volume. The in-plane RMS errors are less than 0.16 microns (0.05%) and the out of plane errors are around 0.81 microns (0.25%).

Yoon *et al.*<sup>98,176</sup> found that in the context of defocusing micro-PIV, experimental calibration is necessary for determining the depth of a particle. A relationship between the depth calibration coefficient,  $\partial z / \partial D$ , and the fluid refractive index,  $RI$ , was presented by the authors, which allowed

for a single calibration to be used for a wide range of fluids given the same experimental setup. The depth calibration coefficient gives an estimate of the particle distance from the focal plane using the diameter of the circle connecting the three defocused particle images. The proposed estimate of the calibration coefficient is

$$\left(\frac{\partial z}{\partial D}\right)_{fluid} \approx \frac{n_{fluid}}{n_{ref}} \left(\frac{\partial z}{\partial D}\right)_{ref}, \quad (4.64)$$

where  $n_{fluid}$  and  $n_{ref}$  are the refractive indices of an experimental fluid and a reference fluid, respectively. This is an approximate relationship that is valid for small angles of refraction. Applying this technique experimentally, the calibration coefficient only had discrepancies on the order of 1% from the directly measured coefficient.

An additional volume self-calibration (VSC) step can be included using only the collected triangulation data. Weineke<sup>177</sup> suggested that the residual triangulation error can be used to correct the mapping functions for each camera. 2D particle image positions,  $x_i$  and  $y_i$ , are mapped to their best fit 3D spatial coordinates. Then these 3D coordinates are mapped back to image coordinates,  $x'_i$  and  $y'_i$ , and a disparity,  $d_i$ , exists for real data. Typical self-calibration techniques adjust their mapping function,  $M_i$ , by this disparity distribution,

$$d_i = (x'_i, y'_i) - (x_i, y_i), \quad (4.65)$$

$$M'_i(X, Y, Z) = M_i(X, Y, Z) - d_i(X, Y, Z). \quad (4.66)$$

This technique works well for low particle seeding densities when particle correspondences can be matched with a high confidence level and few ghost particles. Weineke proposed that one can apply a threshold such that only the brightest 10% of particle images are considered. After thresholding, the volume is subdivided such that multiple particle images exist in each. The

disparity is then plotted for all particles in a sub-volume for each camera and the peak is selected. The disparities are then smoothed throughout the entire volume and used to correct the mapping function of each camera. The procedure can be repeated for further improvement and validation. In the context of tomographic PIV with 0.1 ppp, the maximum calibration errors were shown to be reduced from 1-2 to 0.1-0.2 pixels in some areas. This error is due to the disparity from the VSC triangulation step, and thus carries into PTV analyses as well. Application of this technique to 3D PTV can reduce the number of ghost particles, allowing for increased seeding densities.<sup>177</sup>

Discetti and Astarita<sup>178</sup> investigated the performance of this self-calibration algorithm and determined that including four or more cameras in the procedure can significantly affect the quality of the disparity maps due to the contribution of spurious matches. It was proposed that VSC should first be performed on three cameras, then again for each additional camera using the first three as a reference. This method is recommended for cases with high particle seeding densities and a large number of cameras. It was shown with simulated data at 0.05 ppp that the modified self-calibration algorithm can improve the signal-to-noise ratio, particularly when a search radius greater than five pixels is required for triangulation and more than four cameras are used.

Another calibration approach used by tomo-PTV and IPR is the non-uniform optical transfer function (OTF).<sup>179</sup> The technique uses a spatially variable correction for distorted particle image shapes in order to improve reconstruction results. The OTF algorithm relies upon the VSC results for 3D coordinates of particles and the corresponding image coordinates on each camera. These images are each characterized in terms of the 2D Gaussian peak-fitter,

$$W(x, y) = p^* \exp \left( -\frac{1}{2} \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} a & b \\ b & c \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \right), \quad (4.67)$$

where  $p$ ,  $a$ ,  $b$ , and  $c$  describe the shape and peak height of a particle image. These image characteristics are collected and separated into sub-volumes. Within these sub-volumes the image characteristics are averaged for each camera, allowing for an optical transformation that varies in space. The volume can be discretized into any sized sub-volumes in order to best account for spatial variations.

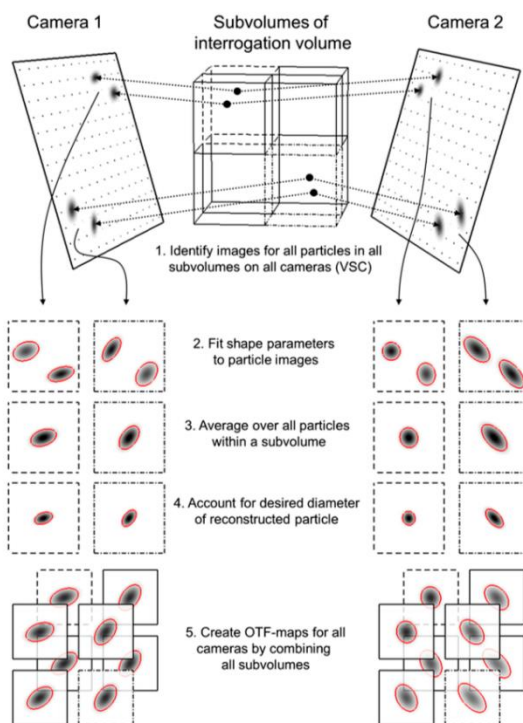


Figure 4.53. Steps required for an OTF calibration.<sup>179</sup>

Both a simulation and an experimental test of the OTF calibration were performed. For both perfect and astigmatic data, SMART reconstruction was performed using OTF, a bilinear interpolation, and a B-spline for weighting functions. These were all compared to a MART algorithm with no optical calibration, and in all cases tested up to 0.1 ppp, where the OTF outperformed the others in terms of reconstruction quality,  $q$ , as shown in Figure 4.54.

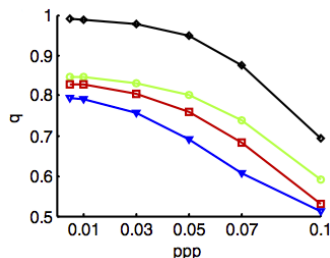


Figure 4.54. Comparison of reconstruction quality for OTF (black) B-spline (green) bilinear interpolation (red) and MART (blue) using synthetic astigmatic data.<sup>179</sup>

## Chapter 5. PARTICLE TRACKING TECHNIQUES

Once particle spatial locations are determined (Chapter 4) the final step in PTV algorithms is tracking particles to determine the velocity field. There are many robust algorithms for matching particles from one frame to the next, which is necessary for properly calculating flow velocity. These algorithms can largely be used in both two- and three-dimensional problems once particle positions have been mapped to the spatial domain. Each section explains the extension of algorithms into the third dimension, which simply requires the addition of a z-coordinate to position vectors in many cases. In this section the following tracking algorithms will be discussed: multi-frame approaches for particle tracking using time-resolved PTV data (Section 5.1); cross-correlation methods for tracking individual particles (Section 5.2); relaxation methods for iteratively selecting probable matches based on the displacement of nearby particles (Section 5.3); tracking algorithms based on various forms of neural networks (Section 5.4); tracking that utilizes the velocity gradient tensor to identify strong deformations (Section 5.5); tracking based on the similarity of nearby particle patterns using polar coordinate similarities (Section 5.6); optimization methods, which employ various iterative minimization techniques for determining particle tracks (Section 5.7); tracking that finds matching using Delaunay tessellation and Voronoi diagrams to track particles (Sections 5.8 & 5.9); and vision-based tracking methods, which match particles

using the principles of proximity, similarity, and exclusion (Section 5.10). Techniques for identifying outlier particle tracks are discussed (Section 5.11), and finally methods that combine tracking and localization techniques are discussed (Section 5.12).

## 5.1 MULTI-FRAME APPROACH

Multi-frame tracking techniques are among the most common for PTV, as they use temporal information to identify a particle's trajectory over many time steps. The techniques use between two and four frames, which must be recorded at a regular time interval. This requirement for time-resolved imaging limits the applicability of these multi-frame techniques to either low-speed flows or high frame-rate cameras.

Hassan and Cnaan<sup>180</sup> introduced a particle tracking approach that works for singly exposed frames. The algorithm uses a four-frame sequence of time-resolved images to track a particle by determining the path that minimizes the variance in the particle's trajectory and velocity. Equations 5.1-5.5 calculate the total variance in a particle's path.

$$\sigma_l = \sqrt{\frac{1}{3} \left[ |d_{ij} - d_m|^2 + |d_{jk} - d_m|^2 + |d_{kl} - d_m|^2 \right]}, \quad (5.1)$$

$$d_m = \frac{1}{3} (d_{ij} + d_{jk} + d_{kl}), \quad (5.2)$$

$$\sigma_\theta = \sqrt{\frac{1}{2} \left[ |\theta_{ik} - \theta_m|^2 + |\theta_{jl} - \theta_m|^2 \right]}, \quad (5.3)$$

$$\theta_m = \frac{1}{2} (\theta_{ik} + \theta_{jl}), \quad (5.4)$$

$$\sigma_t = \sqrt{\frac{\sigma_l^2}{d_m^2} + \sigma_\theta^2}. \quad (5.5)$$

$d_{ij}$  is the distance between particle images  $i$  and  $j$ , potential matches between frames.  $\theta_{ik}$  is the angle between the segments  $d_{ij}$  and  $d_{jk}$ ,  $\sigma_l$  is the standard deviation of the track length,  $\sigma_\theta$  is the standard deviation of the segment angles, and  $\sigma_t$  is the total standard deviation, This method makes no assumptions about the movement of neighboring particles, and does not exclude particles sharing a match in a frame.

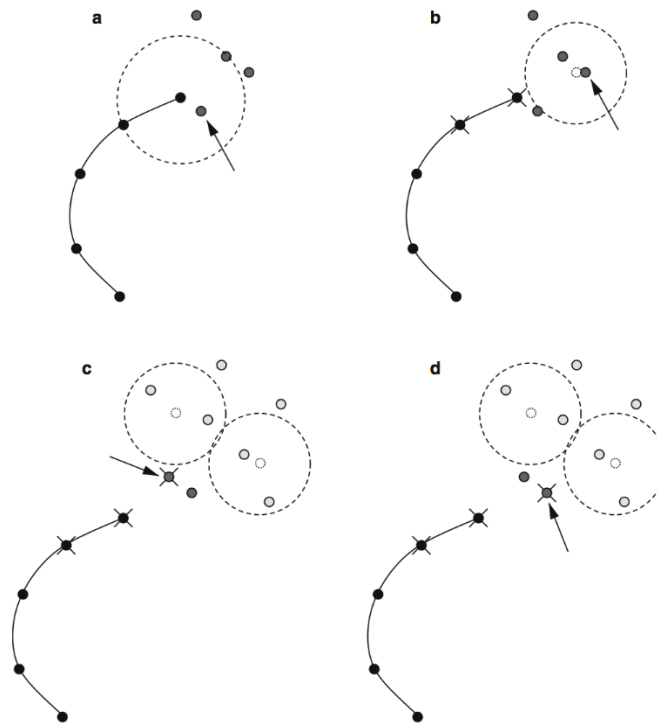


Figure 5.1. The heuristics use for multi-frame particle tracking algorithms: a) nearest neighbor, b) three frame minimum acceleration, c) four frame minimum change in acceleration, and d) four frame best estimate. In each case the arrow points to the particle image that will be selected<sup>181</sup>.

Malik *et al.*<sup>182</sup> introduced both the 3-frame minimum acceleration (3MA) and 4-frame minimum change in acceleration techniques (4MA) for 3D tracking. The 3MA technique uses a time sequence of three frames to determine the most likely path of a particle based on minimizing acceleration while the 4MA approach is based on minimizing the change in acceleration along a

particle path. Both of these minimizations are performed based on triangulated 3D positions, although tracking can be similarly performed in 2D. These techniques require particle tracks for previous time steps, meaning that an initialization is required. It was suggested that this be done using a nearest neighbor heuristic (shown in Figure 5.1a). A cost function,  $\varphi_{ij}^n$ , is defined as the distance between particle  $i$  in frame  $n$  and potential match  $j$  in frame  $n+1$ :

$$\varphi_{ij}^n = \|x_j^{n+1} - x_i^n\|, \quad (5.6)$$

where the pair  $i, j$  corresponding to the minimum  $\varphi$  is selected as a match. This nearest neighbor criterion can be improved by using an initial estimate of local mean flow velocity either from known flow relations or nearby tracks. Once an initial track consisting of two points has been obtained, the 3MA criterion (shown in Figure 5.1b) can be used. The acceleration is calculated for each three-point track

$$\varphi_{ij}^n = \frac{\|x_j^{n+1} - 2x_i^n + x_i^{n-1}\|}{2\Delta t^2}, \quad (5.7)$$

where  $\Delta t$  is the time elapsed between frames. The 4MA algorithm (shown in Figure 5.1c) expands upon this by minimizing the change in acceleration once a three-point track has been established

$$\varphi_{ij}^n = \frac{\|x_j^{n+2} - 2x_i^{n+1} + x_i^n\| - \|x_j^{n+1} - 2x_i^n + x_i^{n-1}\|}{2\Delta t^2}, \quad (5.8)$$

Ouellette *et al.*<sup>181</sup> introduced a modification called the four-frame best estimate (4BE) approach (shown in Figure 5.1d). Instead of minimizing the change in acceleration, the difference between the projected fourth particle position,  $\tilde{x}_i^{n+2}$ , and the actual fourth particle position,  $x_j^{n+2}$  is minimized

$$\varphi_{ij}^n = \|x_j^{n+2} - \tilde{x}_i^{n+2}\|. \quad (5.9)$$

The projected position is based on the known positions and velocities at the first three points assuming constant acceleration. A conflict arises when a candidate particle is considered a best match for two reference particles in the most recent frame, requiring either starting a new track with the new particle, or using a cost minimization function. If the cost minimization approach is used, it is calculated using the Munkres algorithm. Figure 5.2 compares three- and four-frame algorithms using one of these two approaches.  $E_{track}$  is the ratio of the number of tracks that were imperfectly recreated to the total number of tracks and  $\zeta$  is the ratio of average particle displacement to the average particle spacing. Figure 5.2 also shows that the 4BE algorithm outperforms the nearest neighbor (NN), 3-frame minimum acceleration, and 4-frame minimum change in acceleration techniques. Additionally, the Munkres algorithm degraded the performance of each algorithm except the nearest neighbor.

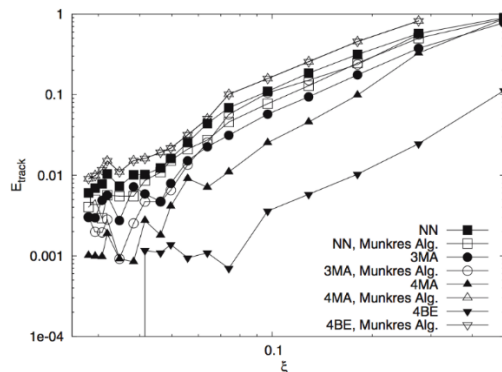


Figure 5.2. Comparison of Nearest Neighbor (NN) method, 3 frame minimum acceleration (3MA), 4 frame minimum change in acceleration (4MA), and 4 frame best estimate algorithms(4BE)<sup>181</sup>

Tarlet *et al.*<sup>183</sup> introduced a three-frame technique that minimizes acceleration for images of colored particles. In using color recognition software, the minimum acceleration approach is capable of identifying particle matches in increased seeding densities using color discrimination. It was found that the use of three different colored tracer particles effectively decreased  $\zeta$  by 39%.

Although the particle density was not given for this comparison, the performance of these multi-frame algorithms depends on  $\zeta$ , which would not only affect the average particle spacing, but also the average particle displacement through  $\zeta$ , which according to Figure 5.2, should be kept less than 0.2 if  $E_{track}$  is to be no larger than 0.01. For example in a 2D application, for 0.03 ppp, the average particle spacing is  $\sim 5.8$  pixels, suggesting that the average particle displacement should be no more than 1.2 pixels.

## 5.2 CROSS-CORRELATION

A common technique for matching particles in two-frame PTV analysis uses the cross-correlation method.<sup>184-186</sup> It is based on the principle that a fluid particle and its closest neighbor particles typically move in quasi rigid-body motion. This knowledge is used to determine the probability of a candidate particle in the first frame being a match with a candidate particle in the second. Figure 5.3 shows a visual representation of this method.

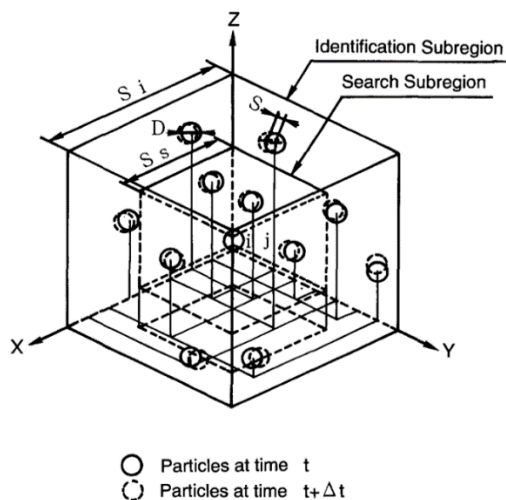


Figure 5.3. Particle positions from consecutive PTV frames with reference particle  $i$  overlapped with candidate particle  $j$ .<sup>184</sup>

In the cross correlation algorithm proposed by Yamamoto *et al.*<sup>184</sup> and Hassan *et al.*<sup>185</sup>, analysis was performed on a processed image pair, in which particle images have been identified and binarized in order to only use the particle positions rather than pixel intensities. This technique is called binary cross-correlation (BCC), and has been implemented in 3D by placing synthetic spherical particles at the location of each triangulated particle location.<sup>184</sup> In the BCC method, a sub-region is defined around the reference particle,  $i$ , and each candidate particle,  $j$ , with  $N_i$  and  $N_j$  neighbor particles, respectively. The neighbor within the superimposed sub-regions that have overlapping spheres are considered. A parameter,  $S$  is defined as the ratio norm of the 3D distance between the  $N$  overlapping particle centroids in the two sub-regions and the synthetic spherical particle diameter. Then the correlation coefficient for each candidate particle is calculated as

$$C_{ij} = \frac{\sum_{k=1}^N 1 - \frac{3}{2}S_k + \frac{1}{2}S_k^3}{\sqrt{N_i N_j}}. \quad (5.10)$$

This correlation coefficient is equivalent to the calculation of the total overlapped volume between the particles in the reference and candidate sub-regions.

A different cross-correlation algorithm was proposed by Saga *et al.*<sup>186</sup> and does not use the binary image data. Instead, the image gray values are used so as to allow CCD values to be used directly. The sub-regions search zones are defined similarly to the implementation by Yamamoto *et al.*<sup>184</sup>, but the correlation windows were varied in size in order to decrease computation time. Initially, correlation was performed using an 8x8 pixel window. The correlation coefficient,  $R$ , is calculated as

$$R = \frac{\sum_M \sum_N (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_M \sum_N (A_{mn} - \bar{A})^2\right)\left(\sum_M \sum_N (B_{mn} - \bar{B})^2\right)}}, \quad (5.11)$$

where  $A$  and  $B$  are the gray value matrices for the reference and second subregions, their mean values being  $\bar{A}$  and  $\bar{B}$ . For candidate particles with a correlation coefficient above 0.6, a second correlation is performed using a 16x16 window. The threshold was then increased to 0.75. After another iteration or two, the candidate particle with the maximum correlation coefficient was selected. This cross-correlation algorithm was compared to the BCC algorithm using the VSJ standard image with 1997 identifiable synthetic particles (0.03 ppp) generated using LES simulation of an impinging jet flow. The results of this comparison are given in Table 5-1.

Table 5-1: Comparison of BCC and cross-correlation algorithm proposed by Saga et al. using VSJ standard image 01.<sup>186</sup>

Method	Particles Tracked	Valid Particle Tracks	Valid vpp
BCC	917	881	0.013
Saga et al. <sup>186</sup>	1592	1573	0.024

The BCC method works for irrotational flows with small velocity gradients. The motions of rotational flows and flows with large velocity gradients prevent the cross-correlation algorithm from correctly recognizing neighbor particle motion.

### 5.3 RELAXATION METHODS

Similar to the BCC method, the relaxation method matches a particle by comparing its motion to that of its neighbors. The difference, however, is that in the relaxation method neighbor particles

need not exactly overlap. This allows the algorithm to adjust for velocity gradients and rotation within the flow.

Relaxation methods were introduced by Rosenfeld *et al.*<sup>187</sup> for labeling features in computer vision. The introduction of the technique to PTV was the result of work done by Wu and Pairman<sup>188</sup> followed by its application to turbulent flows by Baek and Lee<sup>189</sup>. Ohmi and Li<sup>61</sup> further modified the method of Baek and Lee, although, Pereira *et al.*<sup>190</sup> states that these modifications were proven to yield only minor improvements in their 3D implementation. Brevis *et al.*<sup>191</sup> showed that combining cross-correlation with relaxation could reduce computation time and improve results for inhomogeneous seeding densities. Jia *et al.* implemented a bidirectional computation of the relaxation method, which improved the handling of no-match particles.

The most important aspect of the relaxation algorithm is that the analysis is based on the particle match probability between the first and second frames, defined for every potential pair of particles inclusive of the probability of there being a loss-of-pair. Typically, a potential match is any candidate particle within a specified maximum displacement distance of the reference particle. The particle match probabilities are updated iteratively based on the match probabilities and relative motions of neighboring particles. The no-match probability accounts for particles that leave the observation volume from one frame to the next. As relaxation algorithms are iterative, the match probabilities must be initialized. Figure 5.4 shows the initialization method and its formulation is

$$P_{ij(i)}^{(0)} = P_i^{*(0)} = \frac{1}{n_i + 1}, \quad (5.12)$$

where  $P_{ij}$  is the probability that the reference particle  $i$  is a match with candidate particle  $j$ .  $P_i^*$  is the no-match probability of particle  $i$  and  $n_i$  is the number of candidate particles within the

maximum displacement distance of particle  $i$ . Match probabilities can alternatively be initialized using results from prior computations, such as cross-correlation. This typically shortens the computation time of the relaxation method.

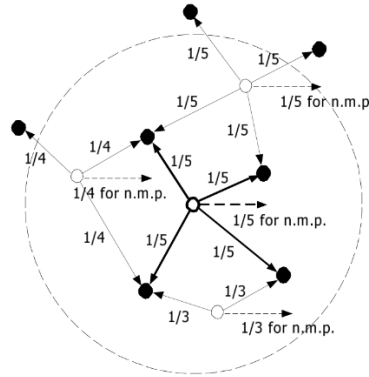


Figure 5.4. Initialization of match probabilities for reference particles in relaxation method.<sup>61</sup>

After initializing the match probabilities, the match probabilities are updated for each iteration,  $n$ . In order to update the match probability between a reference particle,  $j$ , and a candidate particle,  $i$ , the contribution from the match probability of each neighboring particle,  $k$ , and its candidate particle,  $l$ , is

$$\tilde{P}_{ij}^{(n)} = P_{ij}^{(n-1)} \left( A + B \sum_k \sum_l P_{kl}^{(n-1)} \right), \quad (5.13)$$

where  $A$  and  $B$  are constants in the equations, set to be 0.3 and 3.0, respectively, as suggested by Barnard and Thompson.<sup>192</sup> The superscripts signify that the probabilities from iteration  $n-1$  are used to generate the probabilities for iteration  $n$ . The contribution of neighboring particles in the double summation above is limited to the neighboring matches that that satisfy

$$|d_{ij} - d_{kl}| < R_c, \quad (5.14)$$

where  $d_{ij}$  is the displacement vector from particle  $i$  to particle  $j$  and  $d_{kl}$  is that from particle  $k$  to particle  $l$ . A threshold,  $R_c$ , is the radius of the relaxation area, which allows some deviation from parallel motion. The updated probabilities are then normalized using

$$P_{ij}^{(n)} = \frac{\tilde{P}_{ij}^{(n)}}{\sum_j \tilde{P}_{ij}^{(n)} + P_i^{*(n-1)}}, \quad (5.15)$$

which uses the no match probability from the preceding iteration. The no-match probability is not updated, but just renormalized as

$$P_i^{*(n)} = \frac{P_i^{*(n)}}{\sum_j \tilde{P}_{ij}^{(n)} + P_i^{*(n-1)}}. \quad (5.16)$$

This update scheme is iterated 10-20 times, ideally resulting in correct match probabilities near unity and others near zero. A threshold for the maximum difference between iterations is defined to determine when the relaxation algorithm has converged. The highest match probability (or no-match probability) is then selected the correct track for each candidate particle.

Ohmi and Li<sup>61</sup> suggested two changes to this algorithm. It was pointed out that the update procedure for the no-match probability lacked consistency and could only increase if all other matching probabilities thoroughly decreased, which is rare in densely seeded flows. In order to overcome this, a new scheme for updating the no-match probability was suggested as

$$\tilde{P}_i^{*(n)} = \sum_{m < C} \frac{DP_i^{*(n-1)}}{n_i}, \quad (5.17)$$

where  $C$  and  $D$  are constants, equal to 0.1 and 5.0, respectively, and  $m$  is the ratio of the number of interparticle links satisfying Equation 5.14. To the number of candidate particles from all the neighboring particles. This causes the no-match probability to increase at a constant rate if too few

neighbor particles appear to undergo quasi-parallel motion. After this update, the normalization step needs to be modified such that

$$P_{ij}^{(n)} = \frac{\tilde{P}_{ij}^{(n)}}{\sum_j \tilde{P}_{ij}^{(n)} + \tilde{P}_i^{*(n)}}, \quad (5.18)$$

$$P_i^{*(n)} = \frac{\tilde{P}_i^{*(n)}}{\sum_j \tilde{P}_{ij}^{(n)} + \tilde{P}_i^{*(n)}}. \quad (5.19)$$

The second change that the threshold for  $R_c$  should vary with local flow velocity. Ohmi and Li<sup>61</sup> suggested that low-speed flow regions will produce clusters of quasi-parallel, but clearly erroneous vectors if the same threshold is used. Thus, this local threshold is

$$R_c = E + F|x_i - y_j|, \quad (5.20)$$

where  $E$  and  $F$  are constants and  $y_j$  is the position of the candidate particle.  $E$  is image dependent and typically in the range of 1.0 - 4.0.  $F$  is a proportionality constant that can be fixed to 0.05. Despite the seemingly small contribution of this proportionality constant, it reduces the number of spurious vectors in strong shear flows.

To assess the performance of this algorithm, tests were performed using the VSJ PIV standard image 301, which has 4000 synthetic particle images with 0.06 ppp simulating the vertical portion of an impinging jet flow. It is important to note that not all of the 4000 particle images were identifiable and the number of identified particle images was not reported. Table 5-2 gives the results of the comparison of three tracking techniques. It can be seen that the relaxation method outperforms the BCC method in particle matching. Also, the improvements made by Ohmi and Li<sup>61</sup> provided an improvement over the original relaxation algorithm both in number of particle

tracks obtained and in reliability. The resulting density of valid vectors obtained using the improved relaxation method was 0.012 vpp.

Table 5-2: Comparison of four particle tracking algorithms using the PIV standard image 301.<sup>61</sup>

Algorithm	Number of particles matched	Correct matching rate (%)
Four Frame	779	86%
Binary Cross Correlation	860	91%
Original Relaxation	786	96%
Ohmi and Li Relaxation	808	98%

Brevis *et al.*<sup>191</sup> introduced an algorithm that combines the strengths of cross-correlation (Section 5.2) and the relaxation methods. Figure 5.5 outlines how the two techniques are combined. Matches are initially found using cross-correlation, which has fast computation time and works well for low particle density, low gradient, irrotational flows. A cross-correlation threshold, double match filter, mean filter, and median filter (described in detail in Section 5.11) are then used to identify correct tracks from the cross-correlation analysis. Tracks pass the filter are assigned a match probability of one, essentially removing them from relaxation calculations. The remaining tracks then use the cross-correlation coefficients, renormalized using a no-match probability equal to the difference between one and the maximum correlation coefficient, as initial match probabilities for the relaxation algorithm. For these particles and regions in which cross-correlation fails to identify a match, the relaxation method matches the remaining particles. A double match filter and median filter are then applied to remove spurious vectors from the final results. Additionally, as the relaxation method starts with estimates for match probabilities, fewer iterations are required for convergence.

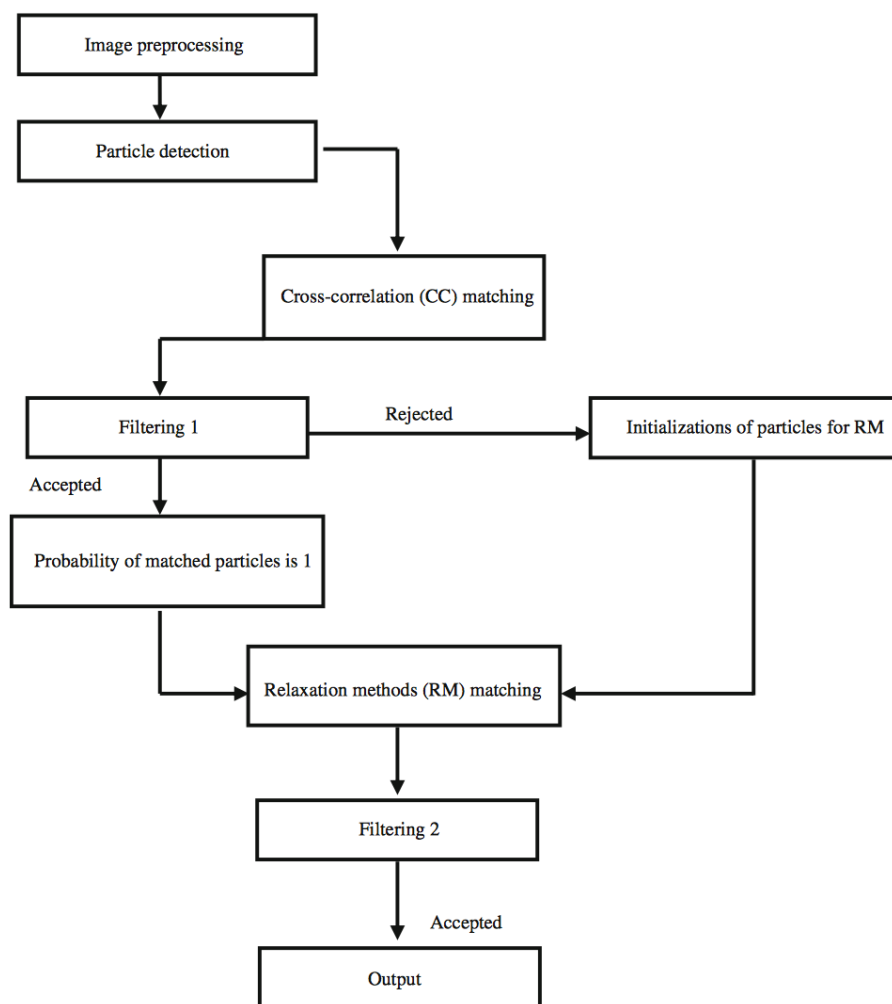


Figure 5.5. Flow chart for the Brevis *et al.* hybrid algorithm.<sup>191</sup>

Brevis *et al.*<sup>191</sup> compared their hybrid algorithm to the cross-correlation and relaxation methods using exact particle locations from VSJ standard image 301, which has 4042 possible vectors (0.062 vpp) simulating a jet impingement on a wall. The hybrid algorithm achieved 98.46% yield, which was roughly 5% greater than the cross-correlation method and 1% greater than the relaxation method. Although the hybrid algorithm performed only marginally better than the relaxation method, it was able to achieve its results several times faster as a result of the computational efficiency of cross-correlation. Additionally, the synthetic flow data had homogeneous particle density, meaning that there were no regions in which particles were scarce enough to give the cross-correlation method an advantage. The algorithm was also applied to

experimental time-resolved data at a seeding density of 0.0045 ppp. At this low density, particle tracks longer than 100 frames were extracted and are shown in Figure 5.6.

Further improvements to the relaxation method were made by Jia *et al.*<sup>193</sup> using Delaunay tessellations. The range for neighbor particles included in the relaxation parameter calculation is determined automatically rather than set to a range. Only particles with direct links to the target particle within the Delaunay tessellation are considered. Additionally, a method of bidirectional computation improves the handling of no match probability. First, no-match probability is directly ignored for calculation. Second, calculations are done for the match probabilities,  $P_{ij}$ , both from the first frame to the second and from the second frame to the first. Then the latter calculation is converted and merged with the former. Considering the particle tracks achieved from both forward and backward computations, final results are selected based if these tracks that do not conflict with one another. When conflicts arise, tracks that appear more often are selected, and ones that appear less often are discarded. If two tracks conflict and appear the same number of times, they are regarded as no-match particles.



Figure 5.6. Long particle tracks obtained using 3D implementation of the ICCRM algorithm.<sup>191</sup>

Results from the bidirectional algorithm are compared to those of the relaxation scheme of Ohmi and Li<sup>61</sup> at 0.046 ppp in Figure 5.7, where  $\eta$  is the fraction of correctly matched vectors,  $\mu_a$  is the ratio of randomly added particles to total particles, and  $\mu_r$  is the ratio of removed particles to total particles. The bidirectional algorithm showed improved performance in the case of removed particles. The computation time of the bidirectional scheme is twice that of the unidirectional one, however, the cost was considered acceptable due to the improvement in performance. Note that for the case of zero removed particles, the bidirectional algorithm achieved 0.046 vpp, as all possible vectors were identified correctly.

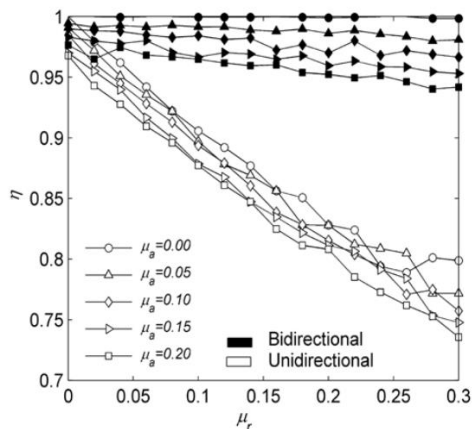


Figure 5.7. Comparison of the unidirectional and bidirectional algorithms in the presence of removed and added particles<sup>193</sup>

Jia *et al.*<sup>194</sup> also performed a 3D test of the bidirectional relaxation algorithm using four different VSJ 256 x 256 pixel standard images. The parameters of each are detailed in Table 5-3. The maximum particle densities of images 301C, 337, 351, and 377 are 0.064, 0.105, 0.033, and 0.015 ppp, respectively. The ratio of correctly identified tracks to available tracks,  $\eta$ , is plotted as a function of the ratio of erased particles to the total number of particles,  $\mu$ . For all images and ranges of  $\mu$ , the bidirectional algorithm performs substantially better than the unidirectional one. The highest resulting vector densities using the bidirectional algorithm for images 301C, 337, 351, and 377 are 0.058, 0.095, 0.031, and 0.015 vpp, respectively. The yield decreases as the ratio of erased particles increases for all seeding densities.

Table 5-3: VSJ 3D standard images used for comparison of unidirectional and bidirectional relaxation algorithms.<sup>194</sup>

Items	Serial number of flows			
	301C	337	351	377
Flow description	Slit light sheet	Stereo-PIV images	3 Angles, wall refraction	3 Angles, unknown wall refraction
Frame number	145	201	145	201
Max particle number	4,202	6,883	2,138	979
Min particle number	3,737	6,509	1,950	899
Max inter-frame particle difference	54	51	29	13
Min inter-frame particle difference	-40	-44	-31	-14
Max particle entering ratio (%)	5.61	3.13	3.49	2.64
Min particle entering ratio (%)	2.70	1.58	1.55	0.21
Max particle escaping ratio (%)	5.82	3.39	3.65	2.08
Min particle escaping ratio (%)	2.72	1.49	1.57	0.11

Inter-frame particle difference is defined as  $N_{P1} - N_{P2}$ ;  $N_{P1}$  and  $N_{P2}$  are particle number in frame  $t$  and frame  $t + \Delta t$ , respectively. The entering and escaping ratios are based on  $N_{P1}$  in a match.

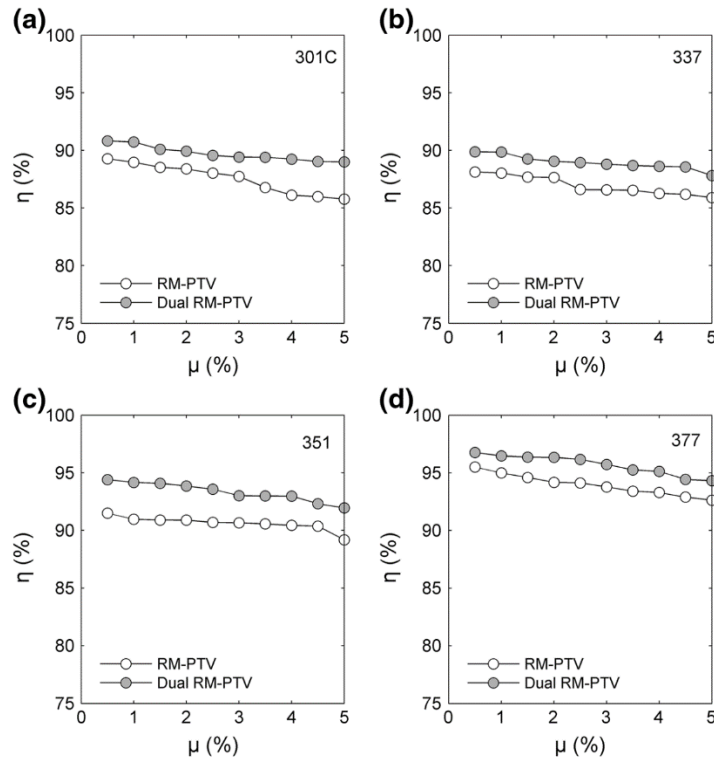


Figure 5.8. comparison of bidirectional (Dual RM-PTV) and unidirectional (RM-PTV) algorithms for four VSJ 3D standard images. <sup>194</sup>

In an improved technique called enhanced particle tracking velocimetry (EPTV)<sup>67</sup>, performance was improved by including a match probability initiation based on the diameter of the particle image. The initial match probability for reference  $i$  and candidate  $j$  is

$$P_{ij} = \alpha \left( 1 - \frac{|d_i - d_j|}{\Delta d_{\max}} \right) + (\alpha - 1) \left( 1 - \frac{|E_i - E_j|}{\Delta E_{\max}} \right), \quad (5.21)$$

$$E_i = \sum_{k,l \in S} I_{k,l}, \quad (5.22)$$

$$d_i = \sqrt{\frac{8E}{\pi I_0}}. \quad (5.23)$$

The values  $E_i$  and  $d_i$  are the total image intensity and image diameter, respectively. A particle image segment area,  $S$ , contains pixel intensities,  $I_{k,l}$ , with a peak value,  $I_0$ . The weight coefficient,  $\alpha$  can vary from 0 to 1 based on setup parameters. For instance, if the particle diameter is less than one pixel,  $\alpha$  should be set to zero so that the match probability is solely determined by the total image intensity.

The results for synthetic images with 0.062 ppp are compared in Table 5-4 for EPTV and other algorithms. Not only was the correct match rate improved over the variational approach, discussed in Section 5.7.2, and previous relaxation method, but the number of found vectors more than doubled to 0.026 vpp. The vast difference in yield between EPTV the other algorithms is due to the different particle identification algorithms used. The dynamic threshold binarization algorithm used by Mikheev and Zubtov identified twice as many particles as the simple threshold binarization of Ohmi and Li and the particle mask correlation of Runhau *et al.* Thus it is difficult to directly compare the performance of the algorithms.

Table 5-4: Comparison of EPTV on a VSJ standard image #301<sup>67</sup>.

PTV method	Frame pair	Number of vectors found ( $N_F$ )	Number of correct vectors ( $N_C$ )	Correct matching rate (%), $N_C/N_F$
True data	0-1	4042	4042	100
	0-2	3917	3917	100
EPTV	0-1	1759	1733	98.6
	0-2	1521	1505	98.9
NRX (Ohmi and Li 2000)	0-1	808	788	98
	0-2	714	680	95
VAR (Ruhnau <i>et al</i> 2005)	0-1	872	865	99
	0-2	904	885	98

## 5.4 NEURAL NETWORKS

A PTV algorithm based on neural networks was introduced by Grant and Pan.<sup>195</sup> Neural networks are commonly used in data science and are modeled based on the structure of the human brain, which is thought to consist of a three-dimensional matrix of interconnected neurons. This structure is capable of implementing simultaneous nonlinear processing strategies. Information is input to neurons, which modify and redistribute the information. Inputs into each neuron are summed, and if the net input exceeds a threshold, then the neuron is activated and transmits a signal to other neurons (shown in Figure 5.9 left). Weights or interconnection efficiencies determine the interaction of information between neurons. In both the human brain and in neural networks these weights can update to learn from input data sets. This characteristic of neural networks allows a system to learn from examples and adapt. The performance of a neural network largely depends on the availability of relevant examples. Neuron activations and weights continue to interact through layers until an output layer is reached (shown in Figure 5.9 right). The result from a neural network algorithm is an output label or classification, which is determined by identifying the neuron within the output layer with the greatest activation.

The neural network for particle tracking suggested by Grant and Pan (described by Labonte<sup>196</sup>) is a feed-forward network with three layers of neurons. The three layers are all the same size, square with  $2N_{max}+1$  nodes per side, where  $N_{max}$  is the integer pixel value of the maximum

displacement of a particle. A binary matrix is made for each of two consecutive PTV images, in which each element corresponds to a pixel. The value of a pixel is set to one if a particle image resides in the pixel, and it is otherwise equal to zero. The input into the neural network is a matrix,  $R$ , of the same size as the neuron layers. There is a single input matrix for each reference particle in the first PTV frame. The input matrix consists of a square sub-region of the binary matrix from the second PTV image centered at the coordinates of the reference particle. The first layer of the neural network takes the input matrix,  $R_{mn}$ , and weights,  $W_{mn}$ , to generate an output matrix

$$S_{mn} = H(R_{mn} W_{mn} - T_1), \quad (5.24)$$

where  $T_1$  is a threshold and  $H$  is the Heaviside function. The output  $S$  is then used as the input to the second layer of the neural network. The second layer outputs the product of  $S_{mn}$  and the neuron weights  $U_{mn}$ . The neuron with the maximum activation, which must also exceed a set threshold, wins and is set to one while the other neurons are set to zero. The winning neuron identifies the candidate particle that is part of the reference particle's track. This is illustrated in Figure 5.10.

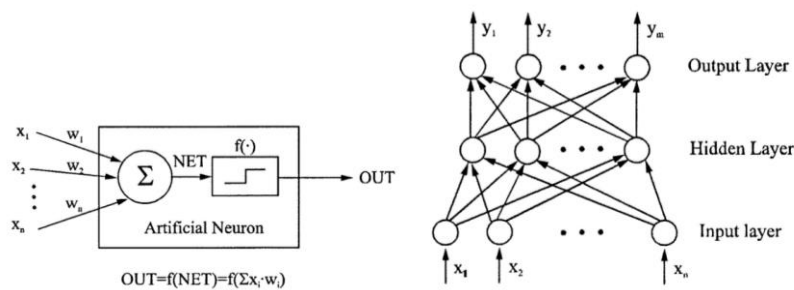


Figure 5.9. Inputs and activation of an artificial neuron (left) and structure of neurons in a feed-forward network (right).<sup>195</sup>

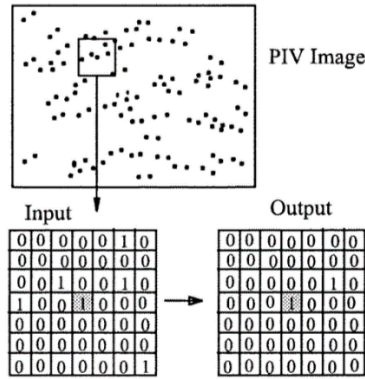


Figure 5.10. Example input and output of the neural network used by Grant and Pan.<sup>195</sup>

The neuron weights are only updated when the winner is identified with a one. If the winner's coordinates are  $(m^*, n^*)$ ,  $I_{mn}$  is the Euclidean distance between the winner and the reference particle,  $\rho_{mn} = (N_{\max} - I_{mn}) / N_{\max}$ , and  $\alpha$  is a constant between zero and one, then the weights are updated as

$$W_{mn} = \max \{ \rho_{mn}, W_{mn} \} \text{ for the first layer, and} \quad (5.25)$$

$$U_{mn} = U_{mn} + \alpha (\rho_{mn} - U_{mn}) \text{ for the second layer,} \quad (5.26)$$

which is then normalized as

$$U_{mn} = U_{mn} / U_{\max}, \quad (5.27)$$

where  $U_{\max}$  is the maximum weight in the second layer of the neural network. It was suggested that the inputs for spatially adjacent particles should be updated successively, for instance the candidate particles in a  $D_{\max} \times D_{\max}$  pixel square patch in the lower right hand corner were processed first. Then each square patch along the bottom was processed in a row. Then the next row of patches above that was processed, and so on. Weight initialization for this method requires statistical estimates of local flow conditions or a learning set of data that is correctly labeled.

A paper by Labonté<sup>196</sup> gives a different technique for applying neural networks to particle tracking. The method creates a self-organizing map (SOM) that constantly shifts particles based on the neighboring particle displacements. This is done to approximately reconstruct the flow of the fluid from the two pictures of tracer particles. Two sub-networks are created, each with one layer of neurons. Each neuron represents a particle image, thus the two subnets represent the two PTV images. The neurons have a value of  $x_i$  for particles in the first image and  $X_i$  for particles in the second. Corresponding weights are  $w_i$  for neurons in the first sub-network and  $W_i$  for neurons in the second sub-network. These weights are initialized as the position vector of the associated particle image.

In order to shift the particles such that they overlap, the weights between layers are compared. For neuron  $j$  in the first sub-network, the weights of all neurons in the second sub-network are compared to compute the Euclidian distance between the two coordinates

$$d_{ij} = \|W_i - w_j\|, \quad (5.28)$$

where the neuron  $i$  corresponding to the minimum  $d_{ij}$  activates neuron  $j$  if  $d_{ij}$  is less than the maximum allowed displacement. Otherwise the neuron is not activated. The winning neuron weight,  $W_i$ , and the candidate neuron weight,  $w_c$ , are then used to update the weight of all  $j$  neurons in the first sub-network as

$$\Delta w_j(W_i) = \alpha_{jc} (W_i - w_c), \quad (5.29)$$

$$\alpha_{jc} = \begin{cases} \alpha & \text{if } |w_c - w_j| \leq r \\ 0 & \text{if } |w_c - w_j| > r \end{cases}, \quad (5.30)$$

where  $\alpha$  is a constant between zero and one and  $r$  is a distance that defines which neighbor particles translate with the candidate. This update procedure influences particle tracks to move coherently with neighboring particle tracks.

Calculations are all performed independently for each neuron, then weights are all updated simultaneously for the first sub-network. The same procedure is then used to update the weights in the second sub-network. After this is completed, the neighborhood radius,  $r$ , is decreased and the weights are all updated again. This causes smaller and smaller groups of particles to move together with each iteration. The radius is decreased linearly until the two sets of weight vectors can be considered equal. The weight vectors can be considered equal if less than a user-defined tolerance. A larger tolerance allows for faster computation time.

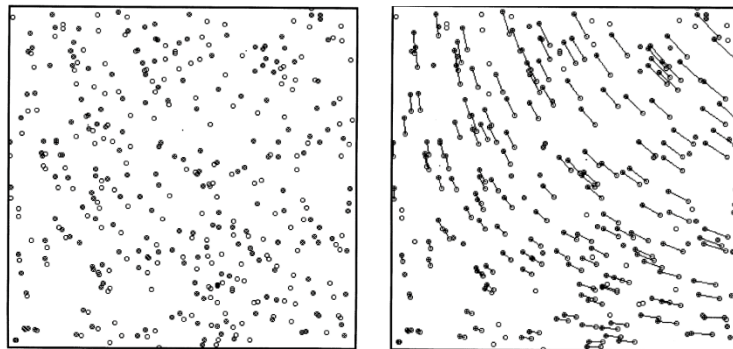


Figure 5.11. Visualization of a self-organizing map, in which particles from two frames are overlaid and organized until a match is found for each.<sup>196</sup>

Unlike Grant and Pan's neural network method, the self-organizing map requires no initialization or training. Both algorithms were tested with a point vortex flow with 200 images per picture, 40 of which were unmatchable. The ratio of mean particle displacement to particle spacing,  $\zeta$ , was 0.26. The image resolution was not given, so no particle density was reported. Grant and Pan's algorithm correctly matched 55% of the particles, while Labonté's algorithm achieved 78% success.

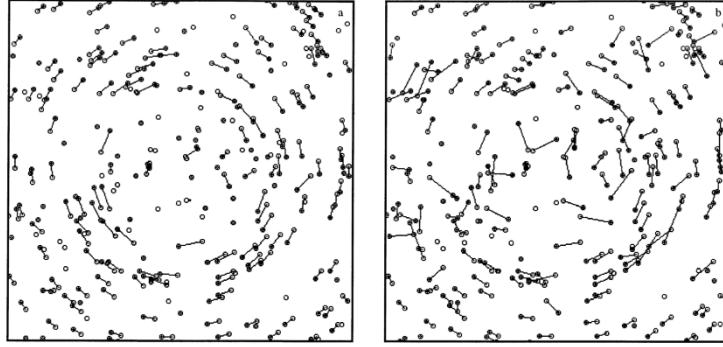


Figure 5.12. Displacement fields determined by the neural nets of Labonte (left) and Grant and Pan (right).<sup>196</sup>

Ohmi<sup>197</sup> suggested a modification to this self-organizing map that involves a more appropriate learning coefficient,  $\alpha_{jc}$  in Equation 5.29, which is based on a distance-dependent Gaussian function. The learning coefficient is often the same as before, but now neighboring particles are updated based on a continuous distance function rather than in a binary manner:

$$\alpha_{jc} = \begin{cases} \alpha & \text{if } |w_c - w_j| \leq r \\ \alpha \exp\left(-\frac{(|w_c - w_j| - r)^2}{2r^2}\right) & \text{if } |w_c - w_j| > r \end{cases} \quad (5.31)$$

A second modification deals with loss of pair particles which was borrowed from a solution to the traveling salesman problem (TSP) algorithm.<sup>198,199</sup> In their algorithm, the output layer neurons can be removed or doubled according to their computational history. In the context of this algorithm, a neuron that is not identified as a winner for ten consecutive iterations is removed. Output neurons that are identified as winners for multiple input neurons in three consecutive iterations are duplicated and treated as independent neurons for the remainder of the algorithm. This aims to improve convergence of the algorithm in the presence of loss-of-pair particles. The improvement of these changes is shown in Figure 5.13 using the VSJ standard image #01 with 0.033 ppp, where the new algorithm has many fewer spurious vectors than the original. It should be pointed out that

the author did not present quantitative measurements of the algorithm's improvement over the neural network of Labonté.

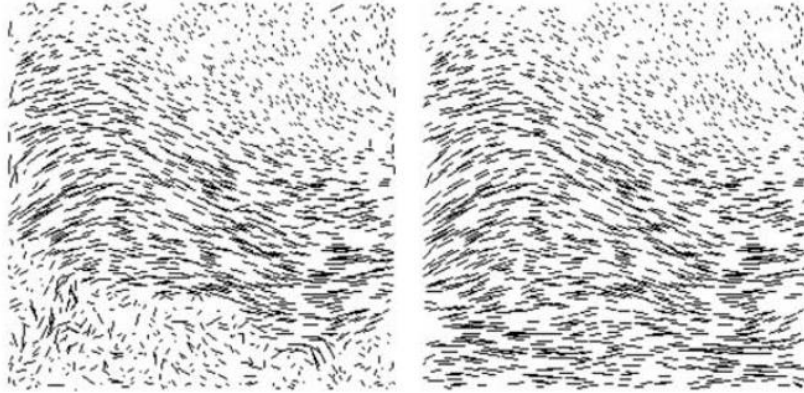


Figure 5.13. Comparison of the original SOM algorithm (left) and the modified SOM (right) for PIV VSJ standard image #01.<sup>197</sup>

The self-organizing map algorithms use particle position vectors as inputs, thus can be extended into three dimensions. Ohmi<sup>197</sup> extended his algorithm to work to 3D in this manner and applied it to VSJ standard image #351 to track 2500 particles, or 0.038 vpp; however, quantitative results for the yield and reliability were not reported.

## 5.5 VELOCITY GRADIENT TENSOR

Cross-correlation methods determine particle displacements best when particles move as non-rotating rigid bodies. As such, these techniques cannot always perform well when strong velocity gradients occur within a flow. The velocity gradient tensor (VGT) tracking method accounts for translation, rotation, and strain when determining particle tracks.<sup>200-202</sup>

Like other methods, the VGT technique chooses potential match particles  $J$  in the second frame for each particle  $I$  in the first frame. In doing so, a velocity  $\mathbf{u}(\mathbf{x}_I)$  is computed for a time step  $\Delta t$

$$\mathbf{x}_j = \mathbf{x}_I + \mathbf{u}(\mathbf{x}_I)\Delta t . \quad (5.32)$$

In the cross-correlation method, it is assumed that the neighbors of particle  $I$ ,  $x_{Ik}$ , translate with the same velocity to neighbors of particle  $J$ ,  $x_{Jk}$ . The VGT method, however, accounts for differences in velocity such that

$$\mathbf{x}_{jk} = \mathbf{x}_{ik} + \mathbf{u}(\mathbf{x}_{ik})\Delta t . \quad (5.33)$$

The velocity of each neighboring particle can be approximated using a first-order Taylor expansion around  $\mathbf{x}_I$

$$\mathbf{x}_{jk} = \mathbf{x}_j + (I + \delta\mathbf{u}(\mathbf{x}_I)\Delta t)(\mathbf{x}_{ik} - \mathbf{x}_I) , \quad (5.34)$$

where  $I$  is the identity matrix and  $\delta\mathbf{u}$  is the velocity gradient tensor. A transformation matrix,  $\mathbf{A}$  is then defined for simplicity. The velocity gradient tensor is incorporated as

$$\mathbf{A} = I + \delta\mathbf{u}(\mathbf{x}_I)\Delta t \quad (5.35)$$

For the local transformation in the neighborhood of particle  $I$ . The transformation is implemented for each of the  $n$  neighbors and a least squares minimization is performed as

$$E_{IJ} = \sum_{k=1}^n |\mathbf{X}_{J,k} - \mathbf{A}\mathbf{X}_{I,k}|^2 , \text{ where} \quad (5.36)$$

$$\mathbf{X}_{J,k} = \mathbf{x}_{jk} - \mathbf{x}_j . \quad (5.37)$$

Least squares minimization is performed for each combination and permutation of neighbors for particle  $I$  and  $J$ . The minimization is also performed for each possible candidate particle  $J$ . The particle pair  $(I,J)$  with minimum corresponding  $E_{IJ}$  is then used to finalize and compute a velocity vector. This procedure is outlined in the flow diagram in Figure 5.14.

To test this algorithm in comparison to others, a Karman vortex-shedding flow was simulated numerically with the 2D incompressible Navier-Stokes equation and a Reynolds equal to 500.<sup>202</sup> The VGT algorithm used eight neighbor particles for least squares minimizations. The displacement ratio,  $\Phi$ , denotes the ratio between the maximum particle displacement and the average distance between particles. The algorithms are compared for a range of displacement ratios and particle numbers in Figure 5.15. All of the tested algorithms performed well for displacement ratios below 2, and the performance of each began to degrade as displacement ratio increased further. The performance of VGT tracking was superior to both 4PTV and BICC methods for high displacement ratios. The authors pointed out that both 4PTV and BICC methods failed to correctly track particles at the highly rotational vortex cores. No pixel resolution was reported, thus no particle or vector concentrations could be ascertained. The results show that the tested tracking algorithms performed better at higher particle densities for a given displacement ratio.

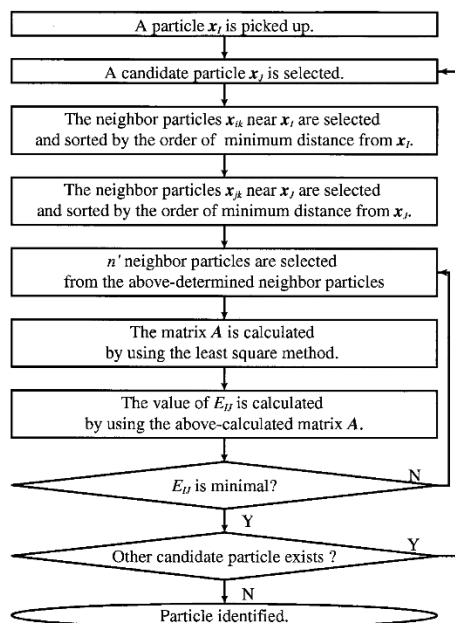


Figure 5.14. Flow chart for the pairing method.<sup>201</sup>

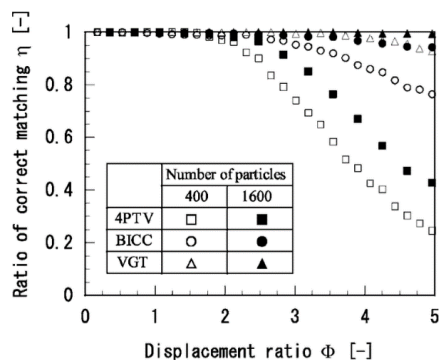


Figure 5.15. Performance comparison of four-frame minimum acceleration (4PTV), binary image cross-correlation (BICC), and velocity gradient tensor (VGT) tracking methods for Karman vortex-shedding flow.<sup>202</sup>

## 5.6 POLAR COORDINATE SIMILARITY

Ruan and Zhao<sup>203</sup> introduced a novel technique in 2005 that relates the similarities in polar coordinate systems surrounding candidate particles for matching. Shindler *et al.*<sup>71</sup> improved this polar coordinate system similarity (PCSS) method by including spatial adaptivity and temporal extrapolation. In order to implement this method, particle centroid locations must be known in either two or three dimensions, although the algorithm has not yet been applied to 3D data. Figure 5.16 shows how polar coordinates for surrounding particles are calculated with respect to reference particle,  $i$ , in the left frame and candidate particle,  $j$ , in the right.

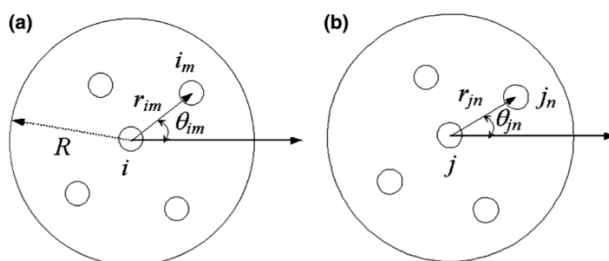


Figure 5.16. Concept of polar coordinate similarity between particles  $i$  and  $j$ .<sup>203</sup>

The relative polar coordinates of neighbors to particles  $i$  and  $j$  are compared and summed to generate a similarity coefficient

$$S_{ij} = \sum_{n=1}^N \sum_{m=1}^M H\left(\varepsilon_r - |r_{in} - r_{jn}|, \varepsilon_\theta - |\theta_{in} - \theta_{jn}|\right), \quad (5.38)$$

where  $H(x,y)$  is the Heaviside function and  $\varepsilon$  is the error allowed in radius coordinate and angle coordinate. In their implementation, the authors used  $\varepsilon_r < 5$  pixels and  $\varepsilon_\theta < 30^\circ$ . The candidate particle with the maximum similarity coefficient is selected as the match for each reference particle.

Table 5-5 compares this polar coordinate similarity (PCSS) tracking algorithm with others. The test was done using two frames of experimental data of flow induced by an oscillating plate with 0.0026 ppp. The PCSS algorithm yielded 0.0025 valid vpp. The number of vectors reported for BICC and 4-PTV algorithms included spurious vectors, thereby showing that the PCSS method outperformed the BICC and 4-PTV methods.

Table 5-5: Comparison of the number of calculated vectors using the polar coordinate similarity method (PCSS), cross correlation (BICC), and 4 frame in line tracking (4-PTV).<sup>203</sup>

Method	Vectors	Remark
PCSS	664	after delete spurious vectors
BICC	650	
4-PTV	604	need long computing time

The most substantial change made to the algorithm by Shindler *et al.*<sup>71</sup> was the adaptive interrogation window size. Originally, a fixed radius,  $R$ , was used to find neighboring particles; however, for flows with regions of sparsely seeded flow, there could be insufficient neighbors to generate a similarity coefficient. Thus, it was suggested that a fixed number of nearest neighbors should be selected. Temporal extrapolation was also suggested for multi-frame tracking with the PCSS algorithm. If data is time-resolved and a particular reference particle in frame  $k$  has a track from the previous frame,  $k-1$ , then a linear extrapolation can be performed to predict the particle's position in the next frame,  $k+1$ . If this is done, then the search radius and number of candidate

particles can be reduced. If a candidate particle's track spans from  $k-2$  to  $k$ , then extrapolation can be performed by assuming uniform acceleration. This can further reduce the search radius and number of candidate particles, which will result in lower computation time and increased probability of determining the correct track. An additional frame-gap technique was proposed, in which if a candidate particle in frame  $k$  does not find a match in frame  $k+1$ , then a search is performed in frame  $k+2$  with double the search radius and a lower threshold for similarity coefficient.

This algorithm was named enhanced polar-coordinate similarity, and in its two-frame implementation was called (2F-EPS). In the context of time-resolved data, a multi-frame enhanced polar-coordinate similarity (MF-EPS) algorithm used the velocity measurements from the first two frames to extrapolate a particle's future position. Table 5-6 shows both the performance enhancement of polar coordinate similarity algorithms over three-frame and relaxation approaches and the benefits realized from EPS and the multi-frame extension. The test was performed using VSJ standard image 301, and the particle centroids were determined using the optical flow equation and a 1D Gaussian estimator (see Section 28). The most substantial result is the improvement of MF-EPS using three images. Adding the third frame resulted in nearly triple the number of vectors, which is equivalent to 47% more vectors per time step. The resulting vector density was 0.026 valid vpp. The correct matching rate gives the ratio of correct vectors to detected vectors. Note that all algorithms, with the exception of 3FIT, achieved correct matching rates above 95%.

Table 5-6: Comparison of the number of calculated vectors and the number of valid vectors using polar coordinate similarity, 3-frame minimum acceleration (3FIT), and relaxation (NRX) methods using VSJ PIV standard image #301 (0.06 ppp).<sup>71</sup>

	Frames employed	Number of vectors ( $n_{vi}$ )	Number of correct vectors ( $n_{vci}$ )	Correct matching rate (%)
Original data	0-1	3917	3917	100.0
	0-2	3785	3785	100.0
	0-1-2	7816	7816	100.0
3FIT	0-1	954	896	93.9
	0-1-2	3038	2637	86.8
MF-EPS	0-1	1160	1146	98.8
	0-1-2	3448	3381	98.1
PCSS	0-1	1122	1104	98.4
	0-2	1019	973	95.5
NRX (Ohmi and Li 2000)	0-1	808	788	97.5
	0-2	714	680	95.2
2F-EPS	0-1	1123	1112	99.0
	0-2	1014	977	96.4

## 5.7 OPTIMIZATION METHODS

In this section, a number of tracking algorithms that employ optimization are discussed, which involve minimization of a cost function. The cost function can be as simple as a nearest neighbor criteria or as complicated as a minimization of velocity gradients within the flow. The techniques discussed include the deterministic annealing approach, the variational approach, genetic algorithms, ant colony optimization, and optimization using fuzzy logic.

### 5.7.1 *Deterministic Annealing*

The deterministic annealing approach is a method for matching particles that attempts to deal with rotational and shearing flows<sup>204</sup>. Essentially, a transformation of particle coordinates from the first frame to the second is determined based on the minimization of a cost function. The transformation contains both a displacement shift,  $t$ , and a mapping matrix,  $A$ , that can account for shear and rotation within an interrogation region, which in this implementation was a circular area with a 32 pixel diameter,

$$T_{\gamma}(X) = Y = t + AX . \quad (5.39)$$

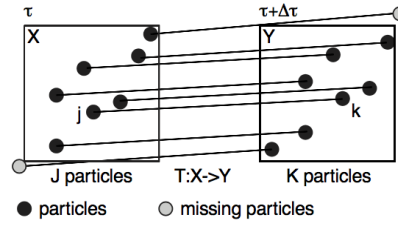


Figure 5.17. Schematic of mapping between frames X and Y.<sup>204</sup>

The transformation is used in the energy cost function

$$E(\mathbf{m}, y) = \sum_{k=1}^K \sum_{j=1}^J \alpha m_{kj} |Y_k - T_g(X_j)|^2 + g(g) + a \sum_{k=1}^K m_{k(J+1)} + a \sum_{j=1}^J m_{(K+1)j}, \quad (5.40)$$

where  $m_{kj}$  is a permutation matrix, which equals one when particle  $k$  is the selected match for particle  $j$ ,  $\alpha$  is a constant chosen to determine the maximum tolerable cost of a particle track, and  $g(\gamma)$  is a regularization term that can be omitted. The third and fourth terms on the right hand side of the cost function exist to account for the possibility of no-match particles. Within  $m_{kj}$ , a particle in the first frame without a match will be assigned  $K+1$  and a particle in the second frame will be assigned  $J+1$ .

Equation 5.40 is minimized iteratively to determine the appropriate transformation for each particle in the first frame. The pseudocode for this minimization is shown in Figure 5.18. There are two steps that are iterated. The first step calculates the probability of each match in matrix  $m_{kj}$  and normalizes them. Note that for the minimization, match matrix is not binary, but rather consists of probabilities that are normalized such that the sum equals one in each row and column. The second step recalculates the translation and mapping matrix. There is a parameter,  $\beta$  that is increased exponentially with each iteration, and allows for the convergence of match probabilities in  $m_{kj}$  to one. The parameter selection method is thoroughly described by Stellmacher and Obermeyer.<sup>204</sup>

```

BEGIN Outer loop
BEGIN Inner loop
E-Step:
  Calculation of the probabilities  $\langle m_{kj} \rangle$ :
  IF  $j \neq J + 1$  and  $k \neq K + 1$ :
     $\langle m_{kj} \rangle = \exp(-\beta |Y_k - T_j(X_j)|^2)$ 
  ELSE
     $\langle m_{kj} \rangle = \exp(-\beta \alpha)$ 
  Normalization of the probabilities  $\langle m_{kj} \rangle$ :
  BEGIN normalization loop
     $\langle m_{kj} \rangle = \frac{\langle m_{kj} \rangle}{\sum_{j'=1}^{J+1} \langle m_{kj'} \rangle}$ 
     $\langle m_{kj} \rangle = \frac{\langle m_{kj} \rangle}{\sum_{k'=1}^{K+1} \langle m_{k'j} \rangle}$ 
  END Normalization loop
M-Step:
  Calculation of transformation parameters:
   $\frac{\partial E}{\partial \gamma} \stackrel{!}{=} 0$ 
END Inner loop
  Increase  $\beta$ 
END Outer loop

```

Figure 5.18. Pseudocode for deterministic annealing minimization.<sup>204</sup>

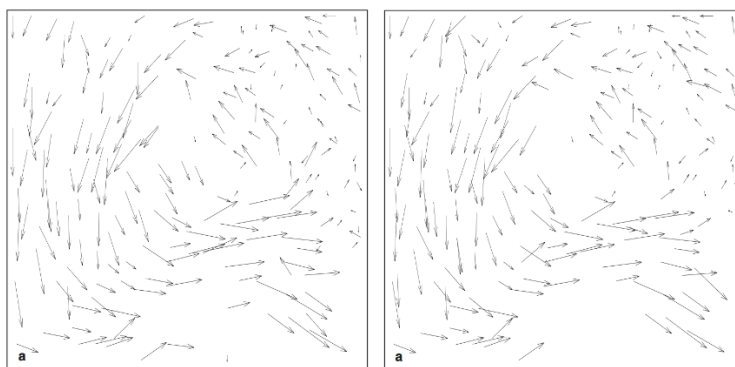


Figure 5.19. Comparison of the deterministic annealing approach (left) and super resolution PIV (right).<sup>204</sup>

Deterministic annealing was applied to experimental data at 0.0036 ppp. The results show that the number of vectors obtained by this approach is comparable to that of super resolution PIV. The resulting particle pairings, however, contain fewer mismatches than nearest neighbor PTV, super resolution PIV, or the relaxation method.

The percentage of incorrectly identified tracks (unsuccessful interrogations) is compared in Figure 5.20 for various algorithms using synthetic data. The particle density in this artificial data was 0.0047 ppp. An unsuccessful interrogation is quantified as any track that results in less than 5% error. For shearing flows, the line for deterministic annealing is not visible, as it lies on the x axis. The deterministic annealing approach yielded 0.0047 vpp for the shearing flow test and 0.003 vpp for flow that rotated one radian in a single time step. The other algorithms failed to recognize more than 0.0002 vpp for rotations greater than 0.2 radians. These results show the performance benefit of deterministic annealing over cross-correlation, relaxation, and nearest neighbor approaches in the presence of severe velocity gradients.

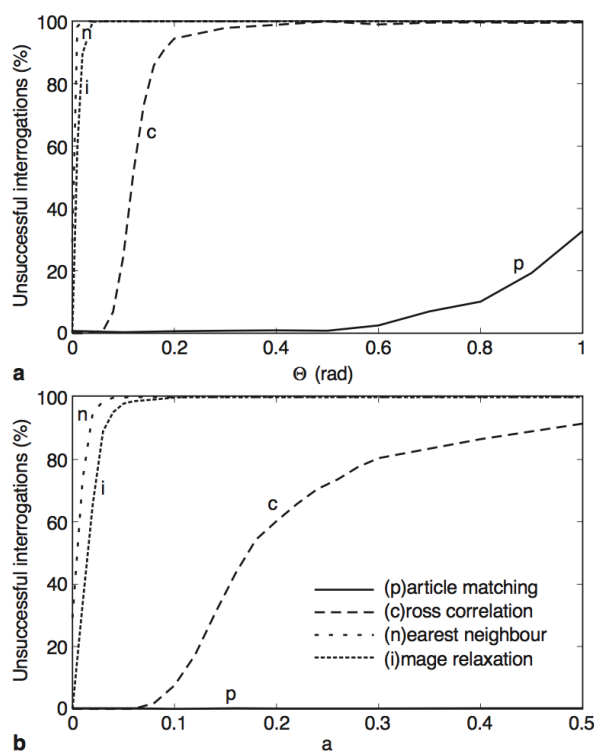


Figure 5.20. Performance of Deterministic annealing tracking (here labeled particle matching) for (a) rotational and (b) shearing flows.<sup>204</sup>

This algorithm was extended to 3D and applied to holographic PTV by Krepki *et al.*<sup>205</sup> Extension into 3D involves a more complicated 3 x 3 transformation matrix, but otherwise behaves

the same as 2D deterministic annealing. In order to test the deterministic annealing tracking technique in 3D, a test was performed using a spherical interrogation volume with diameter 128 pixels. The test had a rotation of  $4^\circ$  and a translation of 12.8 pixels. The seeding density tested for 480 particles was 0.037 ppp. The algorithms compared are PIV cross-correlation, BICC (here labeled concise cross-correlation), nearest neighbor, 3D deterministic annealing (labeled particle matching rotation), and simplified 3D deterministic annealing (labeled particle matching translation). Figure 5.21 shows that deterministic annealing performs poorly with a low particle density, but better than the compared algorithms above 0.003 ppp, corresponding to 40 particles. The nearest neighbor and cross-correlation algorithms showed deteriorating performance at the highest densities tested.

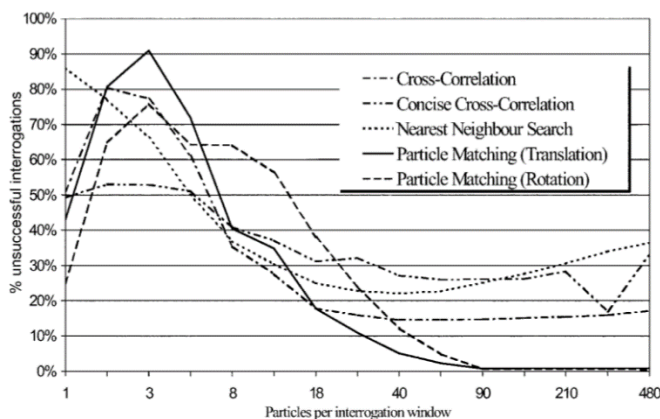


Figure 5.21. Comparison of two deterministic annealing techniques with cross-correlation and nearest neighbor matching techniques in 3D.<sup>205</sup>

The algorithm was also successfully applied to both laminar and turbulent flows for particle densities ranging from 20 to 350 particles per  $3,968 \times 4,032 \times 3,600 \mu\text{m}$  interrogation region, although this was not presented in terms of pixels. The computation time for the algorithms was compared for a test with 40 particles using a 333 MHz processor in Table 5-7. It can be seen that the deterministic annealing technique took twice as long as the BICC method and ten times as long as the nearest neighbor algorithm. The translational-only particle matching algorithm decreased

the computation time to less than that of the BICC method. It should also be noted that all PTV techniques were at least three times faster than the PIV cross-correlation techniques.

Table 5-7: Computational speed of tracking algorithms for 40 particle tracks.<sup>205</sup>

Interrogation procedure	CPU time [s]		
	Interrogation	Particle extraction	Complete procedure
Cross-correlation (64 × 64 × 64)	9.4267	0.00	9.4267
Cross-correlation (128 × 128 × 128)	96.523	0.00	96.523
Nearest neighbor	0.0049	0.32	0.3249
CCC	1.2366	0.32	1.5566
PM (translation only)	0.6772	0.32	0.9972
PM (rotation and translation)	3.3724	0.32	3.6924

### 5.7.2 Variational Approach

The variational approach is an iterative method that takes the nearest neighbor concept of particle matching and imposes a smoothness constraint. The technique is described for PTV as developed by Ruhnau *et al.*<sup>206</sup>. The mechanism of the variational approach is shown in relation to nearest neighbor estimates in Figure 5.22. The smoothness constraint is imposed as a regularization of local velocity vectors.

In order to implement this algorithm, Ruhnau *et al.* introduced a convex attraction potential,  $E_{local}(\vec{u})$ , which attracts every particle to its closest neighbor

$$E_{local}(\vec{u}) = \sum_{i=1}^M \frac{\alpha}{2} (d_T(S_i + \vec{u}_i)), \quad (5.41)$$

where  $S_i$  and  $T_i$  are the extracted particle coordinates in the first and second images, respectively,  $\vec{u}_i$  is the particle displacement,  $\alpha$  is a constant that was set to 0.8 in this test, and the operator  $d_T$  gives the Euclidean distance between its argument and the nearest particle position  $T$  in the second image. This minimization gives nearest neighbor matching for all the  $M$  particles in the set  $S$ .

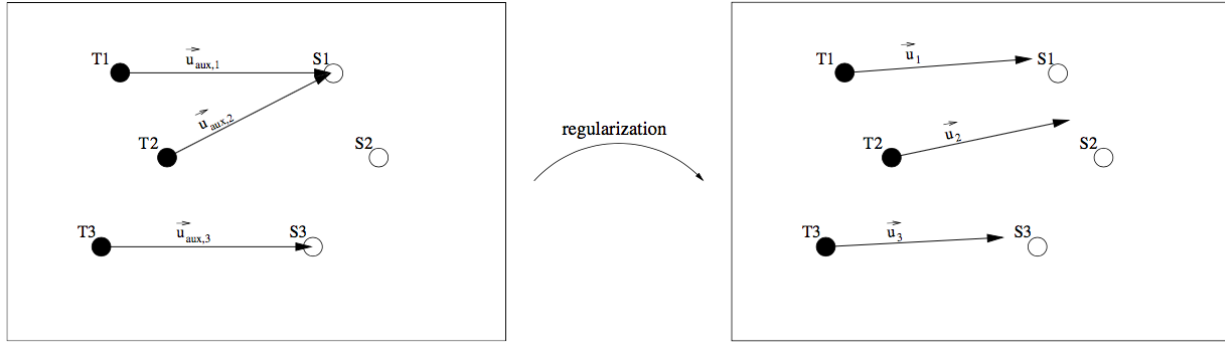


Figure 5.22. Nearest neighbor estimate of particle matches (left) and the result of regularization (right).<sup>206</sup>

The variational approach also applies a smoothness constraint, which is given as a minimization of velocity gradients within the flow. This term is added with a scalar multiplier,  $\lambda$ , to create the variational framework. The selection of the scalar multiplier allows the user to determine how much weight the smoothness constraint should hold in the minimization, and in this paper, it was set to 0.1. The smoothness constraint is

$$E_{global}(\vec{u}) = \int_{\Omega} \sum_{j=1}^N \left| \nabla \vec{u}^j(s) \right|^2 ds, \quad (5.42)$$

where  $N$  is the number of dimensions, and  $\Omega$  is the spatial domain. The two constraints are combined as

$$E(\vec{u}) = E_{local}(\vec{u}) + \lambda E_{global}(\vec{u}), \quad (5.43)$$

where minimization is performed iteratively by deforming the current vector field, then regularizing it with respect to neighbors. Matchless particles are removed if their contribution to the minimization exceeds an adjustable threshold. Iterations are done to minimize the combination of potentials until the solution converges.

A comparison of the variational approach with others is given below in Table 5-8. The test was done on VSJ standard image 301 using the particle mask correlation method for identification, resulting in roughly 0.017 ppp. The table includes the number of found vectors, the number of valid vectors, and the reliability (ratio of estimated and correct matches) of a few algorithms. The variational approach achieved 0.014 valid vpp in the test. It yielded both the most vectors and highest reliability compared to four-frame, binary cross-correlation, and relaxation algorithms.

Table 5-8: Comparison of four frame tracking (FIT), binary image cross correlation (BCC), relaxation (NRX), and variational approach (VAR) for the VSJ 301 image sequence.<sup>206</sup>

Algorithm	Frames	Estimated matches	Correct matches	Reliability (%)
FIT	0, 1, 2, 3	630	559	88.73
BCC	0 → 1	860	788	91.62
	0 → 2	863	691	80.07
NRX	0 → 1	808	788	97.52
	0 → 2	714	680	95.24
VAR	0 → 1	872	865	99.20
	0 → 2	904	885	97.90

In a 3D extension of this method (shown in Table 5-9), tests of the algorithm using VSJ standard image 331 performed with yield and reliability both exceeding 98% except for a severe decline when matching the first and third frames with the percent of removed particles exceeding 25%. The seeding density of this 3D synthetic data was roughly 0.05 ppp, and when particle images weren't removed, the variational approach yielded roughly 0.05 valid vpp. Note that the yield listed defines the ratio of identified matches to possible matches.

Table 5-9: Testing of the variational approach for VSJ standard image 331.<sup>206</sup>

Removed particles (%)	00 → 01			00 → 02		
	Possible matches	Yield (%)	Reliability (%)	Possible matches	Yield (%)	Reliability (%)
0	3364	100.00	99.76	3192	99.97	99.47
5	3037	100.00	99.84	2881	99.86	99.45
10	2731	100.00	99.60	2586	99.38	99.34
15	2440	100.00	99.59	2307	98.22	99.60
20	2170	100.00	99.40	2053	98.30	99.56
25	1885	100.00	99.74	1809	44.83	44.81
30	1649	100.00	99.40	1557	38.79	39.35
35	1403	99.93	99.64	1339	31.14	31.17
40	1211	100.00	99.26	1131	32.98	33.01

### 5.7.3 Genetic Algorithms

A genetic algorithm (GA) works to achieve a goal while minimizing a constraint, or fitness function. Genetic algorithms have been applied to PTV by Oyama and Kaneko,<sup>207</sup> Sheng and Meng,<sup>208</sup> and Doh *et al.*<sup>209</sup> by creating a fitness function that minimizes the sum of particle displacements between two frames. This algorithm works for low particle densities, but performance deteriorates when the mean particle spacing approaches the particle displacement ( $\xi \rightarrow 1$ ). Sheng and Meng<sup>208</sup> tracked 215 particles (0.00087 vpp) in an experimental holographic PTV experiment.

Kimura *et al.*<sup>210</sup> and Furukawa *et al.*<sup>211</sup> have proposed basing the fitness function on the morphology of neighboring particles, which successfully increased the possible number of tracked particles by three to four times, but never more than 1000 particles (0.015 vpp). Ohmi and Yoshida<sup>212</sup> based their fitness function on the rigidity of particle cluster patterns, which achieved more 0.03 vpp.

The following is the genetic algorithm proposed by Ohmi and Panday,<sup>213</sup> which used the same fitness function as Ohmi and Yoshida<sup>212</sup> with changes made only to the genetic mutations in order to increase convergence. The first step for creating a genetic algorithm is defining a genetic encoding. In this case, the genetic encoding is in the form of chromosomes. Each chromosome is

a list of particle IDs for a single frame. An individual is a set of two chromosomes, one for each image. The corresponding elements of each chromosome indicate particle pairs (i.e. the particle listed in the  $i$ th element of the first chromosome is paired with the particle listed in the  $i$ th element of the second chromosome). The two chromosomes do not need to be the same length, allowing for no-match particles to exist. The individuals mutate and reproduce with one another using a number of methods, which allows for production of new individuals. The fitness is evaluated to determine which individuals continue to reproduce,

$$F = \sum_{i=1}^m \sum_{k=1}^p |d_k - r_k|, \quad (5.44)$$

where  $F$  is the fitness function defined to evaluate the genetic algorithm. There are  $m$  particles in the first image, with  $p$  neighbors for each.  $d_k$  is the virtual displacement of particle  $k$ , which is equal to the displacement of the particle  $m$ .  $r_k$  is the actual displacement of particle  $k$ .

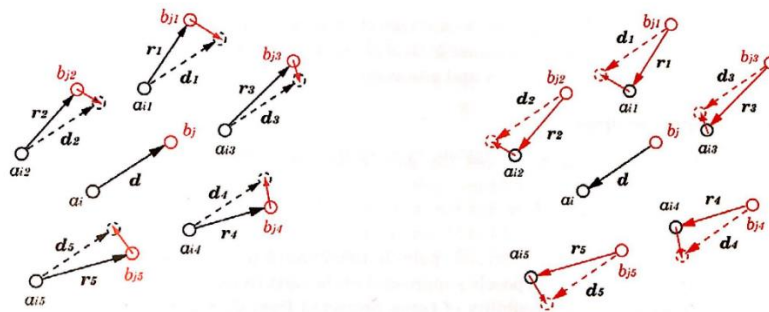


Figure 5.23. Relaxation parameter used as a fitness function by Ohmi and Panday.<sup>213</sup>

The genetic operations performed in this technique include the generation of the initial population, selection and reproduction, sorting, crossover, and mutation. The generation of the initial population is simply a randomized integer labeling of the particle IDs in both image one and image two. Selection is the step of the GA in which the fitness function is evaluated for each individual. Selection is done using either exponentially ranking selection, which generates child

individuals at a rate determined by the exponential ranking of their fitness evaluation, or unconditional elite-reproducing selection, which reproduces from the most-fit individual in the generation history at a prescribed rate. When used with gene code sorting, both methods work effectively if used with the proper crossover and mutation methods. Sorting of genes is done every 10 to 50 generations to ensure that the particles that have good fitness scores are less likely to change than those with poor scores. The individual gene code pairs have their portion of the fitness function calculated, and the lowest scores are moved to the front of the chromosomes. Crossover and mutation are suppressed in the front part of the chromosome to protect the good gene code pairs from being separated. The crossover step uses two parents and splices their chromosomes at certain points to create offspring. The splicing of genes can cause two of the same particle to occur in the child chromosomes, and as such duplicates are moved accordingly. Figure 5.24 schematically shows this crossover process.

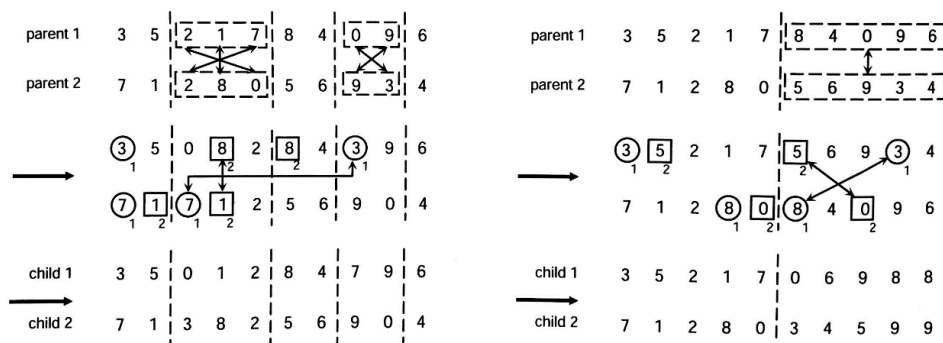


Figure 5.24. Multi-point inverse crossover (left) and single-point partially matched crossover (right).<sup>213</sup>

Mutation occurs randomly, selecting two genes in a single chromosome to switch. The mutation rate is fixed throughout the algorithm. The above genetic operations are all applied iteratively until an individual reaches a converged fitness value and the gene code pairs stabilize. These gene pairs are taken as the final particle match results.

Using the PIV standard image #301 for a 2D test, GA\_PTV calculated 0.031 vpp (from a 0.031 ppp image) with no spurious vectors. The results from this test are shown in Figure 5.25. It was found that for 2D data, this particle concentration was the upper limit for achieving a correct tracking rate of 100%. The highest tested possible vpp was 0.034, and the algorithm achieved 99.7%, so the seeding density could be further increased.

The algorithm also successfully matched 3D PTV results for 2000 particles (0.0076 ppp) with 100% accuracy using exact particle locations in 3-D PIV standard image #351. Using a 1.8 GHz Core2Duo PC the computation took 474 seconds to achieve the 2000 matches, corresponding to 0.0076 vpp.

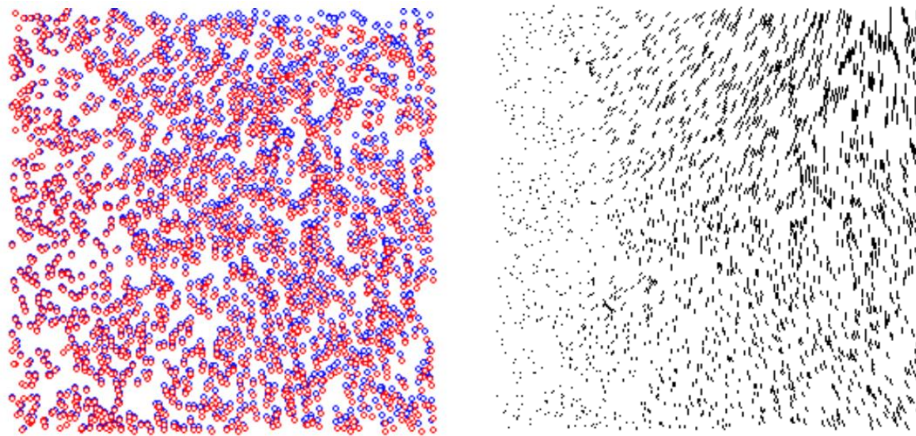


Figure 5.25. Original particle locations (left) and particle matching result (right) using the genetic algorithm for PIV standard image #301. <sup>213</sup>

#### 5.7.4 *Ant Colony Optimization*

Another algorithm for particle tracking is based on ant colony optimization. This is the same technique that is described for stereoscopic particle location in 4.2.1.3 aside from the fitness function used, which was used to update the pheromone for a particular particle track. Its use for particle tracking was introduced by Takagi<sup>214</sup> and improved by Ohmi *et al.*<sup>215</sup> In the context of tracking particle motions, agents investigate particle correspondences at two time steps. As a

nearest neighbor search is not a reliable metric for high seeding densities, the fitness function in this case is based upon the relaxation distance. The pheromone update is

$$\tau(i, j) \leftarrow (1 - \rho)\tau(i, j) + \sum_{k=1}^m \Delta\tau^k(i, j), \text{ and} \quad (5.45)$$

$$\Delta\tau^k(i, j) = \begin{cases} 1/L^k & \text{if } (i, j) \in T^k \\ 0 & \text{otherwise} \end{cases}, \quad (5.46)$$

where  $\rho$  is a constant that determines the dissipation rate of pheromone and  $L^k$  is the new relaxation length given by

$$L^k = \sum_{i=1}^m \sum_{k=1}^p |d_k - r_k|, \quad (5.47)$$

where  $d_k$  is the displacement from particle  $i$  to  $j$  and  $r_k$  is the displacement estimate of nearby particles. This is essentially the same as the relaxation probability from Section 5.3 and is shown in Figure 5.26, where  $b_{jk}$  is the length that is summed and minimized.

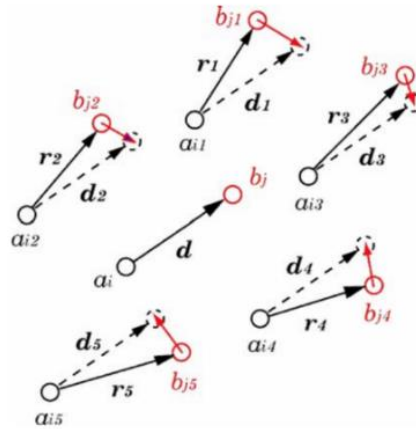


Figure 5.26. Concept of relaxation lengths for ant colony optimization.<sup>215</sup>

This algorithm is referred to as the relaxation minimization algorithm for comparison of results. The comparison of this algorithm with a four frame tracking approach and the binary cross correlation approach is given below in Table 5-10. Four test cases were done for each technique,

two with 500 potential pairs, two with 1000 potential pairs. The ant-colony optimization algorithm outperformed both the four-frame and BCC tracking methods in all the tests, achieving roughly 0.0152 vpp (99.8% correctly identified tracks) in the 1000 particle pair case (0.0153 ppp). The algorithm was also tested for 3D synthetic data using VSJ image #351 (0.008-0.015 ppp) and achieved 100% yield and reliability. This increase in performance is due to the fact that tracking is easier in 3D than in 2D if accurate 3D locations have been determined.

Table 5-10: Comparison of ant colony optimization using the relaxation minimization to other PTV tracking algorithms using VSJ image #301 (0.008-0.015 ppp).<sup>215</sup>

Frame	Name of the PTV methods	Number of particles	Correct pairs	Correct rate (%)
Frame0-1-2-3	Four-frame in-line tracking	500	459	91.8
		1,000	814	81.4
Frame0-1	Binary cross-correlation	500	382	76.4
		1,000	869	86.9
	Relaxation minimization	500	100	100
		1,000	998	99.8
Frame0-1-2-3	Four-frame in-line tracking	500	459	91.8
		1,000	814	81.4
Frame0-2	Binary cross-correlation	500	283	56.6
		1,000	659	65.9
	Relaxation minimization	500	498	99.6
		1,000	985	98.5

### 5.7.5 Fuzzy Logic PTV

Fuzzy logic algorithms employ a tolerance for imprecision while performing an optimization. This concept is often used in control systems; however, identifying particle tracks with noisy data is a good candidate for fuzzy control. This was applied to PTV by Wernet.<sup>216,217</sup> In order to determine particle tracks using fuzzy logic, two frames must be acquired and particle centroids must be located. Centroids in frame 1 are used as starting points and all of the centroids in frame 2 within a maximum displacement are initially considered. At this point, the vector field is noisy and there are many contradictory particle tracks, as shown in Figure 5.27.

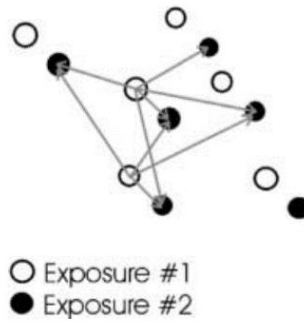


Figure 5.27. Particle centroids from two exposures superimposed. Potential candidates are shown for each particle in exposure 1 with a vector.<sup>217</sup>

For particles in frame 1 that have common matches in frame 2, then all possible matches are compared. The assumption is that if two particles are close enough to share a potential match, then their displacements should be similar. This holds true for higher order interactions as well. There are four inputs for each vector pair: distance between vector midpoints in pixels (Sep), average vector magnitude (Mag), difference in vector magnitude (MagDif), and the sum of squares of the differences of the x and y components of the two velocity vectors (Delta). All of these inputs are processed based on the confidence outputs shown in Table 5-11. Input combinations are all classified as the combination of high, medium, and low. The combination is quantified as a membership to each category,  $\mu$ . The concept of membership is shown in Figure 5.28. The range for each is determined by a minimum and maximum possible value (for example, the minimum and maximum displacements are 0 and 20 pixels, respectively in Table 5-11. The confidence outputs have values of 0.1, 0.5, and 0.8 for low, medium, and high, respectively. The confidence outputs from each category are weighted by a vector pair's membership and summed as

$$Confidence = \frac{\sum_{i=1}^N Conf_{out,i} \mu_{out,i}}{\sum_{i=1}^N \mu_{out,i}}, \quad (5.48)$$

where  $N$  is the number of outputs.

Table 5-11: Confidence outputs based on interaction of fuzzy logic inputs.<sup>217</sup>

	MagDif Small			MagDif Med			MagDif Large		
	Mag Small	Mag Med	Mag Large	Mag Small	Mag Med	Mag Large	Mag Small	Mag Med	Mag Large
<u>Sep Small</u>									
Delta Small	High	High	High	High	Med	Med	Med	Med	Low
Delta Med	High	Med	Low	Med	Med	Med	Low	Low	Low
Delta Large	Med	Low	Low	Low	Low	Low	Low	Low	Low
<u>Sep Med</u>									
Delta Small	High	Med	Med	Med	Med	Low	Med	Low	Low
Delta Med	Med	Med	Low	Med	Med	Low	Low	Low	Low
Delta Large	Low	Low	Low	Low	Low	Low	Low	Low	Low
<u>Sep Large</u>									
Delta Small	Med	Med	Low	Med	Low	Low	Low	Low	Low
Delta Med	Low	Low	Low	Low	Low	Low	Low	Low	Low
Delta Large	Low	Low	Low	Low	Low	Low	Low	Low	None

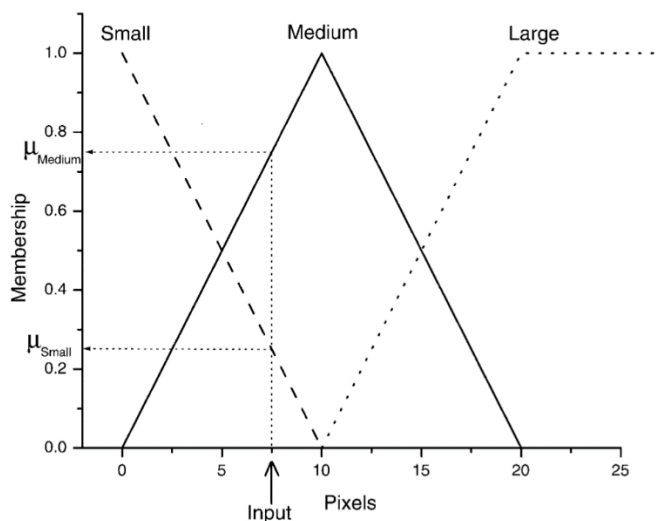


Figure 5.28. Representation of membership based on a fuzzy logic input.<sup>217</sup>

After a single pass of this confidence output, the fuzzy logic algorithm is applied again to compare non-interacting velocity vectors. The resulting vector with the maximum confidence is selected for each particle in frame 1. The final confidences that exceed a threshold value are then considered valid vectors.

Wernet<sup>217</sup> suggested using PIV results as a guide for tracking. Cross-correlation results give an initial estimate of local velocity within each interrogation region. For each first frame particle, the nearest four cross-correlation vectors are used for comparison using the fuzzy logic described above, resulting in a particle match that is most similar to the local cross-correlation results. This is repeated for each first frame particle.

In order to test this tracking algorithm, experimental data imaging a centrifugal compressor with impeller tip speed equal to 490 m/s was used. Within the 1000 x 1000 pixel CCD, 6400 particle images (0.0064 ppp) were detected in each frame. Using the PIV results to guide fuzzy logic tracking, 4000 velocity vectors (0.004 vpp) were tracked, which is equal to a yield of over 60%. Of the 4000 vectors, 294 were identified as spurious because they deviated from the mean by 45% while the turbulent fluctuations were expected to be no more than 15%. Thus the reliability of this tracking was 93%.

## 5.8 DELAUNAY TESSELLATION METHODS

Methods in this category match patterns using Delaunay tessellation in order to find particle matches in PTV (DT-PTV). The DT-PTV techniques are based on an algorithm by Song *et al.*<sup>218</sup> Delaunay tessellation takes a set of scattered points in space and generates a set of triangles. These triangles are generated such that there are no nodes (scattered points) that exist in the circumcircle of any tessellated triangle. The technique for generating a Delaunay tessellation begins by generating a super-triangle (shown in Figure 5.29a) that encompasses all of the nodes. Within the super-triangle, nodes are introduced one at a time. Whichever triangle a new node intersects with is split into three different triangles. This super-triangle and subsequent division is shown in Figure

5.29b. This is repeated until all nodes have been introduced. In most cases, there exists one unique tessellation for a given set of scattered nodes.

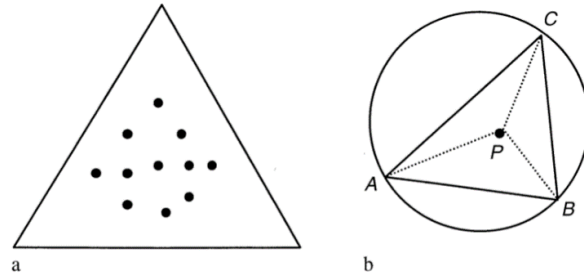


Figure 5.29. (a) Super-triangle and (b) triangle division for a new point,  $P$  introduced in Delaunay Tessellation.<sup>218</sup>

In order to apply Delaunay tessellation to particle tracking, a tessellation is generated for each of two consecutive images, where each particle location is a node. After tessellation, the triangles are each indexed. For each triangle in the first frame, a cross-correlation coefficient is calculated to find the overlapping area with each triangle in the second frame within a maximum displacement distance. This coefficient is calculated as

$$C_{ij} = \frac{\text{Area}(tri_i \cap tri_j)}{\sqrt{\text{Area}(tri_i)\text{Area}(tri_j)}}, \quad (5.49)$$

where the overlapping area of triangles can be calculated using the method named ‘‘Clipping of Polygons,’’ described by Harrington.<sup>219</sup> The overlap area is defined based on the number of vertices used. This can range from 3 to 6 for two overlapping triangles. Thus the area of a polygon with an even number of vertices  $Np$  is

$$S = \frac{1}{2} \sum_{k=1}^{Np/2} \{(x_{2k-1} - x_1)(y_{2k} - y_{2k-2}) + (x_{2k} - x_2)(y_{2k+1} - y_{2k-1})\}, \quad (5.50)$$

and the area for a polygon with an odd number of vortices is

$$S = \frac{1}{2} \sum_{k=2}^{Np} (x_k - x_1)(y_{k+1} - y_{k-1}), \quad (5.51)$$

where  $x_k$  and  $y_k$  denote the coordinates of vertex  $P_k$ .

In order to account for flow deformations, translation and rotation of triangle areas must be applied. Song's algorithm translates the center of gravity of the two triangles to be compared to the same point, then both triangles are rotated such that the vertex of the smallest angle of each triangle lies on the positive x-axis. This translation and rotation procedure is shown for two triangles in Figure 5.30.

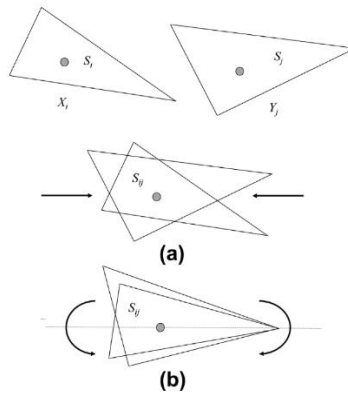


Figure 5.30. Translation and rotation of polygons for coefficient calculation.<sup>218</sup>

The match for a triangle in image 1 is selected as the triangle in image 2 corresponding to the maximum correlation coefficient. The individual particle matches are then selected as the nearest vertices in the rotated and overlapped triangles.

This technique was tested and compared to BICC for a simulated analytical Taylor-Green vortices with Reynolds number of 1000. The results are shown in Table 5-12. For low density images with small deformations the DT-PTV algorithm performed no better than BICC. However, in the tests using larger time steps between frames, and therefore containing more substantial flow

deformations, DT-PTV managed to increase the vector yield by 42% while also reducing the computation time by roughly two orders of magnitude compared to BICC.

Zhang *et al.*<sup>220</sup> made two improvements to DT-PTV that allowed the approach to perform in three-dimensions. The rotation and volume overlap calculation of tetrahedrons involves tedious, though not impossible, algorithms by introducing a characteristic vector to describe each tetrahedron. The characteristic vector for a tetrahedron includes a value for the area of each face of the tetrahedron. These face areas are sorted by increasing area. The tetrahedrons in the first and second images must be sorted the same order, so as to simulate rotation. The correlation coefficient between two tetrahedrons is

$$C_{ij} = \left[ \frac{\min(|X_i'|, |Y_j'|)}{\max(|X_i'|, |Y_j'|)} \right] * \left[ \frac{\text{cov}(X_i', Y_j')}{\sqrt{\text{var}(X_i') \text{var}(Y_j')}} \right], \quad (5.52)$$

where  $X_i'$  is the sorted characteristic vector of the tetrahedron in the first frame and  $Y_j'$  is the characteristic vector of the candidate tetrahedron in the second frame.

Table 5-12: Comparison of computation time and number of vectors obtained for BICC and DT-PTV tracking algorithms.<sup>218</sup>

Num.	$\Delta t$	R	CPU TIME		Vectors num.	
			BICC	DT-PTV	BICC	DT-PTV
632	0.01	0.1	131	6	632	632
631	0.05	0.1	130	6	630	631
1505	0.1	0.12	2480	24	1500	1480
1498	0.2	0.22	7437	26	1245	1383
1500	0.3	0.33	14721	31	910	1295

The second change was to use tessellation to get rid of the maximum displacement criteria for the search area. Instead, triangles in the second frame are organized such that for triangle  $D_i$ , all triangles sharing a vertex are considered a first connection. Then all triangles sharing a vertex with these are considered a second connection to  $D_i$  and so on. A maximum number of connections,

rather than a maximum displacement, is then selected such that all triangles in frame 2 that are within  $\alpha$  connections of the reference triangle in frame 1 are considered potential matches.

The algorithm also uses a dual computation technique (see Section 5.11) that calculates matches going from frame 1 to 2 and from frame 2 to 1. In this way, there are redundant matches found for each particle pair. When two particles claim the same match, the pair with the most occurrences is selected as the correct match. This is the same technique used by Jia *et al.*<sup>193</sup> in Section 5.3.

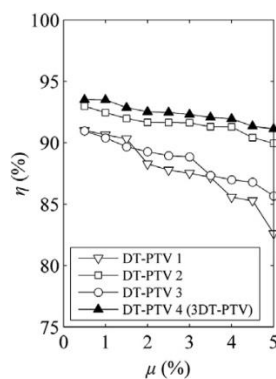


Figure 5.31. Comparison of DT-PTV methods with the improvements suggested by Zhang *et al.* for VSJ standard image #301 ( $\sim 0.06$  ppp).<sup>220</sup>

DT-PTV was tested with and without the above improvements. DT-PTV 1 used a fixed search radius and the net-flux method, DT-PTV 2 used a fixed search radius and the dual computation method, DT-PTV 3 used the DT search with only first connection triangles and the net-flux method, and DT-PTV 4 used the DT search and dual computation. Figure 5.31 compares the performance of the different DT-PTV algorithms using VSJ standard image 301 with 0.06 ppp. DT-PTV 4 achieved the highest ratio of correct matches to possible matches,  $\eta$ , for the flow investigated as a function of the particle dropout rate,  $\mu$ . It is clear that the dual computation method substantially improved results of the DT-PTV algorithm for high particle dropout rates. Given the

particle density of 0.06 ppp, the vector density obtained by DT-PTV 4 for  $\eta=91\%$  and  $\mu=5\%$  was 0.053 vpp.

## 5.9 VORONOI DIAGRAM METHODS

Another technique, called VD-PTV was introduced by Zhang *et al.*<sup>221</sup> and is based on the Voronoi diagram (VD). Voronoi diagrams are similar to Delaunay tessellations; however, the Voronoi diagram segments an image such that a single cell exists that surrounds each data point. For a given data point, its corresponding cell is a region that comprises all points closer to that data point than others. In this case, a data point is a particle image. The Voronoi diagram is generated by creating a Delaunay tessellation, generating a circumcircle for each triangle, identifying the midpoint of each circumcircle, and connecting them in a nearest neighbor sense. An example VD is shown in Figure 5.32.

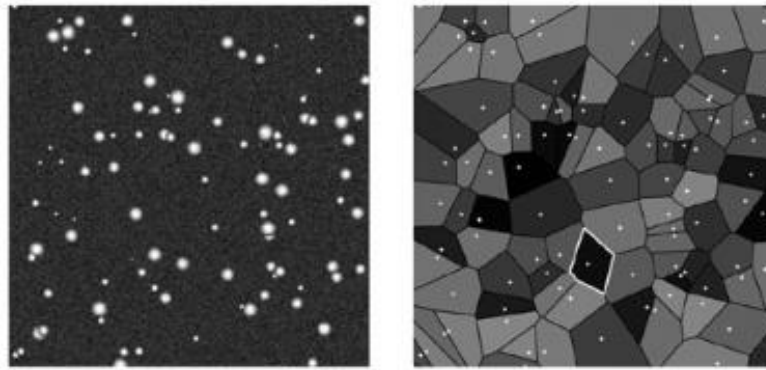


Figure 5.32. Particle images (left) and corresponding Voronoi diagram (right).<sup>221</sup>

For each particle image, the polar radius of the Voronoi cell is found as a function of polar angle. This is called the characteristic curve and is shown in Figure 5.33. The curve for a Voronoi cell is defined as the polar radius,  $r$ , as a function of polar angle,  $\alpha$ . The curve is determined by first dividing the cell into triangles and calculating

$$r(\alpha) = \frac{h}{\left| \sin(\alpha + \theta_1 - \alpha_1) + 1 - \text{logical}(\theta_1 - 90^\circ) \right|}, \quad (5.53)$$

$$\text{logical}(x) = \begin{cases} 1 & (x \neq 0) \\ 0 & (x = 0) \end{cases}, \quad (5.54)$$

$$h = \frac{\left| (y_2 - y_1)x_0 + (x_1 - x_2)y_0 + x_2y_1 - x_1y_2 \right|}{\sqrt{(y_2 - y_1)^2 + (x_1 - x_2)^2}}, \quad (5.55)$$

where  $(x_i, y_i)$  are the coordinates of each vertex in a triangle and  $\theta_l$  is the angle at vertex 1. The curve for each triangle in a Voronoi cell is then combined piecewise such that the polar radius is defined for  $\alpha$  between 0 and 360 degrees.

In the proposed method for particle matching, the characteristic curves for reference particles and candidate particles are compared. A coefficient for the similarity between the curves is calculated and the particle images with the highest similarity are chosen as a match. The similarity coefficient is calculated as

$$C_r = \frac{\text{cov}(r_1, r_2)}{\sqrt{\text{cov}(r_1) \text{cov}(r_2)}}. \quad (5.56)$$

The similarity coefficient is calculated between  $r_1(\theta + \alpha)$  and  $r_2(\alpha)$ , where  $\theta$  ranges from 0 to 360 degrees, which allows for rotation in the matching process. The maximum  $C_r$  calculated is then taken as the match coefficient between reference and candidate particle. The candidate particle with the highest associated similarity coefficient is selected as the match for every reference particle. If two candidate particles share the highest similarity coefficient, then the reference is considered to have no match. If two reference particles claim the same candidate particle, then the one with the higher similarity coefficient is considered the match.

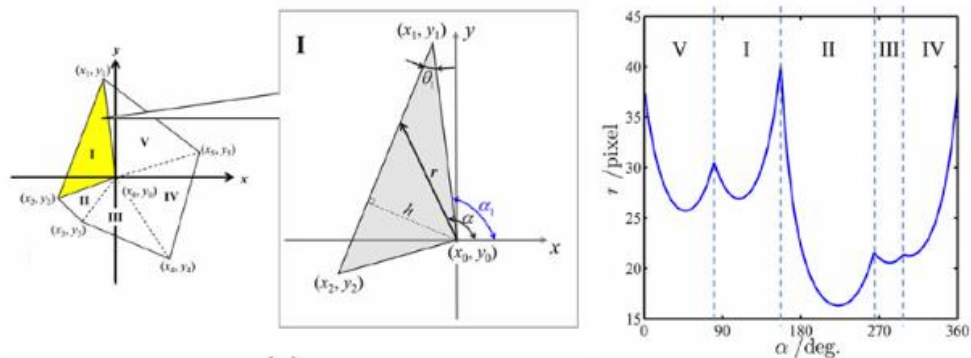


Figure 5.33. The development of the Voronoi cell characteristic curve.<sup>221</sup>

A comparison of VD-PTV with relaxation and DT-PTV was performed using a synthetic flow consisting of a superposition of shear flow, a vortex, and a dipole. The image was 256 x 256 pixels and the mean particle displacement was 2.0 pixels. The particle concentration was varied from 4 to 25 ppp. Note that this particle concentration is not possible in experiments, and the most important parameter to consider is the ratio of mean particle displacement to mean particle spacing,  $\overline{C_{PTV}}$ . Figure 5.34 shows that VD-PTV has the highest correct matching rate,  $\eta$ , at a range of particle densities and search radii,  $R_s$ . Note that while the relaxation DT-PTV algorithms performed well when the search radius was kept near the mean particle displacement, performance deteriorated when the search radius increased. VD-PTV shows that it is effective even when the mean particle displacement is not explicitly known. Furthermore, the computation time for VD-PTV was less than half of that for DT-PTV and less than 15% of that for relaxation.

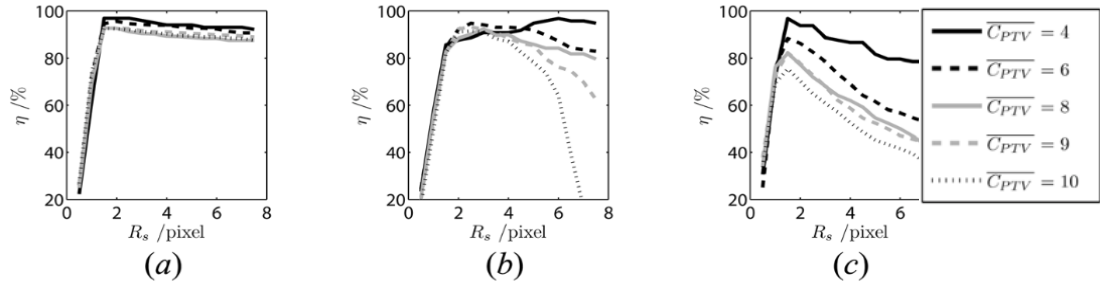


Figure 5.34. Comparison of (a) VD-PTV (b) Relaxation (c) DT-PTV with % proper matches,  $\eta$ , vs. the search radius,  $R_s$ .  $\overline{C_{PTV}}$  is the ratio of average particle displacement to average particle spacing.<sup>221</sup>

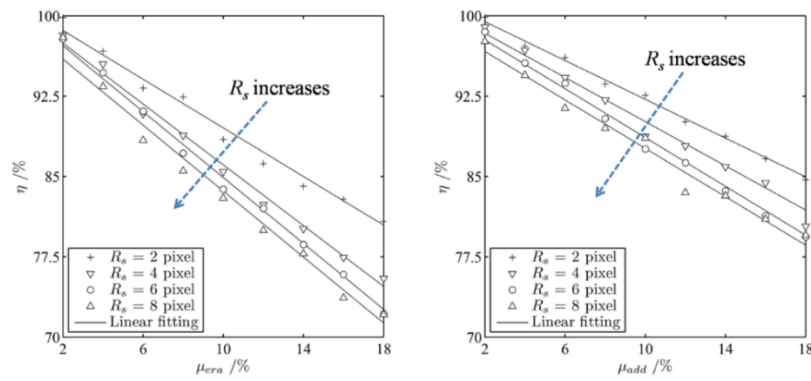


Figure 5.35. Influence of erased (left) and added (right) particles on the performance of VD-PTV for the homemade flow with  $\overline{C_{PTV}} = 8$ .<sup>221</sup>

Using the synthetic flow with  $\overline{C_{PTV}} = 8$ , the effect of adding and removing particles was tested. The fraction of erased particles,  $\mu_{era}$ , and added particles,  $\mu_{add}$ , were each varied from 2 to 18%, the results of which are shown in Figure 5.35. The correct matching rate decreased linearly with increasing  $\mu_{era}$  or  $\mu_{add}$ . This is due to the fact that adding or removing a particle affects Voronoi cells of nearby particle images.

## 5.10 VISION BASED PTV

The vision based matching algorithm, introduced by Lei *et al.*<sup>222</sup> in 2012 and improved upon by Paul *et al.*<sup>223</sup> in 2014, correlates image features by adhering to three principles: proximity, similarity, and exclusion. The method was first introduced by Scott and Longuet-Higgins in

1991<sup>224</sup>. In order to enforce the principle of proximity, the proximity matrix,  $G$ , is created based on  $r_{ij}$ , the distance from point  $i$  in frame 1 to point  $j$  in frame 2,

$$G_{ij} = e^{-r_{ij}^2/2\sigma^2}, \quad (5.57)$$

where  $\sigma$  is a characteristic distance, originally selected to be the maximum expected displacement. Singular value decomposition is then performed on the new  $G$  matrix (Equation 5.58). The two rotation matrices  $T$  and  $U$  are maintained while the singular value matrix,  $D$ , is replaced with the identity matrix of the same dimensions. This normalizes to give the proximity matrix,  $P$ , which has a least squares mapping between  $i$  and  $j$  (Equation 5.59). Within the  $P$  matrix, if element  $i, j$  is the maximum value in both the  $i$ th row and the  $j$ th column, then the two particles are considered a match.

$$G = TDU \quad (5.58)$$

$$P = TIU \quad (5.59)$$

In 1997 Pilu<sup>225</sup> proposed a modified method that resulted in improved particle pair matching, where feature similarities were accounted for using a normalized cross-correlation matrix,  $C_{ij}$ , (see Equation 5.60) by creating a modified  $G$  matrix in (Equation 5.61).

$$C_{ij} = \frac{\sum_{u=1}^W \sum_{v=1}^W (I_A(u,v) - I_{Am})(I_B(u,v) - I_{Bm})}{W^2 \sum I_A \sum I_B}, \quad (5.60)$$

where  $I_A(u,v)$  and  $I_B(u,v)$  are the pixel intensities in image  $A$  and  $B$ , respectively, and  $I_{Am}$  and  $I_{Bm}$  are the mean pixel intensities in each frame. The cross-correlation matrix is then used to create an updated proximity matrix

$$G'_{ij} = e^{-r_{ij}^2/2\sigma^2} \times e^{-(C_{ij}-1)^2/2g^2}, \quad (5.61)$$

where  $\gamma$  is a factor that controls the speed of decay of the weighting of the cross-correlation term. It was set to 0.4 for this algorithm.

The match is then tested using an outlier detection scheme (see Section 5.11) to validate the match. If valid, the row and column are removed from  $P$  in order to maintain the exclusivity principle. If the match is identified as an outlier, the coefficient  $P_{i,j}$  is then set to zero and the next maximum is selected. This scheme is iterated until all particle pair matches are achieved.

The algorithm used by Scott and Longuet-Higgins selected  $\sigma$  to be the average displacement of the particles. The improvement made by Pihu suggested that the PIV result for displacement estimate should give the  $\sigma$  value used. Lei *et al.* suggested the use of twice the PIV displacement for  $\sigma$ , as tracking performance is improved when  $\sigma$  is an overestimate of the displacement.

Further improvement by Paul *et al.* modified the proximity matrix as

$$G_{ij} = \exp\left[-\frac{(r_{ij} - r_{PIV})^2}{2\varepsilon^2}\right] * \exp\left\{\left[\left(\tan^{-1} \frac{dy_{ij}}{dx_{ij}}\right)^2 - \left(\tan^{-1} \frac{dy_{PIV}}{dx_{PIV}}\right)^2\right] / 2\varphi^2\right\}, \quad (5.62)$$

where it was suggested that  $\varepsilon$  be set to 4 and  $\varphi$  be set to 1 for high gradient flows. This modification favors particles with a displacement equal to the PIV results over particles with zero displacement.

Figure 5.36 shows the modification graphically.

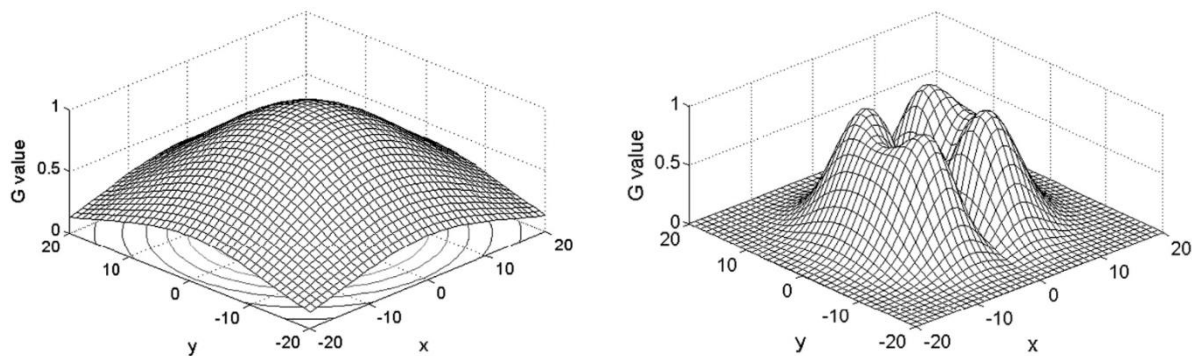


Figure 5.36. Proximity surface with (right) and without (left) modifications made by Paul *et al.*<sup>223</sup>

The modification to the proximity matrix has four peaks. One of these corresponds to the displacement calculated using PIV. This means that the algorithm will favor particles with displacement close to the PIV estimate more than particles with zero displacement. The performance improvement of the modification is apparent in Figure 5.37, where the modified algorithms achieve both lower RMS error and higher match yield, particularly at high displacements and velocity gradients.

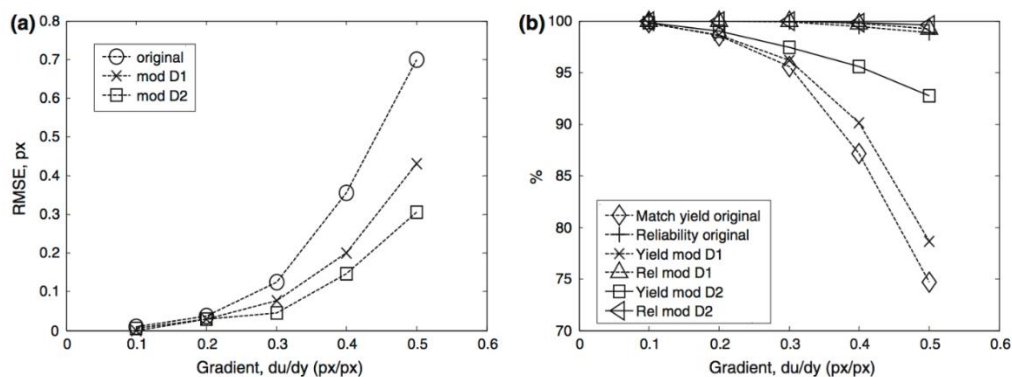


Figure 5.37. Comparisons of the original and modified VB PTV algorithms with a maximum velocity of 25 pixels. D1 and D2 are the modifications of Paul *et al.*<sup>223</sup> that use displacement and directional guidance, respectively.

Table 5-13 shows the performance improvement of VB PTV over previous algorithms. Given the correct particle locations, the reliably identified correct particle tracks at 0.062 ppp, achieving

0.06 vpp. Additionally, using the same image, but first performing cascade correlation particle identification (see Section 3.2.3) to find 0.032 ppp, VB PTV identified 0.027 vpp. The only comparable results came from the EPTV method.

Table 5-13: PTV results with known and unknown particle locations for VB PTV and previous tracking algorithms using VSJ #301.<sup>222</sup>

Algorithm	Particle location	Matches possible	Matches found	Matches correct	Match yield (%)	Reliability (%)
Present work (tracking only)	Known	4,042	4,039	3,927	97.23	97.15
VAR (Ruhnau et al. 2005)	Known	4,042	4,039	3,894	96.34	96.41
EPTV (Mikheev and Zubtsov 2008)	Known	4,042	3,863	3,823	94.58	98.96
ICCRM (Brevis et al. 2011)	Known	4,042	NA	3,980	98.46	NA
Present work (particle identification + tracking)	Unknown	2,095	1,846	1,761	84.06	95.40
EPTV (Mikheev and Zubtsov 2008)	Unknown	2,029	1,759	1,733	85.41	98.52
VAR (Ruhnau et al. 2005)	Unknown	NA	872	865	NA	99.20
NRX (Ohmi and Li 2000)	Unknown	NA	808	788	NA	97.52
MF-EPS (Shindler et al. 2011)	Unknown	NA	1,160	1,146	NA	98.80
2F-EPS (Shindler et al. 2011)	Unknown	NA	1,123	1,112	NA	99.00

## 5.11 OUTLIER DETECTION

In the previously described tracking algorithms, both the yield and the reliability of identified particle tracks are of interest. Algorithms that have a reliability below 100% have at least one incorrect particle track, which not only corrupts the velocity field, but also affects the differential and integral quantities such as vorticity and streamlines.<sup>226</sup> For this reason, it is of the utmost importance to have algorithms determine which particle tracks are outliers and remove them. One technique for outlier detection was developed by Song *et al.*<sup>218</sup>, which checks that flow results satisfy the continuity equation. The algorithm uses Delaunay tessellation and calculates the divergence within each triangle. In 2D, the flux through each triangle edge is estimated by the mean outward normal component of velocity of the adjacent triangle vertices. The flux is summed for each triangle and compared to a threshold. Spurious vectors are identified as vertices that are

part of multiple triangles that exceed a set threshold for divergence. At present, this algorithm is not general due to the incompressibility condition that is imposed.

Jia *et al.*<sup>193</sup> and Zhang *et al.*<sup>220</sup> both proposed that performing tracking from frame 1-2 and from frame 2-1 can identify spurious vectors. The particle pairs from frame 2-1 are then inverted and combined as shown in . The pairs that have no conflict (5~7) are selected as valid matches. Then if conflicts arise, the match with the most occurrences (1~4 and 3~8) is selected as valid. Conflicts that appear equally (2~6 and 2~9) are considered invalid and removed. This procedure is called either dual computation or bidirectional computing. The results for this method compared to the net-flux method using DT-PTV are shown in Figure 5.38, where the bidirectional computation algorithm improved the correct matching rate by roughly 5% for high rates of erased particles.

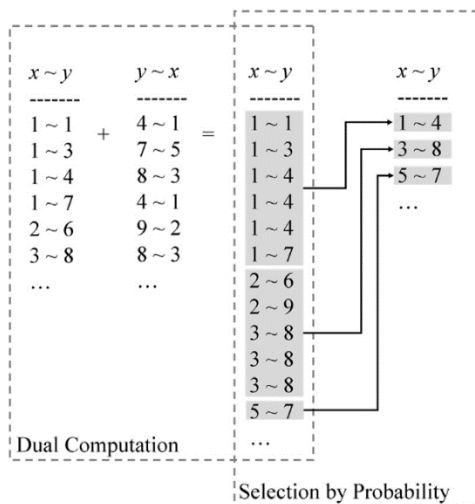


Figure 5.38. Bidirectional computing to remove spurious vectors.<sup>220</sup>

Sapkota and Ohmi<sup>227</sup> presented an outlier detection method that uses fuzzy logic to determine the probability that a vector is valid. A discussion of fuzzy logic schemes and the application to identifying particle tracks is discussed in Section 5.7.5. For the purpose of identifying spurious

vectors, each particle in the first frame is compared to its nearest neighboring particle. From the two corresponding vectors, the fuzzy logic inputs are the distance between vector midpoints,  $M$ , the difference in vector magnitudes  $D$ , the average value of the vector magnitudes  $A$ , and the sum of squares of the differences of  $x$ - and  $y$ -components of the velocity vectors  $XY$ . From these inputs, the fuzzy logic system outputs a confidence value, where a high confidence value suggests that a vector is likely valid, while a low confidence suggests that a vector is likely erroneous.

The rule base that determines the confidence outputs corresponding to inputs is shown in Table 5-14. As explained in Section 5.7.5, the confidence outputs are weighted based on the inputs' degree of membership and summed. This sum is then compared to a threshold value, which was 0.6 in this study. If the confidence exceeds this threshold, the associated vector is considered valid. Invalid vectors were shown to typically have confidence values around 0.1.

Table 5-14: Fuzzy logic rule base for identifying spurious vectors.<sup>227</sup>

<i>MS</i>	<i>DS</i>			<i>DM</i>			<i>DL</i>		
	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>
<i>XYS</i>	<i>High</i>	<i>High</i>	<i>High</i>	<i>High</i>	<i>Med</i>	<i>Med</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>
<i>XYM</i>	<i>High</i>	<i>Med</i>	<i>Low</i>	<i>Med</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>XYL</i>	<i>Med</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>MM</i>	<i>DS</i>			<i>DM</i>			<i>DL</i>		
	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>
<i>XYS</i>	<i>High</i>	<i>Med</i>	<i>Med</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>	<i>Med</i>	<i>Low</i>	<i>Low</i>
<i>XYM</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>XYL</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>ML</i>	<i>DS</i>			<i>DM</i>			<i>DL</i>		
	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>	<i>AS</i>	<i>AM</i>	<i>AL</i>
<i>XYS</i>	<i>Med</i>	<i>Med</i>	<i>Low</i>	<i>Med</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>XYM</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>
<i>XYL</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>	<i>Low</i>

The authors used a cross-correlation analysis to determine the range of values for the small,  $S$ , medium,  $M$ , and large,  $L$ , membership ranges for  $D$ ,  $A$ , and  $XY$  inputs. Additionally, the range for  $M$  inputs was zero to twice the distance between the nearest particles being considered.

This algorithm was tested using a synthetic vortex flow. Figure 5.39 shows that the algorithm successfully detected outliers. It was also suggested that this algorithm can be used to determine

the effectiveness of a tracking technique by examining the average and standard deviation values of confidence. A high mean confidence with low standard deviation suggests that a tracking procedure was successful.

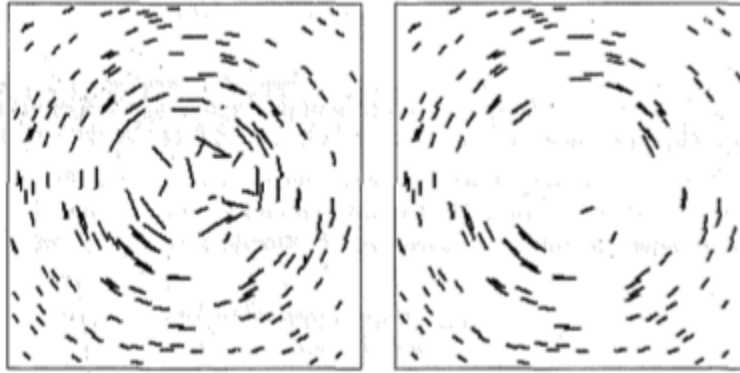


Figure 5.39. Simulated vortex flow with spurious vectors (left) and after spurious vectors have been removed using fuzzy logic algorithm.<sup>227</sup>

The universal outlier detection method of Westerweel and Scarano<sup>228</sup> is robust for gridded PIV data, and has significantly shorter computation time than the bootstrapping method. This method has been generalized to work for scattered PTV data by Duncan *et al.*<sup>229</sup> In the proposed technique, Delaunay tessellation was used to determine the neighbors of each data point. Neighboring data points share a triangle with one another, and on average each point will have five to eight neighbors using this criteria. These neighbors were used to calculate a normalized residual, which is defined as:

$$r_0^* = \frac{|U_0 - \text{med}(U_i)|}{\text{med}|U_i - \text{med}(U_i)| + \varepsilon_a(d + \varepsilon_a)}, \quad (5.63)$$

where  $U_0$  is the velocity measured at the data point in question,  $U_i$  is the velocity of each of its neighbors,  $d$  is the median of the distances between each of the neighbors and the data point in question, and  $\varepsilon_a$  is the adaptive tolerance, which was set to 0.1 pixel, as done by Westerweel and Scarano. Table 5-15 shows the performance of the proposed outlier detection scheme for gridded

and scattered data. The test was done using synthetic turbulent boundary layer data with added spurious vectors that were randomized with a mean about zero and a maximum equal to the maximum displacement in the flow. The algorithm was able to perform reliably in the presence of up to 15% outliers; however, it must be noted that when outliers are adjacent, the scheme becomes unreliable due to the increase in neighborhood fluctuation. It was suggested that the algorithm performs best when the outlier percentage is at or below 5%.

Table 5-15: Outlier detection removal rate for PIV and PTV data with added outliers.<sup>229</sup>

Added outliers	PIV 1	PIV 2	PTV 1	PTV 2
0%	0.1%	0.4%	0.9%	0.6%
5%	5.0%	5.3%	5.7%	5.5%
10%	10.0%	10.0%	10.3%	9.9%
15%	14.4%	16.2%	14.2%	13.8%

Masullo and Theunissen<sup>230</sup> noted that clusters of spurious vectors can cause poor performance for the universal outlier detection scheme. In order to improve the results in the presence of outlier clusters, it was suggested that an adaptive neighborhood radius should be selected. In order to determine an appropriate neighborhood radius, a coherence value,  $C$  is determined for each vector. This is done by calculating a weighted least squares estimate of the local velocity using a second order, 2D polynomial fit, with 8 neighbor vectors

$$\phi = a_0 + a_1 y + a_2 x + a_3 y^2 + a_4 xy + a_5 x^2, \quad (5.64)$$

with the weight of each neighbor vector,  $j$ , as

$$W_j = \exp \left( -1/2 \frac{(u_j - u_{median})^2 + (v_j - v_{median})^2}{\varepsilon + \sum (u_j - u_{median})^2 + (v_j - v_{median})^2} \right), \quad (5.65)$$

where  $\varepsilon$  is the background error estimate. This least squares estimate is then used to determine the coherence value

$$C = \frac{1}{2(|V|_m + \varepsilon)^2} \left\{ (\phi(x_0) - u_0)^2 + (\phi(y_0) - v_0)^2 \right\}, \quad (5.66)$$

where  $|V|_m$  is the median norm of velocity,  $(x_0, y_0)$  and  $(u_0, v_0)$  are the location and velocity components of the vector being investigated. Values of  $C$  that exceed the threshold of 10% are considered outliers. The coherence is calculated for each data point, then the neighborhood of each vector is progressively increased until at least half of the neighbors are coherent. This ensures that clusters of incoherent vectors are evaluated in relation to neighbors that comprise of more valid than invalid vectors. Coherence is then recalculated for the increased neighborhood size and compared to the 10% threshold for removal of outliers. This reduces both over-detection and under-detection of spurious vectors.

A further improvement to the above algorithm was the use of an updated weighting function to minimize the contribution of outlier vectors. Instead of comparing a vector to a weighted least squares estimate, an average-weighted median estimate (*awmed*) is generated and used for comparison. This is computed for each neighborhood of vectors by assigning each vector a weight based on adaptive Gaussian windowing (AGW)

$$w_i = \exp\left(\frac{-d_i^2}{\sigma^2}\right), \quad (5.67)$$

$$\sigma = \frac{1.24}{N_0} \sum_{i=0}^{N_0} d_i, \quad (5.68)$$

where  $d_i$  is the distance between vector  $i$  and the vector being analyzed, and  $N_0$  is the number of neighbor particles. The vectors are then organized by increasing magnitude and given widths equal to their weights. This is shown in Figure 5.40, where a width  $\Delta_w$  is centered around the median, and a weighted average of the vectors within this window is calculated. The value of  $\Delta_w$  between

0.3 and 0.5 was shown to give the best results. In this way, an estimate for the local velocity is obtained without the bias of outliers, which will be excluded from the local average.

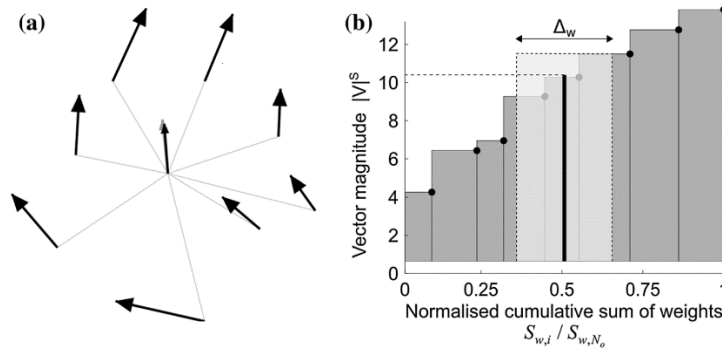


Figure 5.40. Sum of weighted vector magnitudes and calculation of the average-weighted median estimate.<sup>230</sup>

It was also suggested that the residual should be calculated in polar coordinates rather than Cartesian. Thus a combined residual was defined as

$$r^* = \left( r_\alpha^2 + r_{|V|}^2 \right)^{0.5}, \quad (5.69)$$

where the magnitude residual component is

$$r_{|V|} = \frac{\left| |V_0| - |V|_{awm} \right|}{\text{median} \left( \left| |V_j| - |V|_{awm} \right| \right) + \varepsilon}, \quad (5.70)$$

$$|V|_{awm} = \sqrt{\text{awmed} \left( u_j \right)^2 + \text{awmed} \left( v_j \right)^2}, \text{ and} \quad (5.71)$$

$$|V_j|^2 = u_j^2 + v_j^2. \quad (5.72)$$

The angular residual component is defined as

$$r_\alpha = \frac{\alpha_0^*}{\text{median} \left( \alpha_j^* \right) + \varepsilon_\alpha}, \text{ where} \quad (5.73)$$

$$\alpha_j^* = \min \left( \left| \alpha_j - \alpha_{awm} \right|, 2\pi - \left| \alpha_j - \alpha_{awm} \right| \right), \quad (5.74)$$

$$\alpha_j = \tan^{-1} \left( \frac{v_j}{u_j} \right), \quad (5.75)$$

$$\alpha_{awm} = \tan^{-1} \left( \frac{awmed(\sin(\alpha_j))}{awmed(\cos(\alpha_j))} \right), \text{ and} \quad (5.76)$$

$$\varepsilon_\alpha = \tan^{-1} \left( \frac{\varepsilon}{|V|_{awm}} \right) \quad (5.77)$$

Any vector with a residual  $r^*$  below 2 is considered invalid, which the authors noted agreed with the findings of Westerweel and Scarano.<sup>228</sup>

The proposed algorithm was tested for a simulated cellular vortex generated using DNS. Though the velocity vectors were obtained using PIV on an unstructured grid, the approach can be used on PTV data. Clusters of spurious points were generated at a time so as to test the ability of the outlier detection schemes in handling large clusters of invalid vectors. The number of vectors included in each cluster,  $C_f$ , was varied between 1 and 40. The spurious vectors were generated in a random direction and within a percentage,  $M$ , of the mean local velocity. For this test,  $M$  was set to 10%. Two ratios were used to quantify the algorithm's ability to identify spurious vectors:  $R_u$  is the ratio of missed outliers to imposed outliers,  $R_u^*$  is the ratio of missed outliers to the total number of vectors, and  $R_o$  is the ratio of correct vectors that were labeled outliers to the total number of vectors (see Figure 5.41).

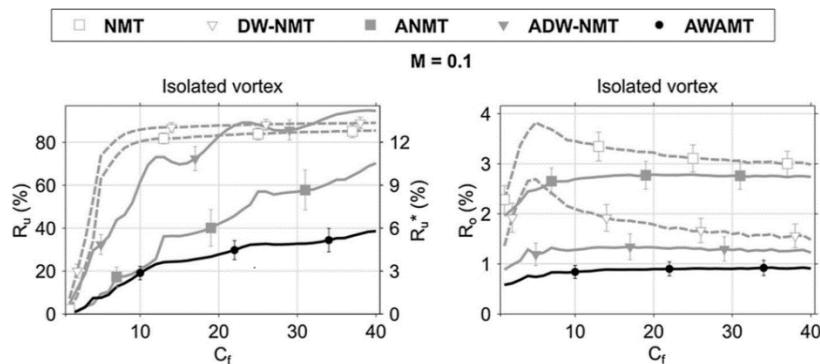


Figure 5.41. Comparison between universal outlier detection (NMT), distance-weighted outlier detection (DW-NMT), NMT with a variable neighborhood size (ANMT), DW-NMT with a variable neighborhood (ADW-NMT), and the proposed adaptive weighted angle and magnitude thresholding (AWAMT).<sup>230</sup>

The results show that the universal outlier detection algorithm could no longer identify more than 20% of outliers when the cluster size exceeded 10 vectors. Applying only the variable neighborhood improvement increased the outlier detection rate to 80% for 10-vector outlier clusters, although this rate dropped to roughly 30% for outlier clusters of 40 vectors. The proposed AWAMT algorithm succeeded in identifying 60% of outliers when added in clusters of 40 and managed to have a lower over-detection rate and under-detection rate than the compared algorithms for all outlier cluster sizes.

## 5.12 COMBINED TRACKING AND LOCALIZATION FOR 3D PTV

While standard PTV algorithms identify particle images, determine their spatial coordinates, and then perform tracking, some techniques do not follow this order. The algorithms described in this section use aspects of the previously discussed 3D spatial localization (see Section 4.2) and particle tracking techniques.

### 5.12.1 Stereoscopic Pairing of Particle Tracks

Guezennec *et al.*<sup>231</sup> first suggested that stereo matching (triangulation with two cameras) could be performed for time-resolved data using particle tracks rather than individual particles. The technique tracks particle images in 2D over 5 frames (see Section 5.1) and matches the tracks seen in each camera using triangulation (see Section 4.2.1.2). A cost minimization function quantifies the shortest distance between optical rays for each point in a potential particle track match, as shown in Figure 5.42. The stereo matching algorithm was tested up to 0.0034 ppp and a yield of 98.1% was achieved when exact particle locations were known. When particle locations were identified computationally, the yield decreased to 76.3%.

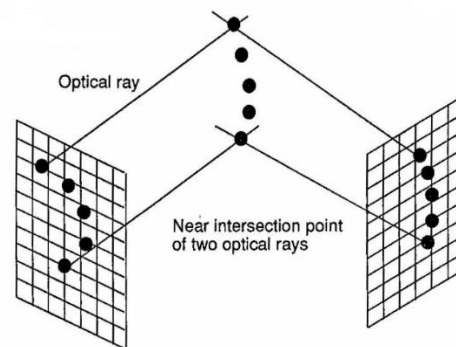


Figure 5.42. Stereo matching for 3D positions of 2D particle tracks.<sup>231</sup>

### 5.12.2 Exclusion of Short Particle Tracks

Another technique was suggested by Kitzhofer and Brücker<sup>130</sup> in order to improve the 3D localization results of their tomo-PTV. They performed a tomographic reconstruction using telecentric lenses (see Section 4.2.2) then applied particle tracking using four frame minimum change in acceleration (see Section 5.1).

The particle tracks were used to separate real particles from ghost particles. Two criteria were used to identify ghost particle tracks: length and consistency. Short ghost trajectories with tracks

that consist of fewer than 3 consecutive frames were identified and deleted. Longer ghost trajectories with inconsistent trajectories or particle image sizes between frames were also deleted. Both of these types of ghost trajectories are shown in Figure 5.43. Notice that the long, real particle track persists over many time steps and the particle image size remains consistent throughout. This technique reduced the erroneous particle tracks from 37% to roughly 5-10% for a particle density of 0.01 ppp, which was the highest particle density tested.

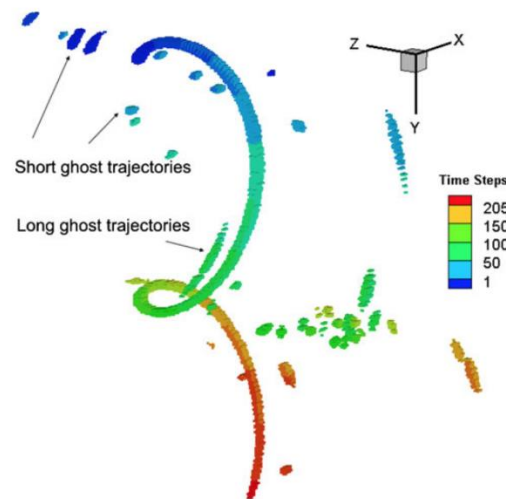


Figure 5.43. Reconstructed trajectories color coded by time step. Short and long ghost trajectories are included.<sup>130</sup>

### 5.12.3 Shake-The-Box

An advanced technique introduced by Schanz *et al.*<sup>232-234</sup> is the shake-the-box method. This method further uses the idea that particle tracks can be used to improve spatial localization results. It relies heavily upon temporal information and tracks from previous time steps to predict particle locations. The predicted particle locations are then “shaken” in space until they agree with a triangulation. By predicting particle locations, computation time and the number of ghost particles is reduced for high particle seeding densities.

The basic scheme for a single time step using the shake-the-box method is as follows:

1. Fit the last  $n$  positions of every tracked particle using an optimal Weiner filter.
2. Predict the position of the Particle in the next time-step  $n+1$  by evaluating the Weiner filter coefficients.
3. “Shake” the particle predictions iteratively until they line up with triangulation.
4. Find new particles entering the measurement zone using triangulation on the remaining particle images.
5. Shake all particles again to correct for residual errors.
6. Remove particles that are leaving the volume or if the intensity falls below a threshold.
7. Iterate steps 4-6 if necessary.
8. Add new tracks for new particles that are identified within the last four consecutive time steps.

These steps describe how the algorithm handles a time step when nearly all the particle trajectories are known. As none of the trajectories are initially known, there exist three phases of the shake-the-box algorithm: initialization, convergence, and converged state.

In the initialization phase, particles are identified using triangulation. Then either TOMO-PIV or CFD is used to give an estimate for the trajectory of each particle. Particles that become part of a track that exists for four time steps are sanity checked and added to the system of tracked particles.

Once tracks have been initialized, the Weiner filter, which can be tuned for different signal and noise spectra, extrapolates each track to the next time step. The predicted position is then compared to the images on each camera by calculating and summing the residuals. The projected image is then shifted in the  $x$  direction in increments of 0.1 pixels and the position with the smallest residual is selected. The same is then done with the  $y$  and  $z$  directions. This process is called shaking and is very similar to the technique used in iterative particle reconstruction (4.2.1.2).

Shaken particles with an intensity below 5% of the average particle intensity are considered lost and deleted. Further, an intensity correction is applied after each shake iteration to get rid of ghost particles and noise.

At each time step, new particle candidates are identified and new tracks consisting of four time steps are searched. The velocity of particle candidates is predicted by calculating a weighted average of nearby particle tracks. If valid, these tracks are added for predicting the next time step.

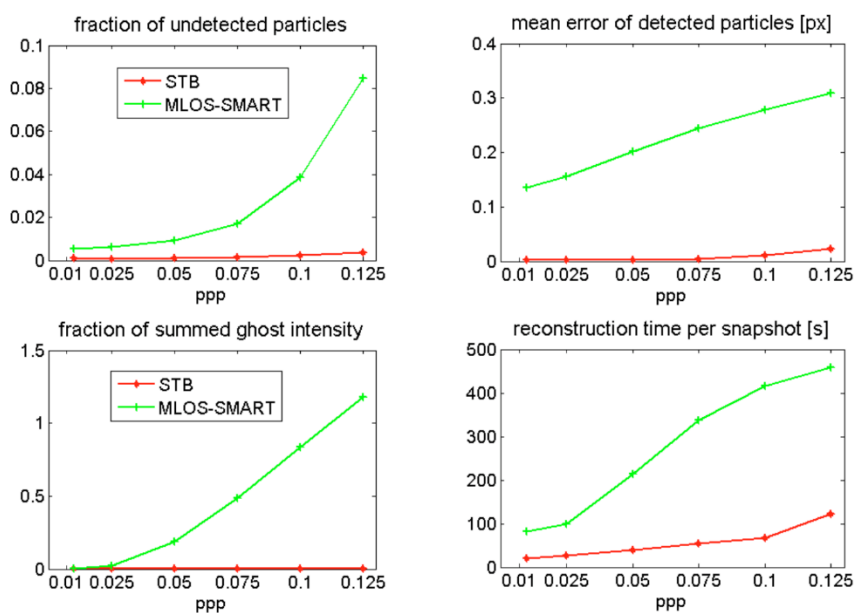


Figure 5.44. Shake-the-box performance vs. a MART algorithm.<sup>234</sup>

In the converged phase, the number of particle tracks should stabilize, as the vast majority of the particles is known and tracked. Convergence can take 20 or more time steps to achieve, depending on seeding density and image noise.

The 3D calibration techniques discussed in Section 4.3.3 can all be applied to STB. Schanz *et al.* used volumetric self-calibration and the non-uniform optical transfer function.

The performance of the technique for noiseless synthetic data is compared to a MLOS-SMART based algorithm in 4.2.2. It outperforms the MART algorithm in every aspect, particularly for high particle densities. Figure 5.45 shows the performance of shake-the-box for noisy simulated data. The seeding density and noise levels increase the number of time steps needed to reach convergence, the localization error, and the number of tracked ghost particles. In all cases, the STB algorithm achieves more than 95% yield for particle tracks given sufficient time steps and less than 1% tracked ghost particles.

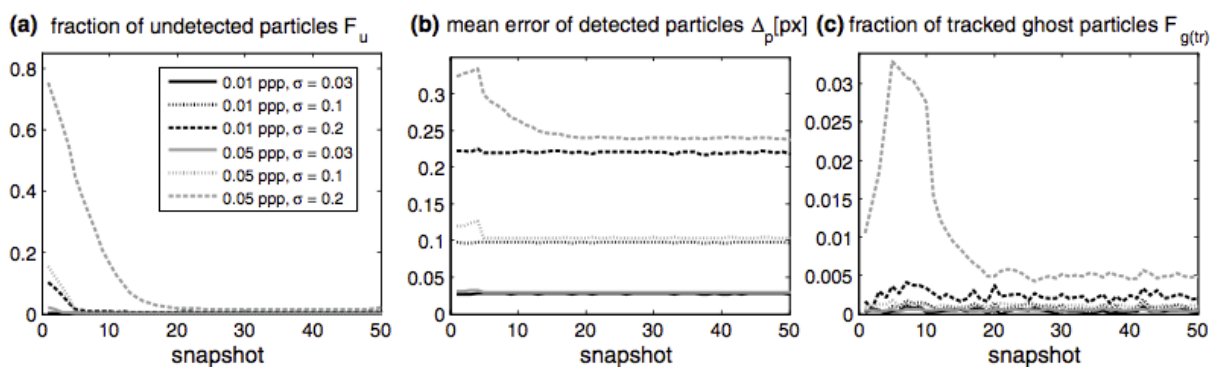


Figure 5.45. Temporal development of STB for 0.01 and 0.05 ppp with different image noise levels,  $\sigma$ , (a) fraction of undetected particles, (b) particle position accuracy, and (c) fraction of ghost particles.<sup>234</sup>

Shake-the-box was applied to an experiment consisting of a water jet with diameter 10 mm and exit velocity 0.43 m/s. The average particle image density was 0.035 ppp. The resulting

isosurfaces of vorticity at one time step superimposed on particle tracks spanning ten time steps are shown in Figure 5.46.

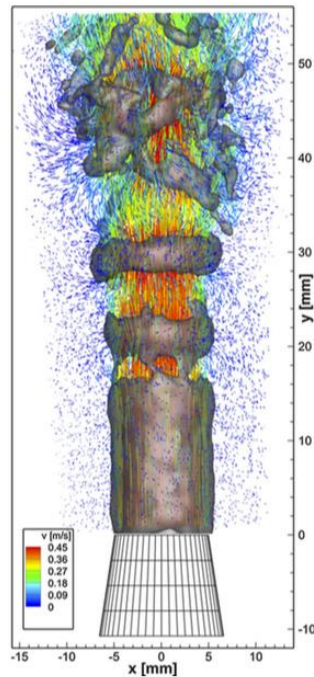


Figure 5.46. Isosurfaces of vorticity with superimposed particle tracks.<sup>234</sup>

## Chapter 6. POST-PROCESSING

The primary goal of PTV algorithms is to provide velocity estimates at discrete, scattered points within a measurement space. In many cases differential quantities, such as vorticity, strain, and acceleration, are of interest. As PIV data is generated on a grid, velocity gradients can be estimated using finite difference methods, which are shown in Figure 6.1. The second order central difference scheme predicts velocity derivatives for gridded data using

$$\frac{\partial u_i}{\partial x_j}(x_k) = \frac{u_i(x_{k+1}) - u_i(x_{k-1}))}{2\delta x_j}, \quad (6.1)$$

where  $x_k$  is the point at which the velocity derivative is being estimated, and  $x_{k+1}$  and  $x_{k-1}$  are the points located  $+\delta x_j$  and  $-\delta x_j$  from point  $x_k$ , respectively.<sup>19</sup> The central difference scheme can also be extended to a fourth-order estimate,

$$\frac{\partial u_i}{\partial x_j}(x_k) = \frac{u_i(x_{k+2}) + 8u_i(x_k) - 8u_i(x_{k-1}) - u_i(x_{k-2})}{12\delta x_j}. \quad (6.2)$$

For simply calculating the vorticity at a point, the 8-point circulation method can also be used. This is done by calculating the circulation around point  $x_k$  using the tangential components of velocity at neighboring points.<sup>235</sup>

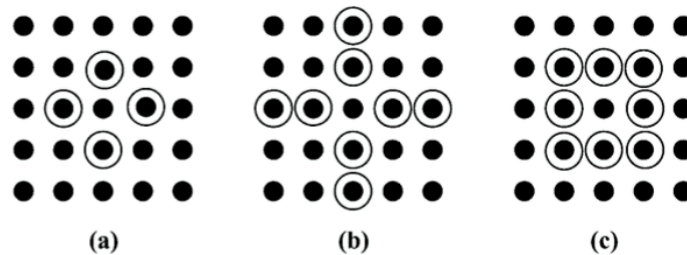


Figure 6.1. Velocity measurement locations used for estimation of spatial derivatives using (a) second order central difference, (b) fourth-order central difference, and (c) 8-pt circulation methods.<sup>235</sup>

These methods of estimating spatial velocity derivatives are used with regularly gridded data, thus PTV vectors are typically interpolated for analysis. Interpolation schemes and their use for calculating differential quantities will be discussed in Section 6.1.

Temporal velocity derivatives can be estimated when velocity measurements are time-resolved. For Lagrangian particle tracks, the acceleration of a tracer particle over a single time-step gives the material derivative

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u}, \quad (6.3)$$

where the acceleration can be integrated towards obtaining the pressure field within a flow. Pressure calculation from PTV data will be discussed in Section 6.2.

## 6.1 INTERPOLATION

One of the first and most common methods for interpolating randomly distributed particle tracking vectors is adaptive Gaussian window interpolation (AGW).<sup>236</sup> The proposed method generates velocity estimates on a grid, and it was suggested to use a grid spacing,  $H$ , equal to 1.24 times the mean particle spacing. Thus, the velocity estimate at point  $x$  is

$$u(x) = \frac{\sum_i \alpha_i u_i}{\sum_i \alpha_i}, \quad (6.4)$$

where  $u_i$  are the known velocities measured at positions  $x_i$ , and

$$\alpha_i = \exp\left(\frac{-|x - x_i|^2}{H^2}\right). \quad (6.5)$$

Agui and Jimenez<sup>236</sup> also proposed a constrained smoothing algorithm. First, an estimate of the interpolation error is obtained using a bootstrapping method. Bootstrapping is performed by taking many randomly selected subsets of the given data and estimating the remaining data points using interpolation. These results are statistically analyzed to obtain an estimate of the interpolation error. Second, this error estimate is then used as a constraint for the smoothing step; the smoothing is done by iteratively replacing each interpolated value with the mean of its four neighbors if it exists within the appropriate one standard deviation error bound. Otherwise, the error bound is selected as the smoothed value. Iterating this formula was shown to quickly converge to a solution.

After interpolating and smoothing, a second-order central difference scheme is used to estimate the velocity derivatives. Figure 6.2 shows the vorticity RMS error and bootstrap estimated variance before and after smoothing versus  $h/\delta$ , where  $h$  is the distance between grid points used for the central difference scheme, and  $\delta$  is the mean particle spacing. The high frequency fluctuations, which appear at small differentiation distances, are smoothed out when a large interpolation window,  $H$ , is used. When the smoothing process is applied, the vorticity RMS is reduced across all ranges of  $h/\delta$  to roughly the same level regardless of interpolation window size.

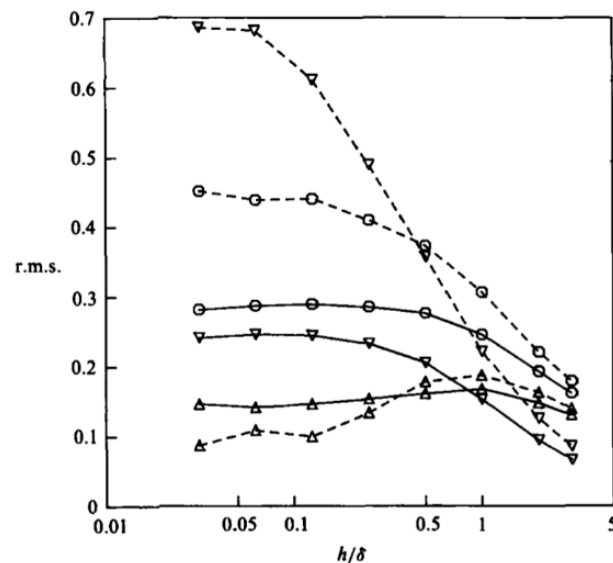


Figure 6.2. RMS of vorticity estimation on interpolation parameters,  $H/\delta$  and  $h/\delta$ . Unsmoothed vorticity (circle); bootstrap predicted variance ( $\nabla$ ); smoothed vorticity ( $\Delta$ ).  $H/\delta=1.24$  (solid line);  $H/\delta=0.62$  (dashed line).<sup>236</sup>

It was later suggested that least squares Taylor polynomials could be used to interpolate data and the coefficients could be directly used to determine differential quantities.<sup>237</sup> Cohn and Koochesfahani<sup>235</sup> tested the effects of vector density and noise on this differentiation scheme. The grid size was selected such that the vector density of the remapped field is the same as the initial irregularly spaced field. Tests were done to compare 2nd, 3rd, and 4th order Taylor polynomial fits with a fit radius  $R$  for interpolation of a simulated Oseen vortex with a grid spacing of  $L/\delta=3.0$ ,

where  $L$  is the vortex core radius. Figure 6.3 shows that the most substantial interpolation error is present as a bias within the vortex core. This bias is a maximum for the second and third-order Taylor polynomials with large fit radii. The minimum bias error was found using a 4th order fit with  $R/\delta=3$ , although its advantage was minimal compared to a 2nd order fit with  $R/\delta=2$ . The random component of error was small compared to the bias error, and larger fit radius fits performed better with more noise and worse with less noise; therefore, the authors suggested using a 2nd order fit with  $R/\delta=2$  for simplicity.

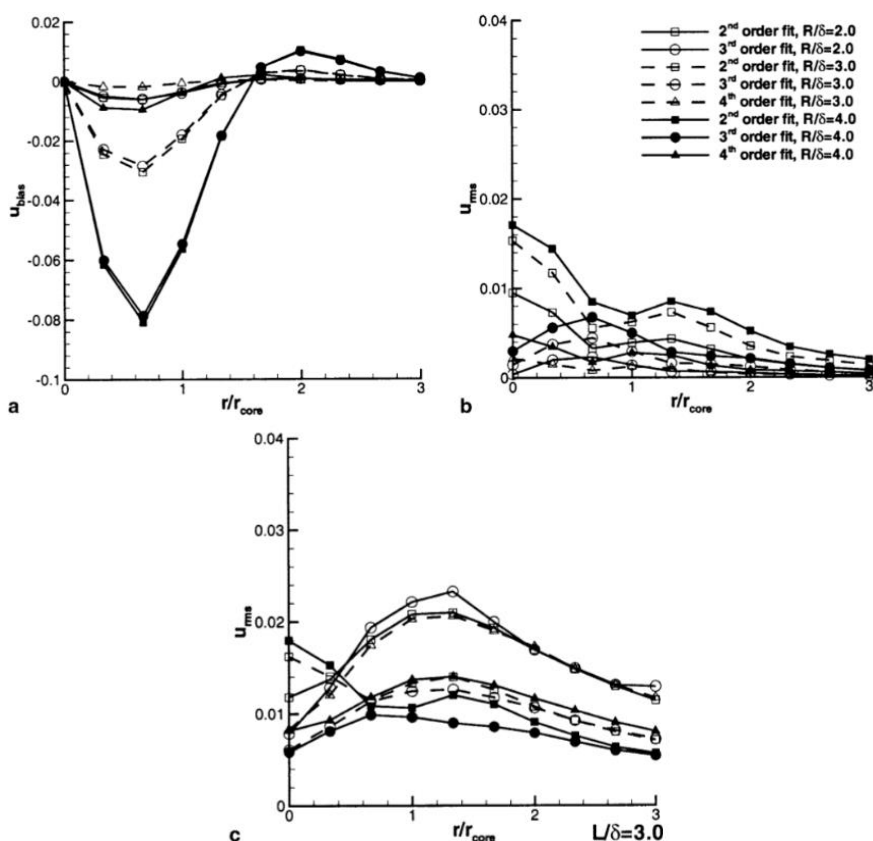


Figure 6.3. Accuracy of interpolated velocity field as (a) bias error with 0% noise, (b) random error with 0% noise, and (c) random error with 6% noise.<sup>235</sup>

Using this interpolation method, Cohn and Koochesfahani<sup>235</sup> compared the previously described 2nd order central difference, 4th order central difference, and eight-point circulation

methods with a direct differentiation of a 4th order interpolation polynomial. The same Oseen vortex flow used for the interpolation comparison was used for analysis, shown in Figure 6.4.

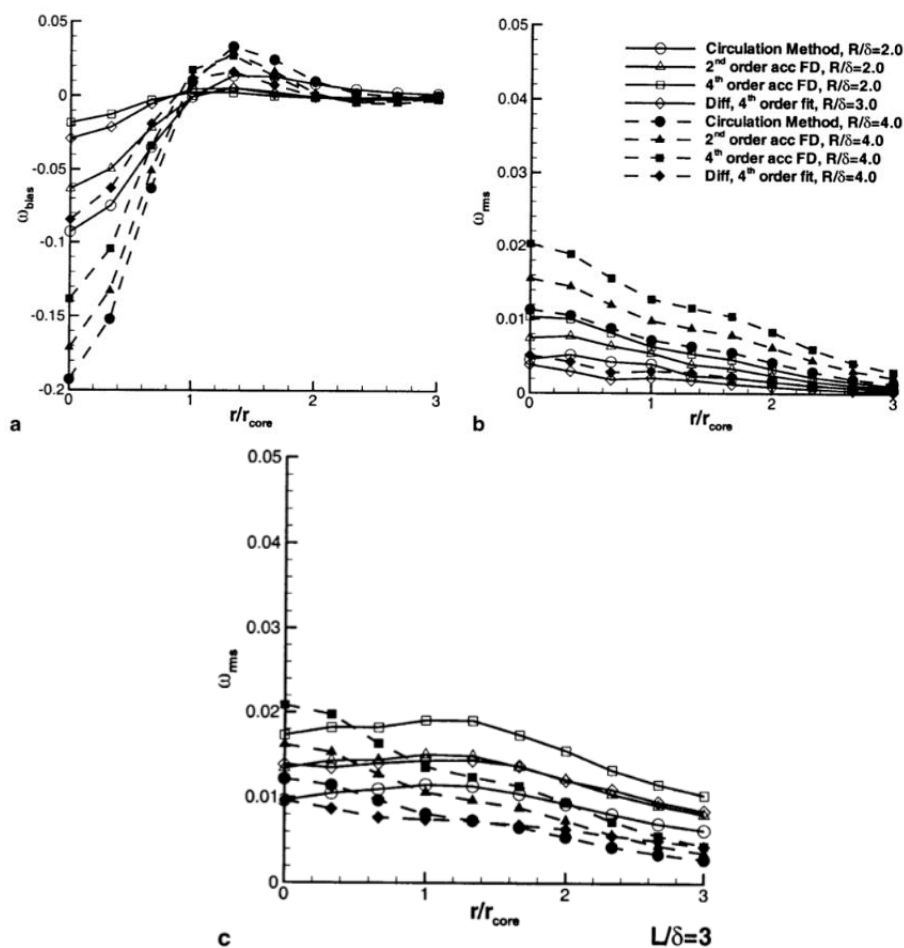


Figure 6.4. Accuracy of out-of-plane vorticity field computed for different values of  $R/\delta$  using four different differentiation methods; (a) bias error, (b) random error with 0% added noise, and (c) random error with 6% added noise.<sup>235</sup>

The 4th order central difference and polynomial differentiation using the smallest interpolation radius yielded the lowest bias error, however, these also gave the greatest random errors in the presence of noise. Again, the difference in bias error is more substantial than the difference of RMS, so either the fourth-order finite difference method or the polynomial differentiation technique was considered best. For the sake of simplicity and computational efficiency, the authors suggested using the fourth-order finite difference method, as both the bias

and random components of error were less than 2% of the maximum vorticity in the presence of 6% noise.

Ido *et al.*<sup>238</sup> proposed solving ellipsoidal equations to interpolate data points onto a regular grid and iteratively correct them according to the continuity equation. The first technique solves the Laplace (LER) equation

$$\nabla^2 u = 0, \quad (6.6)$$

with PTV vectors as fixed points, which results in a linear interpolation onto any desired grid. Using this Laplace equation, the authors found that the first derivatives of the interpolated velocity field are not smooth. In order to have smooth derivatives, the authors suggested a second technique that uses the biquadratic-ellipsoidal (BER) equation

$$\nabla^4 u = 0, \quad (6.7)$$

which gives third-order spatial continuity, yielding smooth derivative estimates. A velocity correction potential (VCP) was also introduced to iteratively adjust interpolated vectors to satisfy the continuity equation. An improve velocity correction potential (IVCP) was also proposed, in which the measured PTV vectors can be adjusted in order to improve spatial smoothness and reduce the impact of noise.

The LER, BER, inverse distance rearrangement (IDR), and AGW interpolation methods were tested using a Taylor-Green vortex flow field. LER and BER were tested both with and without the smoothing corrections. Figure 6.5 shows the correlation coefficient of vorticity and the relative error for vorticity, which is simply the vorticity error normalized by the largest vorticity in the field, versus the vector density. The performance of all algorithms improves with an increase in the number of vectors per vortex. Thus the authors stated that identifying the number of vectors

per vortex required to achieve a normalized vorticity error below 10% is a good measure of the performance of an algorithm. LER, IDR, and AGW all required at least 50 vectors per vortex to achieve the desired error level, while BER required only 14. With the addition of IVCP smoothing, the BER algorithm achieved less than 10% error with only 11 vectors per vortex.

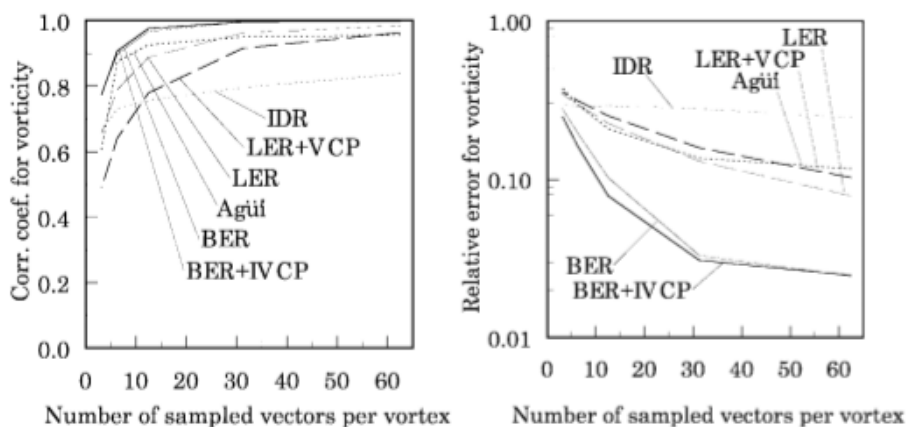


Figure 6.5. Plot of the vorticity correlation coefficient (left) and normalized vorticity error (right) of the proposed ellipsoidal equation algorithms with and without smoothing.<sup>238</sup>

In a later paper, Ido and Murai<sup>239</sup> described a technique for interpolation using the hexagonal-ellipsoidal equation rearrangement (HER),

$$\nabla^6 u = 0 \quad (6.8)$$

In order to obtain a unique solution for the sixth order differential equation, the LER and BER interpolations need to be performed and used as a computational initial condition. It was shown that the HER algorithm was capable of reconstructing a more valid energy spectrum than the BER and LER algorithms in the case of isotropic turbulence.

Vedula and Adrian<sup>240</sup> suggested using a linear stochastic estimator for interpolating velocity onto a grid. It was suggested that an optimal estimate will have the minimum mean square error when compared to the measured point velocities. This is done by defining a Cartesian component of the filtered velocity field as

$$\tilde{u}_i(\mathbf{x}, t) = \int h_{ij}(\xi, \mathbf{x}) u_j(\xi, t) d\xi, \quad (6.9)$$

where  $\mathbf{x}$  is the position vector,  $h_{ij}$  denotes the filter impulse response function, and  $u_j$  is a Cartesian component of the unfiltered (measured) velocity field. In the case of incompressible flow, the solenoidal property was imposed while finding the optimal filter, so that the divergence of the filtered velocity field is zero. By expanding the filtered estimate to a first order power series, the velocity and its spatial derivatives are estimated such the divergence is zero and the mean square error of the estimate with respect to the measured point velocities is a minimum.

In order to evaluate the performance of this interpolation algorithm, DNS was used to generate incompressible, isotropic turbulence on a  $256^3$  grid. The domain was subdivided into non-overlapping cubic sub-domains, in which the velocity was sampled at  $M$  scattered points. A non-dimensional data density was defined as  $\psi \equiv M \lambda^3 / L_s^3$ , where  $\lambda$  is the Taylor micro-scale of the flow and  $L_s$  is the side length of each cubic sub-domain. Additionally, a Fourier cut-off filter of width  $\Delta_c \equiv \pi/k_c$ , where  $k_c$  is the cutoff wavenumber, was applied to the simulated data. The resulting mean squared error from this optimal interpolator,  $e$ , was normalized by the variance of the filtered velocity field,  $\sigma_{\tilde{u}}^2$  and is shown as a function of filter width in Figure 6.6. For comparison using the highest normalized data density, the normalized error was also determined using AGW interpolation, shown by the dotted line. Applying the optimal filter width for this data density resulted in a normalized error of 4% while AGW resulted in 14% error. The filter width corresponding to a minimum normalized error appears to decrease with increase in data density.

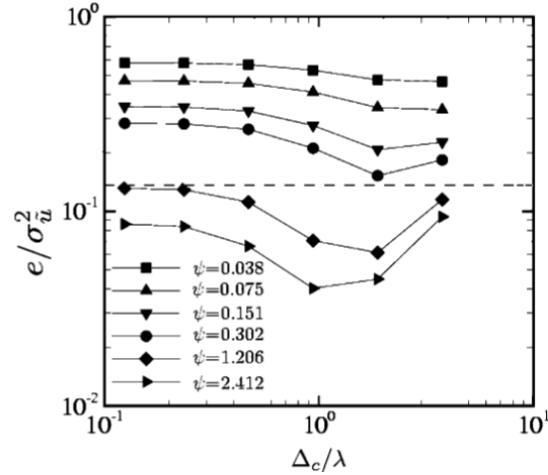


Figure 6.6. Normalized mean square error versus normalized cut-off filter widths at various data densities,  $\psi$ , and a sub-volume size of  $L_v/\lambda=3.756$ . The dashed line shows the error for AGW for  $\psi=2.412$ .<sup>240</sup>

Casa and Krueger<sup>241</sup> proposed using Gaussian weighted radial basis functions (RBF) for interpolation. An advantage of the RBF interpolation is that it is infinitely smooth and can be used to find smooth velocity gradients through differentiation. Additionally, boundary conditions and constraints, such as incompressibility, can be introduced. The general form of Gaussian RBFs is

$$u_j(\mathbf{x}) = \sum_{i=1}^N b_{ji} \exp\left[\frac{-\|\mathbf{x} - \boldsymbol{\mu}_i\|^2}{2\sigma_{ji}^2}\right], \quad (6.10)$$

where  $u_j$  is the function approximation,  $\mathbf{x}$  is the position vector,  $N$  is the number of RBFs used,  $b_{ij}$  are RBF weights,  $\boldsymbol{\mu}_i$  are the locations of the RBF centers, and  $\sigma_{ji}$  are the widths of the Gaussian RBFs. The  $\boldsymbol{\mu}_i$  were fixed within the fluid volume while  $b_{ij}$  and  $\sigma_{ji}$  were determined using nonlinear least squares optimization. The two parameters were optimized independently in order to reduce computation time.

The interpolation algorithm was applied to an experimental 3D data set of an unbounded vortex ring. Both the AGW and RBF interpolation methods were used to identify the vortex center,

axis, and radius. The AGW vorticity estimates were obtained using central differences, while the RBF estimates were found via differentiation of  $u_j(\mathbf{x})$ . The impact of noise on the AGW determination of vorticity is shown in Figure 6.7. Both methods managed to identify the vortex ring, but the RBF is shown to be less noisy while allowing for computation of various properties on a much finer grid than AGW without repeating interpolation. It should also be noted that the AGW interpolation shows the vortex ring to be more symmetric, though it is noisier, while the RBF results shows the vortex right to be quite asymmetric.

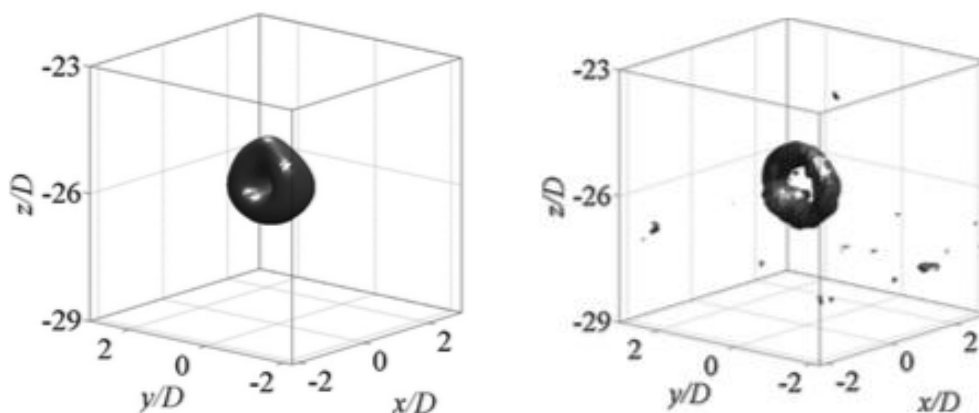


Figure 6.7. Isovorticity surfaces using RBF at  $\omega_n=0.5$  (left) and AGW at  $\omega_n=0.25$  (right).<sup>241</sup>

The 'FlowFit' algorithm was proposed by Gesemann<sup>242</sup> and used in STB by Schanz *et al.*<sup>234</sup> The algorithm models each component of the flow field using a weighted sum of three dimensional and evenly spaced quadratic B-splines. In order to evaluate the flow field at arbitrary coordinates, the velocity and acceleration of particles must be known at discrete locations. Systems of equations are defined for each known velocity measurement, and additional equations used to regularize the flow penalize nonzero divergence and nonzero curvatures. The conjugate-gradient algorithm is used to iteratively solve the systems of equations. Then the flow field is sampled on a regular grid, including both velocity and spatial derivative information. Typically 5-20 grid points are generated per tracer particle.

Schneiders *et al.*<sup>243,244</sup> proposed the vortex in cell (VIC) and vortex in cell plus (VIC+) methods which similarly interpolate vectors onto a high resolution grid. While VIC uses the vorticity transport equation to interpolate data, the VIC+ method includes the material derivative. The authors suggest generating 64 grid nodes per tracer particle for the VIC+ interpolation. RBFs are used with a window size,  $\sigma=1.1h$ , where  $h$  is the grid node spacing. The vorticity,  $\boldsymbol{\omega}_h$ , is defined on the grid points using RBFs and is related to velocity,  $\mathbf{u}_h$ , by

$$\nabla^2 \mathbf{u}_h = -\nabla \times \boldsymbol{\omega}_h, \quad (6.11)$$

with boundary conditions on velocity. Second-order central differences are used for calculating spatial derivatives, except at the boundaries, where first-order single-sided differences are used. The vorticity transport equation is used to calculate the temporal vorticity derivative

$$\frac{\partial \boldsymbol{\omega}_h}{\partial t} = (\boldsymbol{\omega}_h \cdot \nabla) \mathbf{u}_h - (\mathbf{u}_h \cdot \nabla) \boldsymbol{\omega}_h. \quad (6.12)$$

With this, the temporal velocity derivative is calculated from the temporal vorticity derivative using

$$\nabla^2 \frac{\partial \mathbf{u}_h}{\partial t} = -\nabla \times \frac{\partial \boldsymbol{\omega}_h}{\partial t}, \quad (6.13)$$

with boundary conditions on the temporal velocity derivative. This is then integrated to provide  $\frac{\partial \mathbf{u}_h}{\partial t}$ , which is used to calculate the velocity material derivative, defined on the grid as

$$\frac{D\mathbf{u}_h}{Dt} = \frac{\partial \mathbf{u}_h}{\partial t} + (\mathbf{u}_h \cdot \nabla) \mathbf{u}_h. \quad (6.14)$$

Optimization is performed to minimize the difference between the original velocity and acceleration measurements and estimates using linear interpolation from the computational grid.

The VIC and VIC+ interpolation algorithms were compared with AGW, linear interpolation (LIN), and PIV results for a simulated turbulent boundary layer. The normalized turbulent statistics are given in Figure 6.8. PIV has the greatest damping of turbulent fluctuations, while VIC+ most closely matches the DNS results. VIC+ has a substantial improvement even over the standard VIC technique; however, the computation time for VIC+ was approximately four times longer than for PIV cross-correlation.

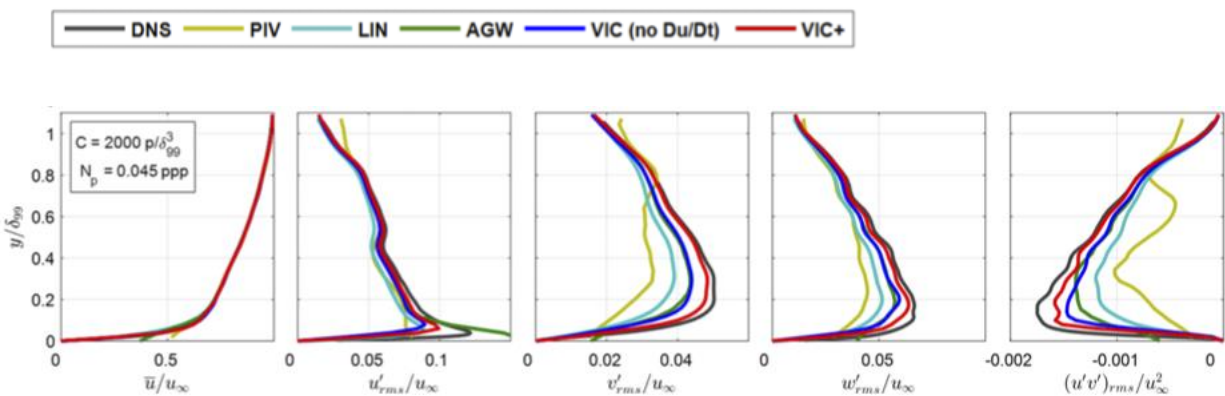


Figure 6.8. Turbulence statistics from different velocity field reconstructions in a turbulent boundary layer.<sup>244</sup>

## 6.2 PRESSURE CALCULATION

Determining a fluid pressure field can help identify flow structures, calculate surface forces, and give insight into sound generation. Velocimetry results can be used to determine the pressure gradient using the momentum equation

$$\nabla p = -\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u}. \quad (6.15)$$

In the case of inviscid flow, the dissipation term can be neglected, giving the pressure gradient as a function of only density and the material derivative. For incompressible flow, the density can be

considered constant; if flow is compressible, the equation can be rewritten to eliminate density as an independent variable assuming adiabatic flow.

Interpolation techniques like FlowFit and VIC+ provide material derivative estimates on a fine grid using time-resolved Lagrangian particle tracks. Thus, once boundary conditions have been defined, the pressure field can be determined using path integration. The path integration is generally averaged over many paths to reduce the impact of noise.<sup>245</sup>

Alternatively, the pressure can be presented in the form of the Poisson equation

$$\nabla^2 p = \nabla \cdot \left( -\rho \frac{D\mathbf{u}}{Dt} + \mu \nabla^2 \mathbf{u} \right). \quad (6.16)$$

The Poisson equation can also be modified using the Reynolds-averaged approach given by van Oudheusden<sup>245</sup>

$$\nabla^2 \bar{p} = -\rho \nabla \cdot (\bar{\mathbf{u}} \cdot \nabla) \bar{\mathbf{u}} - \rho \nabla \cdot \nabla \cdot (\overline{\mathbf{u}'\mathbf{u}'}). \quad (6.17)$$

This modification allows for ensemble averaging to give higher spatial resolution and turbulent statistics. From this, the Poisson equation can be solved iteratively once boundary conditions have been defined.<sup>246</sup>

A technique proposed by Huhn *et al.*<sup>247</sup> suggests that the pressure gradient equation can be solved using fast fourier transform (FFT) integration. The three components of the pressure gradient were introduced as

$$\partial_x P = -\rho a_x, \quad (6.18)$$

$$\partial_y P = -\rho a_y, \text{ and} \quad (6.19)$$

$$\partial_z P = -\rho a_z. \quad (6.20)$$

This is converted to Fourier space such that

$$\tilde{P}(\mathbf{k}) = \frac{\mathbf{k} \cdot \nabla P}{i|\mathbf{k}|^2} = \frac{k_x \partial_x P + k_y \partial_y P + k_z \partial_z P}{i(k_x^2 + k_y^2 + k_z^2)}, \quad (6.21)$$

and then converted back to normal space

$$P'(\mathbf{x}) = \mathcal{F}^{-1} \tilde{P}(\mathbf{k}), \quad (6.22)$$

where  $\mathbf{k}$  is the wave number vector. For  $\mathbf{k}=0$ , there exists a singularity, so the constant component of the pressure gradient is set to zero. Therefore, the pressure gradient must be reconstructed as

$$P(\mathbf{x}) = P'(\mathbf{x}) + \langle \partial_x P \rangle x + \langle \partial_y P \rangle y + \langle \partial_z P \rangle z, \quad (6.23)$$

where  $\langle \cdot \rangle$  is the spatial mean over the entire domain. The authors suggest enforcing continuous pressure through the boundaries mirroring pressure across the boundaries. The proposed FFT pressure integration algorithm is completed by adding a constant offset pressure to obtain absolute pressure.

Neeteson and Rival<sup>248,249</sup> proposed pressure-field extraction directly from unstructured flow data instead of interpolating to a grid. The inviscid form of the Poisson equation was used and an unstructured mesh was generated using Voronoi tessellation and Delaunay tessellation. For a point  $i$ , the Voronoi cell is shown with neighbors  $j$  in Figure 6.9. Two length measures are used to define the mesh: the distance from point  $i$  to point  $j$ ,  $h_{ij}$  and the shared side length of the Voronoi cell,  $s_{ij}$ . In 3D space,  $s_{ij}$  is the shared cell face area, and additionally a normal vector  $\hat{\mathbf{n}}_{ij}$  can be determined for every connection. Near the boundary, some Voronoi cell faces will be infinitely large, thus these unbounded connections are ignored and boundary conditions are imposed instead.

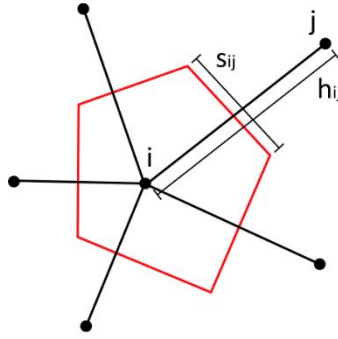


Figure 6.9. Example of Voronoi tessellation in a two-dimensional space.<sup>248</sup>

Three operators are needed to numerically solve the Poisson equation: the gradient operator, the divergence operator, and the Laplace operator. An expression for evaluating the gradient of a scalar field in an unstructured mesh is

$$\nabla \psi_i = \frac{\sum_j [(\psi_i + \psi_j) s_{ij} \hat{\mathbf{n}}_{ij}]}{\frac{1}{D} \sum_j (s_{ij} h_{ij})}, \quad (6.24)$$

and the divergence of a vector is given as

$$\nabla \cdot \mathbf{F}_i = \frac{\sum_j [s_{ij} \hat{\mathbf{n}}_{ij} \cdot (\mathbf{F}_i + \mathbf{F}_j)]}{\frac{1}{D} \sum_j (s_{ij} h_{ij})} \quad (6.25)$$

where  $D$  is the dimension of the space. The Laplace operator of a scalar is then given as

$$\nabla^2 f_i = \frac{\sum_j \left( \frac{s_{ij}}{h_{ij}} f_j \right) - f_i \sum_j \left( \frac{s_{ij}}{h_{ij}} \right)}{\frac{1}{2D} \sum_j (s_{ij} h_{ij})} \quad (6.26)$$

These equations are used to iteratively solve the Poisson equation and enforce Neumann boundary conditions. The iterative integration technique used is known as the successive over-relaxation

algorithm, which can be used to stabilize a diverging iterative process or speed up convergence of a slowly converging iterative process.

An experimental comparison of these pressure field reconstruction techniques was done by van Gent *et al.*<sup>250</sup> using a simulated flow over an axisymmetric step flow using zonal detached eddy simulation (ZDES). The model main body has a diameter of 50 mm and after the 15 mm step height, the afterbody has a diameter of 20 mm. The free stream Mach number was 0.7 and the total pressure and temperature were 200 kPa and 285 K, respectively. The measurement domain was 60 mm x 24 mm x 4 mm. Both clean and noisy data sets were processed using Shake-the-Box. The RMS velocity errors in clean and noisy data were 0.01 and 0.04 voxels, respectively.

The PTV algorithms used include FlowFit A (FFA), FlowFit B (FFB), VIC+, iterative least-square pseudo-tracking (ILAG), and Voronoi-based pressure integration (VOR). The FFA algorithm uses quadratic B-splines for interpolation and performs pressure integration in the Fourier space. FFB generates cubic B-spline functions for the scalar pressure field and estimates its parameters by minimizing a weighted sum of the squared error with measured pressure gradients at particle locations, squared error at the boundary conditions, and squared curvatures of the pressure function. This minimization is a linear least-squares problem that is solved iteratively. The VIC+ algorithm iteratively solves the Poisson equation using the interpolated material derivatives. ILAG uses a first-order least-square fit of velocity to create imaginary particle tracks to determine the material derivative throughout the measurement volume. Spatial integration of the material derivative is then performed to determine the pressure field.

In order to compare the algorithms, a global RMS error was defined as

$$\mathcal{E}_{RMS}^{global} = \sqrt{\frac{1}{N_t N_p} \sum_{i=1}^{N_t} \sum_{j=1}^{N_p} \left( (C_p)_{i,j} - (C_{p,ref})_{i,j} \right)^2} \quad (6.27)$$

where  $N_p$  is the number of gridpoints and  $N_t$  is the number of time instances. The global errors for the tested techniques are shown in Table 16. Note that the error is expressed in 0.01  $C_p$ , which is equivalent to a percentage of the freestream dynamic pressure. The global level of pressure fluctuations is 4% of the local dynamic pressure, thus a 1% error corresponds to 25% of the global pressure fluctuations. The results show that the global pressure error was lowest using VIC+, with only a 0.63% error for noisy data. The FlowFit techniques performed nearly as well as VIC+, although the Voronoi pressure integration method had an error of 1.63%, more than double that of the VIC+ and FlowFit methods.

Table 16: Global pressure RMS error for PTV pressure extraction techniques for clean and noisy data.<sup>250</sup>

Pressure Field Reconstruction	Global Pressure Error ( $10^{-2}$ )	
	Clean	Noisy
<b>VIC+</b>	0.68	0.63
<b>FFA</b>	0.71	0.74
<b>FFB</b>	0.72	0.76
<b>ILAG</b>	0.88	0.89
<b>VOR</b>	1.71	1.63

The spatial spectra of pressure reconstructions were compared by taking the pressure measurements in straight, horizontal lines across the measurement domain and calculating the Fourier transform. Forty horizontal lines were used for each measurement technique and the Fourier coefficients ( $E_{pp}$ ) were averaged and scaled by the ratio of the window size and the wavelength ( $WS/\lambda$ ). Figure 6.10 shows the pressure fluctuation intensities at wavelengths up to the Nyquist frequency ( $WS/\lambda=2$ ). The left figure shows the pressure-field spectra from PIV measurements, while the right figure shows the pressure-field spectra from PTV measurements. In

comparison to the simulated results (ZDES), it can be seen that the PIV approaches are able to generally capture the relative spatial spectral variations, though not matching their amplitudes well. The PTV approaches, on the other hand, are better able to capture the spectral behavior well until the algorithm's failure to resolve the smallest flow features is reached. For example, the VOR, FFB, and ILAG approaches reach this limit at  $WS/\lambda = 0.7$ ,  $1.0$ , and  $1.25$ , respectively. The figure also shows that ZDES simulations may have reached its limit at  $WS/\lambda = 1.25$ , where its curve flattens out and its fluctuation begins to increase. The FFA and VIC+ approaches, however, show that they are not only able to capture the smallest scales' spectra quite well, but that they extend beyond the limitations of the ZDES simulations.

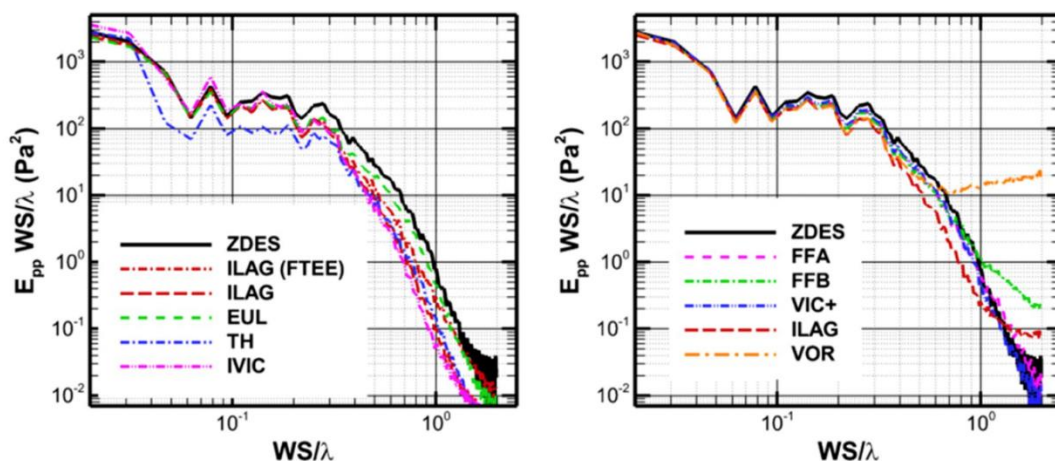


Figure 6.10. Comparison of spatial spectra of pressure fluctuations along 40 horizontal lines within the shear layer calculated using PIV (left) and PTV (right) algorithms.<sup>250</sup>

## Chapter 7. CONCLUSIONS

In conclusion, Particle Tracking Velocimetry is becoming a powerful quantitative flow visualization technique, capable of outputting high-resolution vector fields that detail flow features such as those shown in Figure 5.46. Recent methodologies in triangulation, tomo-PTV, and Shake-the-Box methods have been able to process images with 0.035-0.05 ppp with localization

uncertainties as low as 0.1 px and 0.15 px for in-plane and out-of-plane errors, respectively,<sup>87</sup> and most recently as high 0.125 ppp with localization uncertainties of 0.3 px and 0.3 px for in-plane and out of plane errors, respectively.<sup>234</sup> Advances in laser and camera technologies now allow for volume illumination and image acquisitions up to 20kHz, thereby allowing for time-resolved particle velocimetry techniques. Such time-resolved methods are therefore able to provide flow accelerations, and consequently pressure distributions within flow, which are exciting capabilities, though at present, the uncertainties of these approaches tend to be rather high.<sup>250</sup> In addition, with the consequential increased data rate collection, a typical bottleneck for current PTV systems is therefore the increased computational time required to process acquired data.

The sources of error in PTV experiments come from each of the steps involved. First, error inherently exists in the difference between the tracer particle motion and the fluid in which it is suspended. Additionally, in 3D-PTV, the error associated with proper particle identification and spatial localization, which can be amplified by noise, compile. These errors then propagate throughout velocity calculations, but careful calibration and implementation of the most advanced particle locating algorithms can minimize the reconstruction errors. Finally, particle matching errors can cause loss of data and incorrect velocity vectors. The Shake-the-Box approach is to resolve these by combining the individual steps for 3D reconstruction and tracking to ensure particle tracks over at least four sequential time steps.

While PTV has flourished significantly, more efficient and accurate systems with increased temporal and spatial resolution can be required for analysis of complex flow structures. These limitations can be removed with imaging sensors capable of higher frame rates and higher resolution, as well as illumination sources that have greater illumination intensities with in addition to the higher repetition rates so as to allow for larger volume imaging. Furthermore, improving

localization and tracking algorithms can allow for higher density images to be analyzed with increased accuracy as well as reducing ghost particles. Within the context of time-resolved PTV, this could also lead to more accurate pressure measure measurements. With these improvements, 3D-PTV will lead current researchers into a future where high-resolution and accurate 3D experimental data can be obtained that would allow us to pursue scientific understanding of three-dimensional fluid flows.

## REFERENCES

- 
- <sup>1</sup> Gharib, M., Kremers, D., Koochesfahani, M.M., Kemp, M., "Leonardo's vision of flow visualization," *Experiments in Fluids* (2002) 33: 219-223.
  - <sup>2</sup> Prandtl, L., Über Flüssigkeitsbewegung bei sehr kleiner Reibung, Proc. *Verhandlungen des III. Internationalen Mathematiker-Kongresses, Heidelberg, 1904*, Teubner, Leipzig, pp. 404-491.
  - <sup>3</sup> Fage, A., Townend, H.C.H., "An examination of turbulent flow with an ultramicroscope," *Proceedings of the Royal Society of London Series A-Containing Papers of a Mathematical and Physical Character* (1932) 135: 656-677.
  - <sup>4</sup> Adrian, R.J., "Particle-imaging techniques for experimental fluid mechanics" *Annual review of fluid mechanics*, (1991), 23: 261-304.
  - <sup>5</sup> Dudderar, T.D. Simpkins, P.G., "Laser Speckle Photography in a Fluid Medium," *Nature* (1977) 270: 45-47.
  - <sup>6</sup> Simpkins, P.G., Dudderar, T.D., "Laser Speckle Measurements of Transient Benard Convection," *Journal of Fluid Mechanics* (1978) 89: 655.
  - <sup>7</sup> Meynart, R., "Instantaneous Velocity-field Measurements in Unsteady Gas-Flow by Speckle Velocimetry," *Applied Optics* (1983) 22: 535-540.
  - <sup>8</sup> Pickering, C.J.D., Halliwell, N.A., "Speckle photography in fluid-flows - signal recovery with 2-step processing," *Applied Optics*, 1984 Vol 23, pages 1128-1129.
  - <sup>9</sup> Adamczyk, A.A. & Rimai, L. "2-Dimensional particle tracking velocimetry (PTV): Technique and image processing algorithm," *Experiments in Fluids* (1988) 6: 373.
  - <sup>10</sup> Dimotakis, P.E., Debussy, F.D., Koochesfahani, M.M., "Particle streak velocity field measurements in a two-dimensional mixing layer," *Physics of Fluids* (1981) 24: 995.
  - <sup>11</sup> Kähler, C.J., Scharnowski, S., Cierpka, C., "On the resolution limit of digital particle image velocimetry," *Experiments in Fluids* (2012) 52: 1629-1639.
  - <sup>12</sup> Adrian RJ (1997) Dynamic ranges of velocity and spatial resolution of particle image velocimetry. *Meas Sci Technol* 8:1393. doi:10.1088/0957-0233/8/12/003
  - <sup>13</sup> Buchhave, P., "Particle Image Velocimetry - Status and Trends," *Experimental Thermal and Fluid Science* (1992) 5: 586-604.
  - <sup>14</sup> Grant, I., "Particle image velocimetry: A review," *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science* (1997) 211: 55-76.
  - <sup>15</sup> Raffel, M., Willert, C., Kompenhans, J., *Particle Image Velocimetry: A Practical Guide*, Springer-Verlag Berlin Heidelberg, Germany (1998).
  - <sup>16</sup> Gharib, M. & Dabiri, D. (2000) *An Overview of Digital Particle Image Velocimetry in Flow Visualization: Techniques and Examples*, Eds. Smits, A. & Lim, T. T. London, Imperial College Press, March.
  - <sup>17</sup> Adrian, R.J., "Twenty years of particle image velocimetry," *Experiments in Fluids* (2005) 39: 1-54.
  - <sup>18</sup> Dabiri, D. (2006) "Cross-Correlation Digital Particle Image Velocimetry - A Review," *Turbulencia*, Eds. Freire, A.S., liha A., Breidenthal B., ABCM pgs. 155-199.
  - <sup>19</sup> Adrian, R.J., Westerweel, J., *Particle Image Velocimetry*, Cambridge University Press, New York, New York (2011).
  - <sup>20</sup> Gao, Q., Wang, H.P., Shen, G.X., "Review on development of volumetric particle image velocimetry," *Chinese Science Bulletin* (2013) 58: 4541-4556.
  - <sup>21</sup> Westerweel, J., Elsinga G.E., Adrian R.J. "Particle Image Velocimetry for Complex and Turbulent Flows," *Annual Review of Fluid Mechanics* (2013) 45: 409

- 
- <sup>22</sup> Beresh, S.J., Henfling, J.F., Spillers, R.W., “Postage-Stamp PIV:’ Small Velocity Fields at 400 kHz for Turbulence Spectra Measurements,” *AIAA SciTech Forum* 9-13 January 2017, Grapevine, Texas.
- <sup>23</sup> Willert, C. E. (2015). High-speed particle image velocimetry for the efficient measurement of turbulence statistics. *Experiments in Fluids*, 56(1), 17–17.
- <sup>24</sup> Maas, H.G., Gruen, A., Papantoniou, D., "Particle tracking velocimetry in three-dimensional flows Part 1. Photogrammetric determination of particle coordinates," *Experiments in Fluids* (1993) 15:133-146.
- <sup>25</sup> Melling, A., "Tracer particles and seeding for particle image velocimetry," *Measurement Science and Technology*, (1997) 8: 1406-1416.
- <sup>26</sup> Chen, C.J., Emrich, R.J., “Investigation of the Shock-Tube Boundary Layer by a Tracer Method,” *Physics of Fluids* (1963) 6: 1.
- <sup>27</sup> Mei R (1994) Flow due to an oscillating sphere and an expression for unsteady drag on the sphere at finite Reynolds number. *J Fluid Mech* 270:133 174
- <sup>28</sup> Mei, R., “Velocity fidelity of flow tracer particles,” *Experiments in Fluids* (1996) 22: 1-13.
- <sup>29</sup> TOSCHI F. AND BODENSCHATZ E., “LAGRANGIAN PROPERTIES OF PARTICLES IN TURBULENCE,” *ANNU. REV. FLUID MECH.* 41, 375 (2009).
- <sup>30</sup> CALZAVARINI E., VOLK R., BOURGOIN M., LEVEQUE E., PINTON J. F., AND TOSCHI F., “ACCELERATION STATISTICS OF FINITE-SIZED PARTICLES IN TURBULENT FLOW: THE ROLE OF FAXÉN FORCES,” *J. FLUID MECH.* 630, 179 (2009).
- <sup>31</sup> CALZAVARINI E., KERSCHER M., LOHSE D., AND TOSCHI F., “DIMENSIONALITY AND MORPHOLOGY OF PARTICLE AND BUBBLE CLUSTERS IN TURBULENT FLOW,” *J. FLUID MECH.* 607, 13 (2008).
- <sup>32</sup> BEC J., BIFERALE L., BOFFETTA G., CELANI A., CENCINI M., LANOTTE A., MUSACCHIO S., AND TOSCHI F., “ACCELERATION STATISTICS OF HEAVY PARTICLES IN TURBULENCE,” *J. FLUID MECH.* 550, 349 (2006).
- <sup>33</sup> Mathai, V., Calzavarini, E., Brons, J., Sun, C., Lohse, D., “Microbubbles and Microparticles are Not Faithful Tracers of Turbulent Acceleration,” *Physical Review Letters* (2016) PRL 117.
- <sup>34</sup> Pedocchi, F., Martine, J.E., Garcia, M.H., “Inexpensive fluorescent particles for large-scale experiments using particle image velocimetry,” *Experiments in Fluids* (2008) 45: 183-186.
- <sup>35</sup> Wohl CJ, Kiefer JM, Petrosky BJ\*, Tiemsin PI, Lowe KT, Maisto PM\* and Danehy PM 2015 “Synthesis of Fluorophore-Doped Polystyrene Microspheres: Seed Material for Airflow Sensing,” *ACS Applied Materials & Interfaces*, 7(37), 20714-20725.
- <sup>36</sup> Petrosky BJ\*, Lowe KT, Danehy PM, Wohl CJ and Tiemsin PI 2015 “Improvements in laser flare removal for particle image velocimetry using fluorescent dye-doped particles,” *Measurement Science and Technology*, 26(11), 115303
- <sup>37</sup> Overbruggen, T., Schroder, F., Klass, M., Schroder, W., “A particle-image velocimetry tracer generating technique for liquid flows,” *Measurement Science and Technology* (2014) 25.
- <sup>38</sup> Kähler, C.J., Sammler, B., Kompenhans J. “Generation and control of tracer particles for optical flow investigations in air”, *Experiments in Fluids*, (2002) 33, 736-742
- <sup>39</sup> Biwole, P.J., Yan, W., Zhang, Y.H., Roux, J.J., “A complete 3D particle tracking algorithm and its applications to indoor airflow study,” *Measurement Science and Technology* (2009) 20: 11.
- <sup>40</sup> Scarano, F., Ghaemi, S., Caridi, G.C.A., Bosbach, J., Dierksheide, U., Sciacchitano, A., “On the use of helium-filled soap bubbles for large-scale tomographic PIV in wind tunnel experiments,” *Experiments in Fluids* (2015) 56: 42.
- <sup>41</sup> Schneiders, J.F.G., Caridi, G.C.A., Sciacchitano, A., Scarano, F., “Large-scale volumetric pressure from tomographic PTV with HFSB tracers,” *Experiments in Fluids* (2016) 57: 164.
- <sup>42</sup> Tedeschi, G., Gouin, H., Elena, M., “Motion of tracer particles in supersonic flows,” *Experiments in Fluids* (1999) 26: 288-296.

- 
- <sup>43</sup> E. Cunningham, "On the velocity of steady fall of spherical particles through fluid medium," *Proc. R. Soc. A* 83, 357–365 (1910).
- <sup>44</sup> Loth, E., "Compressibility and Rarefaction Effects on Drag of a Spherical Particle," *AIAA Journal* (2008) 46: 9.
- <sup>45</sup> Williams, O.J.H., Nguyen, T., Schreyer, A.M., Smits, A.J., "Particle response analysis for particle image velocimetry in supersonic flows," *Physics of Fluids* (2015) 27.
- <sup>46</sup> Ghaemi, S., Schmidt-Ott, A., and Scarano, F., "Nanostructured tracers for laser-based diagnostics in high-speed flows," *Measurement Science and Technology* (2010) 21: 105403.
- <sup>47</sup> Raffel, M., Willert, C., Kompenhans, J., *Particle Image Velocimetry: A Practical Guide*, Springer-Verlag Berlin Heidelberg, Germany, 1998.
- <sup>48</sup> Westerweel, J., Elsinga, G.E., Adrian, R.J., "Particle Image Velocimetry for Complex and Turbulent Flows," *Annual Review of Fluid Mechanics* (2013) 45: 409.
- <sup>49</sup> Lindken, R., Merzkirch, W., "A novel PIV technique for measurements in multiphase flows and its application to two-phase bubbly flows," *Experiments in Fluids* (2002) 33: 814-825.
- <sup>50</sup> Bröder, D., Sommerfeld, M., "Planar shadow image velocimetry for analysis of the hydrodynamics in bubbly flows," *Measurements Science and Technology* (2007) 18: 2513-2528.
- <sup>51</sup> Willert, C., Stasicki, B., Klinner, J., Moessner, S., "Pulsed operation of high-power light emitting diodes for imaging flow velocimetry," *Measurement Science and Technology* (2010) 21.
- <sup>52</sup> Buchmann, N.A., Willert, C.E., Soria, J., "Pulsed, high-power LED illumination for tomographic particle image velocimetry," *Experiments in Fluids* (2012) 53: 1545-1560.
- <sup>53</sup> Maheo, P. 1998 "Free-Surface Turbulent Shear Flows", Ph. D. Thesis, CalTech.
- <sup>54</sup> Novara, M., Scarano, F., "Lagrangian acceleration evaluation for tomographic PIV: a particle-tracking based approach," *16<sup>th</sup> Int Symp on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, 09-12 July, 2012.
- <sup>55</sup> Novara, M., Scarano, F., "A particle-tracking approach for accurate material derivative measurements with tomographic PIV," *Experiments in Fluids* (2013) 54: 1584.
- <sup>56</sup> Troolin, D.R., Longmire, E.K., "Volumetric velocity measurements of vortex rings from inclined exits," *Experiments in Fluids* (2010) 48: 409-420,
- <sup>57</sup> Elsinga, G.E., *Tomographic particle image velocimetry and its application to turbulent boundary layers* dissertation (2008).
- <sup>58</sup> Willert, C.E., Gharib, M., "Digital Particle Image Velocimetry," *Experiments in Fluids* (1991) 10: 181-193.
- <sup>59</sup> Hain, R., Kähler, C.J., Tropea, C., "Comparison of CCD, CMOS, and intensified cameras," *Experiments in Fluids* (2007) 42: 403-411.
- <sup>60</sup> G. Dezsó-Weidinger, A. Stitou, J. van Beeck, M.L. Riethmuller, Measurement of the turbulent mass flux with PTV in a street canyon, *J. Wind Eng. Ind. Aerodyn.* 91 (2003) 1117–1131
- <sup>61</sup> K. Ohmi, H.-Y. Li, Particle-tracking velocimetry with new algorithms, *Meas. Sci. Technol.* 11 (2000) 603–616.
- <sup>62</sup> Lei, Y.C., Tien, W.H., Duncan, J., Paul, M., Ponchaut, N., Mouton, C., Dabiri, D., Rösgen, T., Hove, J., "A vision-based hybrid particle tracking velocimetry (PTV) technique using a modified cascade correlation peak-finding method," *Experiments in Fluids* (2012).
- <sup>63</sup> Alexander, B.F., Ng, K.C., "Elimination of systematic error in subpixel accuracy centroid estimation," *Optical Engineering* (1991) 30: 1320-1331.
- <sup>64</sup> Cowen, E.A., Monosmith, S.G., "A hybrid digital particle tracking velocimetry technique," *Experiments in Fluids* (1997) 22:199-211.

- 
- <sup>65</sup> Marxen, M., Sullivan, P.E., Loewen, M.R., Jahne, B., "Comparison of Gaussian particle center estimators and the achievable measurement density for particle tracking velocimetry," *Experiments in Fluids* (2000) 29:145-153.
- <sup>66</sup> Nobach, H., Honkanen, M., "Two-dimensional Gaussian regression for sub-pixel displacement estimation in particle image velocimetry or particle position estimation in particle tracking velocimetry," *Experiments in Fluids* (2005) 38: 511-515.
- <sup>67</sup> Mikheev, A.V., Zubtsov, V.M., "Enhanced particle-tracking velocimetry (EPTV) with a combined two-component pair-matching algorithm," *Measurement Science and Technology* (2008) 19.
- <sup>68</sup> Carosone, F., Cnedese, A., Querzoli, G., "Recognition of partially overlapped particle images using the Kohonen neural network," *Experiments in Fluids* (1995) 19: 225-232.
- <sup>69</sup> Takehara, K., Etoh, T., "A Study on Particle Identification in PTV – Particle Mask Correlation Method –" *Journal of Visualization* (1999) 1: 313-323.
- <sup>70</sup> Angarita-Jaimes NC, Roca MG, Towers CE, Read ND, Towers DP (2009) Algorithms for the automated analysis of cellular dynamics within living fungal colonies. *Cytom A* 75(9): 768–780.
- <sup>71</sup> Shindler, L., Moroni, M., Cenedese, A., "Spatial-temporal improvements of a two-frame particle-tracking algorithm," *Measurement Science and Technology* (2010) 21.
- <sup>72</sup> L. Shindler, M. Moroni, and A. Cenedese, "Using optical flow equation for particle detection and velocity prediction in particle tracking," *Applied Mathematics and Computation*, vol. 218, no. 17, pp. 8684–8694, 2012.
- <sup>73</sup> Lucas, B. and Kanade, T. 1981. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674- 679.
- <sup>74</sup> Tomasi, C., and T. Kanade Shape and Motion from Image Streams: a Factorization Method- Part 3 Detection and Tracking of Point Features. Technical Report CMU-CS-91-132, School of Computer Science, Carnegie Mellon University, April, 1991.
- <sup>75</sup> Okamoto K., Nishio S., Kobayashi T. and Saga T.: Standard Images for Particle Imaging Velocimetry, Proc. of the 2nd International Workshop on PIV '97- Fukui (1997), 229-236.
- <sup>76</sup> Cardwell ND, Vlachos PP, Thole KA (2011) A multi-parametric particle-pairing algorithm for particle tracking in single and multiphase flows. *Meas Sci Technol* 22(10):105406
- <sup>77</sup> Prasad, A.K., "Stereoscopic particle image velocimetry," *Experiments in Fluids* (2000) 29: 103-116.
- <sup>78</sup> Graff, E.C., Gharib, M., "Performance prediction of point-based three-dimensional volumetric measurement systems," *Measurement Science and Technology* (2008) 19.
- <sup>79</sup> Watanabe, R., Gono, T., Yamagata, T., Fujisawa, N., "Three-dimensional flow structure in a highly buoyant jet by scanning stereo PIV combined with POD analysis," *International Journal of Heat and Fluid Flow* (2015) 52: 98-110.
- <sup>80</sup> Doh, D.H., Hwang, T.G., Saga, T., "3D-PTV measurements of the wake of a sphere," *Measurement Science and Technology* (2004) 15: 1059-1066.
- <sup>81</sup> Kreizer, M., Liberzon, A., "Three-dimensional particle tracking method using FPGA-based real-time image processing and four-view image splitter," *Experiments in Fluids* (2011) 50: 613-620.
- <sup>82</sup> Murai S., Nakamura, H., Suzuki Y., "Analytical orientation for non-metric camera in the application of terrestrial photogrammetry", *Int'l Arch. Photogrammetry XXIII*, Commision V, pp. 516-525.
- <sup>83</sup> Nishina, K., Kasagi N., Hirata M. "Three-Dimensional Particle Tracking Velocimetry Based on Automated Digital Image Processing," *Transactions of the ASME* (1989) 111: 384-391
- <sup>84</sup> Brown, D., "Close-range camera calibration:," *Photogrammetric Eng.* (1971), 37(8)
- <sup>85</sup> El-Hakrm. S. F. "Real-Time Image Metrology with CCD Cameras." *Photogrammetric Engineering and Remote Sensing*, (1986) 52:1757-1766
- <sup>86</sup> Albertz, J.; Kreiling, W. 1989:P hotogrammetrischeTsa schenbuch. Wichmann Verlag, Karlsruhe
- <sup>87</sup> Fuchs, T., Hain, R., Kähler, C.J., "Double-frame 3D-PTV using a tomographic predictor," *Experiments in Fluids* (2016) 57: 174.

- 
- <sup>88</sup> Fuchs, T., Hain, R., Kähler, C.J., “Uncertainty quantification of three-dimensional velocimetry techniques for small measurement depths,” *Experiments in Fluids* (2016) 57: 73.
- <sup>89</sup> Wieneke, B., “Iterative reconstruction of volumetric particle distribution,” *Measurement Science and Technology* (2013) 24.
- <sup>90</sup> Bown, M.R., MacInnes, J.M., Allen, R.W.K., and Zimmerman, W.B.J., “Three-dimensional, three-component velocity measurements using stereoscopic micro-PIV and PTV,” *Measurement Science and Technology* (2006) 17: 2175-2185.
- <sup>91</sup> Yu, C.H., Yoon, J.H., and Kim, H.B., “Development and validation of stereoscopic micro-PTV using match probability,” *Journal of Mechanical Science and Technology* (2009) 23: 845-855.
- <sup>92</sup> Peterson, K., Regaard, B., Heinemann, S., and Sick, V., “Single-camera, three-dimensional particle tracking velocimetry,” *Optics Express* (2012) Vol. 20 No. 8: 9031-9037,
- <sup>93</sup> Bao, X., Li, M., "Defocus and binocular vision based stereo pairing method for 3D particle tracking velocimetry," *Optics and Lasers in Engineering* (2011) 49: 623-631.
- <sup>94</sup> Panday, S.P., Ohmi, K., Nose, K., "An ant colony optimization based stereoscopic particle pairing algorithm for three-dimensional particle tracking velocimetry," *Flow Measurement and Instrumentation* (2011) 22: 86-95.
- <sup>95</sup> Hoyer, K., Holzner, M., Lüthi, B., Guala, M., Liberzon, A., Kinzelbach, W., "3D scanning particle tracking velocimetry," *Experiments in Fluids* (2005) 39: 923-934.
- <sup>96</sup> Willert, C.E., Gharib, M., "3-Dimensional particle imaging with a single camera," *Experiments in Fluids* (1992) 12: 353-358.
- <sup>97</sup> Pereira, F., Gharib, M., Dabiri, D., Modarress., "Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. Application to bubbly flows," *Experiments in Fluids* (2000) 29: S78-S84.
- <sup>98</sup> Yoon S.Y., Kim K.C. “3D particle position and 3D velocity field measurement in a microvolume via the defocusing concept” *Meas. Sci. Technol.* (2006), 17, 2897-2905
- <sup>99</sup> Pereira F., Lu J., Castano-Graff, E., Gharib M. “Microscale 3D flow mapping with  $\mu$ DDPIV”, *Exp. Fluids*, (2007), 42:589-599
- <sup>100</sup> Tien W.H.,Kartes P.,Yamasaki T.,Dabiri D., “A Color-Coded Backlighting Defocusing Particle Image Velocimetry System”, *Experiments in Fluids*, (2008) 44, 1015-1026
- <sup>101</sup> Tien, W.H., Dabiri, D., Hove, J.R., “Color-coded three-dimensional micro particle tracking velocimetry and application to micro backward-facing step flows,” *Experiments in Fluids* (2014) 55: 1684.
- <sup>102</sup> Tien W-H “Development of multi-spectral three-dimensional micro particle tracking velocimetry”, *Meas. Sci. Technol.*, (2016) 084010
- <sup>103</sup> Kajitani, L. and Dabiri, D., “A full three-dimensional characterization of defocusing digital particle image velocimetry,” *Measurement Science and Technology* (2005) 16: 790-804.
- <sup>104</sup> Grothe, R.L., Dabiri, D., “An improved three-dimensional characterization of defocusing digital particle image velocimetry (DDPIV) based on a new imaging volume definition,” *Measurement Science and Technology* (2008) 19.
- <sup>105</sup> Pereira, F., Gharib, M., “Defocusing digital particle image velocimetry and the three-dimensional characterization of two-phase flows,” *Measurement Science and Technology* (2002) 13: 683-694.
- <sup>106</sup> Kao HP, Verkman AS (1994) Tracking of Single Fluorescent Particles in Three Dimensions Cylindrical Optics to Encode Particle Position. *Biophys J* 67:1291-1300.
- <sup>107</sup> Towers, C.E., Towers, D.P., Campbell, H.I., Zhang, S., Greenaway, A.H. “Three-dimensional particle imaging by wavefront sensing”, *Optics Letters*, 31(9).

- 
- <sup>108</sup> Angarita-Jaimes, N., McGhee, E., Chennaoui, M., Campbell, H.I., Zhang, S., Towers, C.E., Greenaway, A.H., Towers, D.P. "Wavefront sensing for single view three-component three-dimensional flow velocimetry", *Exp. Fluids*, (2006) 41:881-891
- <sup>109</sup> Chen, S., Angarita-Jaimes, N., Angarita-Jaimes, D., Pelc, B., Greenaway, A.H., Towers, C.E., Lin, D., Towers, D.P., "Wavefront sensing for three-component three-dimensional flow velocimetry in microfluidics," *Exp. Fluids*, (2009) 47:849-863
- <sup>110</sup> Cierpka, C., Segura, R., Hain, R., Kähler, C.J., "A simple single camera 3D3C velocity measurement technique without errors due to depth of correlation and spatial averaging for microfluidics," *Measurement Science and Technology* (2010) 21.
- <sup>111</sup> Cierpka, C., Rossi M., Segura R., Kähler C.J., "On the calibration of astigmatism particle tracking velocimetry for microflows," *Meas. Sci. Technol.* (2011) 22:015401
- <sup>112</sup> Rossi, M., Kähler, C.J. "Optimization of astigmatic particle tracing velocimeters", *Exp. Fluids*, (2014), 55:1809
- <sup>113</sup> Fuchs, T., Hain, R., Kähler, C.J. "Three-dimensional location of micrometer-sized particles in macroscopic domains using astigmatic aberrations", *Optics Letters*, (2014), 39(5)
- <sup>114</sup> Fuchs, T., Heinold, M., Hain, R., Kähler, C.J. "Stereoscopic Astigmatism Particle Tracking Velocimetry for macroscopic 3D3C flow measurements in air," 17th International Symposium on Applications of Laser Techniques to Fluid Mechanics, Lisbon, Portugal, (2014).
- <sup>115</sup> Fuchs, T., Hain, R., Kähler, C.J. "Macroscopic three-dimensional particle location using stereoscopic imaging and astigmatic aberrations" *Optics Letters*, (2014), 39(24)
- <sup>116</sup> Cagnet M., Francon M., Thierr J.C. Atlas of optical phenomena, (1962) Springer, Berlin Heidelberg, New York
- <sup>117</sup> Gibson S.F. and Lanni F. "Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy", *J. Opt. Soc. Am. A*, (1991) 8, 1601-13
- <sup>118</sup> Park J.S. Study of microfluidic measurement techniques using novel optical imaging diagnostics. PhD Dissertation, Texas A&M University.
- <sup>119</sup> Speidel M., Jonás A., and Florin E.-L., "Three-dimensional tracking of fluorescent nanoparticles with subnanometer precision by use of off-focus imaging," *Opt. Lett.* (2003), 28(2), 69–71.
- <sup>120</sup> Wu, M., J. W. Roberts, and M. Buckley. "Three-dimensional fluorescent particle tracking at micron-scale using a single camera". *Exp. Fluids.*, (2005) 38:461–465.
- <sup>121</sup> Park J.S. & Kihm K.D. "Three-dimensional micro-PTV using deconvolution microscopy," *Exp. Fluids*, (2006) 40: 491-499
- <sup>122</sup> Luo R., Yang X.Y., Peng X.F., Sun Y.F. "Three-dimensional tracking of fluorescent particles applied to micro-fluidic measurements," *J. Micromech. Microeng.*, (2006), 16,1689-1699
- <sup>123</sup> Haeberlé O. "Focusing of light through a stratified medium: a practical approach for computing microscope point spread functions: Part I. Conventional microscopy," (2003) *Opt. Commun.* 216, 55-63
- <sup>124</sup> Luo R, Sun Y F, Peng X F and Yang X Y "Tracking sub-micron fluorescent particles in three dimensions with a microscope objective under non-design optical conditions," *Meas. Sci. Technol.* (2006) 17 1358–66
- <sup>125</sup> Luo R, Sun Y F "Pattern matching for three-dimensional tracking of sub-micron fluorescent particles," *Meas. Sci. Technol.* (2011) 22 045402
- <sup>126</sup> Peterson S.D., Chuang H-S, Wereley S.T., "Three-dimensional particle tracking using micro-particle image velocimetry hardware", (2008), 19, 115406
- <sup>127</sup> Doh, D.H., Cho, G.R., Kim, Y.H., "Development of a tomographic PTV," *Journal of Mechanical Science and Technology* (2012) 26: 3811-3819.
- <sup>128</sup> Elsinga, G.E., Scarano, F., Wieneke, B., van Oudheusden, B.W., "Tomographic particle image velocimetry," *Experiments in Fluids* (2006) 41: 993-947.

- 
- <sup>129</sup> Atkinson, C., Soria, J., “An efficient simultaneous reconstruction technique for tomographic particle image velocimetry,” *Experiments in Fluids* (2009) 47: 553-568.
- <sup>130</sup> Kitzhofer, J., Brücker, C., “Tomographic particle tracking velocimetry using telecentric imaging,” *Experiments in Fluids* (2010) 49:1307-1324.
- <sup>131</sup> Belden J, Truscott TT, Axiak MC, Techet AH (2010) Three-dimensional synthetic aperture particle image velocimetry. *Meas Sci Technol* 21(12):125403
- <sup>132</sup> Bajpayee A, Techet AH (2013) 3D particle tracking velocimetry (PTV) using high speed light field imaging. In: PIV13; 10th International Symposium on Particle Image Velocimetry, Delft, The Netherlands, July 2013
- <sup>133</sup> Vaish V, Garg G, Talvala E-V, Antunez E, Wilburn B, Horowitz M, Levoy M (2005) Synthetic aperture focusing using a shear-warp factorization of the viewing transform. In: IEEE computer society Conference on computer vision and pattern recognition- workshops, 2005. CVPR Workshops, pp 129–129
- <sup>134</sup> Belden J (2013) Calibration of multi-camera systems with refractive interfaces. *Exp Fluids* 54(2):1–18
- <sup>135</sup> Bajpayee, A. and Techet, A.H., “Fast volume reconstruction for 3D PTV,” *Experiments in Fluids* (2017) 58:95.
- <sup>136</sup> E. H. Adelson and J. Y. A. Wang, “Single lens stereo with a plenoptic camera,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 99–106, Feb. 1992. [Online]. Available: <http://dx.doi.org/10.1109/34.121783>
- <sup>137</sup> R.Ng, “Digital Light Field Photography,” Ph.D.dissertation, Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Jul. 2006.
- <sup>138</sup> R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, “Light field photography with a hand-held plenoptic camera,” Stanford Univ., Stanford, CA, USA, Tech. Rep. CSTR 2005-02, Apr. 2005. [Online]. Available: <http://graphics.stanford.edu/papers/lfcamera/>
- <sup>139</sup> A. Lumsdaine and T. Georgiev, “The focused plenoptic camera,” in *Proc. 2009 IEEE Int. Conf. Comput. Photogr.*, Apr. 2009, pp. 1–8.
- <sup>140</sup> T. Fahringer, K. Lynch, and B. Thurow, “Volumetric particle image velocimetry with a single plenoptic camera,” *Meas. Sci. Tech.*, vol. 26, 2015, Art. no. 115201.
- <sup>141</sup> T. Fahringer and B. Thurow, “The effect of grid resolution on the accuracy of tomographic reconstruction using a plenoptic camera,” in *Proc. 51st AIAA Aerosp. Sci. Meeting Including New Horizons Forum Aerosp. Expo.*, 2013. [Online]. Available: <http://dx.doi.org/10.2514/6.2013-39>
- <sup>142</sup> T. Fahringer and B. Thurow, “Filtered refocusing: A volumetric reconstruction algorithm for plenoptic-PIV,” *Meas. Sci. Tech.*, vol. 27, 2016, Art. no. 094005.
- <sup>143</sup> Anglin, P., Reeves, S.J., and Thurow, B.S., “Characterization of Plenoptic Imaging Systems and Efficient Volumetric Estimation From Plenoptic Data,” *IEEE Journal of Selected Topics in Signal Processing* (2017) 11: 1020-1033.
- <sup>144</sup> Chen, H., Sick, V., “Three-Dimensional Three-Component Air Flow Visualization in a Steady-State Engine Flow Bench Using a Plenoptic Camera,” *SAE International Journal of Engines* (2017) 10: 625-635.
- <sup>145</sup> Katz, J. and Sheng, J., “Applications of Holography in Fluid Mechanics and Particle Dynamics,” *Annual Review of fluid Mechanics* (2010) 42: 531-555.
- <sup>146</sup> Yu, X., Hong, J., Liu, C., and Kim M.K., “Review of digital holographic microscopy for three-dimensional profiling and tracking,” *Optical Engineering* (2014) 53: 112306.

- 
- <sup>147</sup> Memmolo, P., Miccio, L., Paturzo, M., Di Caprio, G., Coppola, G., Netti, P.A., and Ferraro, P., “Recent advances in holographic 3D particle tracking,” *Advances in Optics and Photonics* (2015) 7: 713.
- <sup>148</sup> Toloui, M., Mallery, K., and Hong, J., “Improvements on digital inline holographic PTV for 3D wall-bounded turbulent flow measurements,” *Measurement Science and Technology* (2017) 28: 044009.
- <sup>149</sup> Toloui, M. and Hong, J., “High fidelity digital inline holographic method for 3D flow measurements,” *Optics Express* (2015) 23: 10.1364.
- <sup>150</sup> Choi, Y.S., Seo, K.W., Sohn, M.H., Lee, S.J., “Advances in digital holographic micro-PTV for analyzing microscale flows,” *Optics and Lasers in Engineering* (2012) 50: 39-45.
- <sup>151</sup> Buchmann, N.A., Atkinson, C., and Soria, J., “Ultra-high-speed tomographic digital holographic velocimetry in supersonic particle-laden jet flows,” *Measurement Science and Technology* (2013) 24: 024005.
- <sup>152</sup> Lu, J., Fuga, J.P., Nordsiek, H., Saw, E.W., Shaw, R.A., and Yang, W., “Lagrangian particle tracking in three dimensions via single-camera in-line digital holography,” *New Journal of Physics* (2008) 10: 125013.
- <sup>153</sup> Satake, S., Anraku, T., Kanamori, H., Kunugi, T., Sato, K., and Ito, T., “Measurements of Three-Dimensional Flow in Microchannel With Complex Shape by Micro-Digital-Holographic Particle-Tracking Velocimetry,” *Journal of Heat Transfer* (2008) 130: 042412-1.
- <sup>154</sup> Schablinski, J., Kroll, M., and Block, D., “Particle Tracking Velocimetry of Dusty Plasmas Using Stereoscopic In-Line Holography,” *IEEE Transactions on Plasma Science* (2013) 41: 779.
- <sup>155</sup> P. J. Shaw, “Comparison of wide-field/deconvolution and confocal microscopy for 3D imaging,” in *Handbook of Biological Confocal Microscopy*, J. B. Pawley, Ed., pp. 373–387, Plenum Press, New York and London (1995).
- <sup>156</sup> J. G. McNally et al., “Three-dimensional imaging by deconvolution microscopy,” *Methods* 19(3), 373–385 (1999).
- <sup>157</sup> P. Sarder and A. Nehorai, “Deconvolution methods for 3-D fluorescence microscopy images,” *IEEE Signal Process. Mag.* 23(3), 32–45 (2006).
- <sup>158</sup> W. Wallace, L. H. Schaefer, and J. R. Swedlow, “A working person’s guide to deconvolution in light microscopy,” *Biotechniques* 31(5), 1076–1097 (2001).
- <sup>159</sup> T. Latychevskaia, F. Gehri, and H. Fink, “Depth-resolved holographic reconstructions by three-dimensional deconvolution,” *Opt. Express* 18(21), 22527–22544 (2010).
- <sup>160</sup> Latychevskaia, T., and Fink, H.W., “Holographic time-resolved particle tracking by means of three-dimensional volumetric deconvolution,” *Optics Express* (2014) 22: 10.1364.
- <sup>161</sup> Y. S. Choi and S. J. Lee, “Three-dimensional volumetric measurement of red blood cell motion using digital holographic microscopy,” *Appl. Opt.* 48(16), 2983–2990 (2009).
- <sup>162</sup> S. J. Lee, K. W. Seo, Y. S. Choi, and M. H. Sohn, “Three-dimensional motion measurements of free-swimming microorganisms using digital holographic microscopy,” *Meas. Sci. Technol.* 22(6), 064004 (2011).
- <sup>163</sup> Seo, K.W., and Lee, S.J., “High-accuracy measurement of depth-displacement using a focus function and its cross-correlation in holographic PTV,” *Optics Express* (2014) 22: 10.1364.
- <sup>164</sup> Kim, S., and Lee, S.J., “Effect of particle number density in in-line digital holographic particle velocimetry,” *Experiments in Fluids* (2008) 44: 623-631.
- <sup>165</sup> Toloui, M., Mallery, K., and Hong, J., “Improvements on digital inline holographic PTV for 3D wall-bounded turbulent flow measurements,” *Measurement Science and Technology* (2017) 28: 044009.

- 
- <sup>166</sup> Soloff, S.M., Adrian, R.J., Liu, Z.C., "Distortion compensation for generalized stereoscopic particle image velocimetry," *Measurement Science and Technology* (1997) 8: 1441-1454.
- <sup>167</sup> Kent J C, Trigui N, Choi W-C, Guezennec Y G and Brodkey R S (1993) "Photogrammetric calibration for improved three-dimensional particle tracking velocimetry" *Optical Diagnostics in Fluid and Thermal Flow* ed S Cha and J Trolinger (Bellingham, WA: SPIE) pp 400–12.
- <sup>168</sup> Prasad A K, Adrian R J, Landreth C C and Offutt P W (1992) "Effect of resolution on the speed and accuracy of particle image velocimetry interrogation," *Exp. Fluids* 13: 105–116.
- <sup>169</sup> Wieneke, B., "Stereo-PIV using self-calibration on particle images," *Experiments in Fluids* (2005) 39: 267-280.
- <sup>170</sup> Choi, W. and Guezennec, Y., "In situ calibration for wide-angle, three-dimensional stereoscopic image analysis," *Applied Optics* (1994) 36: 7364-7373.
- <sup>171</sup> Schosser, C., Fuchs, T., Hain, R., and Kahler, C.J., "Non-intrusive calibration for three-dimensional particle imaging," *Experiments in Fluids* (2016) 57: 69.
- <sup>172</sup> Maas HG (1995) New developments in multimedia photogrammetry. In: Gru n A, Kahmen H (eds) *Optical 3-D measurement techniques III*. Wichmann Verlag, Karlsruhe
- <sup>173</sup> Mulsow C (2010) A flexible multi-media bundle approach. In: *International archives of photogrammetry, remote sensing and spatial information sciences*, vol XXXVIII, part 5. Commission V symposium, Newcastle upon Tyne, UK
- <sup>174</sup> Kent, JC, and Trigui, N., "Photogrammetric Calibration for Improved Three Dimensional Particle Tracking Velocimetry (3DPTV)" *SPIE* (1993) vol. 400, 0-8194-1254-6.
- <sup>175</sup> Tien, W.H., Dabiri, D., Hove, J.R., "Color-coded three-dimensional micro particle tracking velocimetry and application to micro backward-facing step flows," *Experiments in Fluids* (2014) 55: 1684.
- <sup>176</sup> Yoon, S.Y., Kihm, K.D., and Kim, K.C., "Correlation of fluid refractive index with calibration coefficient for micro-defocusing digital particle image velocimetry," *Measurement Science and Technology* (2011) 22 doi:10.1088/0957-0233/22/3/037001.
- <sup>177</sup> Wieneke, B., "Volume self-calibration for 3D particle image velocimetry," *Experiments in Fluids* (2008) 45: 549-556.
- <sup>178</sup> Discetti, S. and Astarita, T., "The detrimental effect of increasing the number of cameras on self-calibration for tomographic PIV," *Measurement Science and Technology* (2014) 25.
- <sup>179</sup> Schanz, D., Gesemann, S., Schroder, A., Wieneke, B., Novara, M., "Non-uniform optical transfer functions in particle imaging: calibration and application to tomographic reconstruction," *Measurement Science and Technology* (2013) 24.
- <sup>180</sup> Hassan, Y., Cannan, R., "Full-field bubbly flow velocity-measurements using a multiframe particle tracking technique," *Experiments in Fluids* (1991) 12: 49-60.
- <sup>181</sup> Oullette, N., Xu, H., Bodenschatz, E., "A quantitative study of three-dimensional Lagrangian particle tracking algorithms," *Experiments in Fluids* (2006) 40:301-313.
- <sup>182</sup> Malik, N.A., Dracos, T., Papantoniou, D.A., "Particle tracking velocimetry in three-dimensional flows – part 2. Particle tracking," *Experiments in Fluids* (1993) 15: 279-294.
- <sup>183</sup> Tarlet, D., Bendicks, C., Roloff, C., Bordas, R., Wunderlich, B., Michaelis, B., Thevenin, D., "Gas Flow Measurements by 3D Particle Tracking Velocimetry Using Coloured Tracer Particles," *Flow Turbulence and Combustion* (2012) 88: 343-365.
- <sup>184</sup> Yamamoto, F., Uemura, T., Tian, H.Z., Ohmi, K., "Three-Dimensional PTV Based on Binary Cross-Correlation Method," *JSME International Journal* (1993) 36: 279.
- <sup>185</sup> Hassan, Y.A., Blanchat, T.K., Seeley, C.H., "PIV Flow Visualization Using Particle Tracking Techniques," *Measurement Science and Technology* (1992) 3: 633-642.
- <sup>186</sup> Saga, T., Segawa, S., Kobayashi, T., Hu, H., "Development and evaluation of an improved correlation based PTV method," *Institute of Industrial Science, University of Tokyo*.

- 
- <sup>187</sup> Rosenfeld, A., Hummel, R. A., and Zcker, S.W., "Scene labeling by relaxation operations," *IEEE Trans. SMC* Vol. 6 No. 6 (1976) pp 420-433.
- <sup>188</sup> Wu, Q.X., Pairman, D., "A Relaxation Labeling Technique for Computing Sea Surface Velocities from Sea Surface Temperature," *IEEE Transactions on Geoscience and Remote Sensing* (1995) 33: 216.
- <sup>189</sup> Baek, S.J., Lee, S.J., "A new two-frame particle tracking algorithm using match probability," *Experiments in Fluids* (1996) 22: 23-32.
- <sup>190</sup> Pereira, F., Stüer, H., Graff, E.C., Gharib, M., "Two-frame 3D particle tracking," *Measurement Science and Technology* (2006) 17: 1680-1692.
- <sup>191</sup> Brevis, W., Niño, Y., Jirka, G. H., "Integrating cross-correlation and relaxation for particle tracking velocimetry," *Experiments in Fluids*, (2011) 50: 135-147.
- <sup>192</sup> Barnard S T and Thompson W B 1980 Disparity analysis of images *IEEE Trans. Pattern Analysis Machine Intelligence* 2 333–40
- <sup>193</sup> Jia, P., Wang, Y., Zhang, Y., "Improvement in the independence of relaxation method-based particle tracking velocimetry," *Measurement Science and Technology* (2013) 24.
- <sup>194</sup> Jia, P., Wang, Y., Zhang, Y., Yang, B., "Relaxation algorithm-based PTV with dual calculation method and its application in addressing particle saltation," *Journal of Flow Visualization* (2015) 18: 71-81.
- <sup>195</sup> Grant, I., Pan, X., "An investigation of the performance of multi layer neural networks applied to the analysis of PIV images," *Experiments in Fluids* (1995) 189:159-166.
- <sup>196</sup> Labonte, G., "A new neural network for particle-tracking velocimetry," *Experiments in Fluids* (1999) 26: 340-346.
- <sup>197</sup> Ohmi, K., "SOM-Based particle matching algorithm for 3D particle tracking velocimetry," *Applied Mathematics and Computation* (2008) 205: 890-898.
- <sup>198</sup> B. Angéniol, G.C. Vaubois, J.Y. Le Texier, Self-organizing feature maps and the traveling salesman problem, *Neural Networks* 1 (1988) 289–293.
- <sup>199</sup> K. Fujimura, H. Tokutaka, T. Maenou, K. Iseki, E. Kuwabara, M. Ishikawa, Performance of improved SOM-TSP algorithm for traveling salesman problem of many cities (in Japanese), in: *IEICE Technical Report: Neuro-Computing 97–623*, 1998, pp. 95–102.
- <sup>200</sup> Murai Y, Song X Q, Takagi T, Ishikawa M, Yamamoto F and Ohta J 1999 *Trans. JSME* **65** B 1339 (in Japanese) [2000 *JSME Int. J. Fluids Thermal Engng* at press].
- <sup>201</sup> Ishikawa, M., Murai, Y., Wada, A., Iguchi, M., Okamoto, K., Yamamoto, F., "A novel algorithm for particle tracking velocimetry using the velocity gradient tensor," *Experiments in Fluids* (2000) 29: 519-531.
- <sup>202</sup> Ishikawa, M., Murai, Y., Yamamoto, F., "Numerical validation of velocity gradient tensor particle tracking velocimetry for highly deformed flow fields," *Measurement Science and Technology* (2000) 11: 677-684.
- <sup>203</sup> Ruan X and Zhao W 2005 A novel particle tracking algorithm using polar coordinate system similarity *Acta Mech. Sin.* **21** 430–5.
- <sup>204</sup> Stellmacher, M., Obermayer, K., "A new particle tracking algorithm based on deterministic annealing and alternative distance measures," *Experiments in Fluids* (2000) 28: 506-518.
- <sup>205</sup> Krepki, R., Pu, Ye., Meng, H., Obermayer, K., "A new algorithm for the interrogation of 3D holographic PTV data based on deterministic annealing and expectation minimization optimization," *Experiments in Fluids* (2000) S99-S107.
- <sup>206</sup> Runhau, P., Guetter, C., Putze, T., Schnörr, C., "A variational approach for particle tracking velocimetry," *Measurement Science and Technology* (2005) 16: 1449-1458.
- <sup>207</sup> Ohyama, R. and Kaneko, K., Experimental Study on Space and Time Correspondence of Traveling Particles for Three-Dimensional Particle Image Velocimetry by Genetic Algorithm, *Proc. SPIE*, Vol.3172, (1997), 688-699.

- 
- <sup>208</sup> Sheng, J. and Meng, H., A Genetic Algorithm Approach for 3D Velocity Field Extraction in Holographic Particle Image Velocimetry, *Experiments in Fluids*, 25 (1998), 461-473.
- <sup>209</sup> Doh, D. H., Kim, D. H., Cho, K. R., Cho, Y. B., Lee, W. J., Saga, T. and Kobayashi, T., Development of Genetic Algorithm Based 3D-PTV Technique, *Journal of Visualization*, 5-3 (2002), 243-254.
- <sup>210</sup> Kimura, I., Hattori, A. and Ueda, M., Particle Pairing Using Genetic Algorithms for PIV, *Proc. VSJ-SPIE 98*, (1998), AB-093.
- <sup>211</sup> Furukawa, T., Kimura, I., Kuroe, Y. and Kaga, A., Hybrid PTV using Neural Networks and Genetic Algorithms, *Proceedings of the 2nd Pacific Symposium on Flow Visualization and Image Processing*, (1999), PF-075.
- <sup>212</sup> Ohmi, K. and Yoshida, N., A New Type of GA-Based Particle Tracking Velocimetry, *Proceedings of 9th International Symposium on Flow Visualization*, (2000), #401.
- <sup>213</sup> Ohmi, K., Panday, S.P., "Particle Tracking Velocimetry Using the Genetic Algorithm," *Journal of Visualization* (2009) 3: 217-232.
- <sup>214</sup> Takagi T (2007) Study on particle tracking velocimetry using ant colony optimization (in Japanese). *J Visual Soc Japan* 27(S2): 89–90.
- <sup>215</sup> Ohmi, K., Panday, S.P., Sapkota, A., "Particle tracking velocimetry with an ant colony optimization algorithm," *Experiments in Fluids* (2010) 48: 589-605.
- <sup>216</sup> Wernet, M.P., "Fuzzy logic particle tracking velocimetry," *Proceedings fo the SPIE conference on Optical Diagnostics in Fluid and Thermal Flow* (1993) San Diego, CA, 11-16 July.
- <sup>217</sup> Wernet, M.P., "New insights into particle image velocimetry data using fuzzy-logic-based correlation/particle tracking processing," *Experiments in Fluids* (2001) 30: 434-447.
- <sup>218</sup> Song, X., Yamamoto, F., Iguchi, M., Murai, Y., "A new tracking algorithm of PIV and removal of spurious vectors using Delaunay tessellation," *Experiments in Fluids* (1999) 26: 371-380.
- <sup>219</sup> Harrington S (1987) *Computer graphics — a programming approach*, pp. 190—195, New York: McGraw-Hill Int.
- <sup>220</sup> Zhang, Y., Wang, Y., and Jia, P., "Improving the Delaunay tessellation particle tracking algorithm in the three-dimensional field," *Measurement Science and Technology* (2014) 49: 1-14.
- <sup>221</sup> Zhang, Y., Wang, Y., Yang, B., He, W., "A particle tracking velocimetry algorithm based on the Voronoi diagram," *Measurement Science and Technology* (2015) 26.
- <sup>222</sup> Lei Y-C, Tien W-H, Duncan J, Paul M, Ponchaut N, Mouton C, Dabiri D, Roßgen T, Hove J (2012) A vision-based hybrid particle tracking velocimetry (PTV) technique using a modified cascade-correlation peak-finding method. *Exp Fluids* 53(5): 1251–1268.
- <sup>223</sup> Paul, M., Tien, W.H., Dabiri, D., "A displacement-shifted vision-based hybrid particle tracking velocimetry (PTV) technique," *Experiments in Fluids* (2014) 55: 1676.
- <sup>224</sup> Scott G, Longuet-Higgins H (1991) An algorithm for associating the features of two images. *Proc R Soc Lond B Biol Sci* 244(1309): 21–26.
- <sup>225</sup> Pilu M (1997) A direct method for stereo correspondence based on singular value decomposition. In: *Computer vision and pattern recognition, 1997. Proceedings, 1997, IEEE Computer Society Conference on. IEEE*, pp 261-266.
- <sup>226</sup> Pun, C.S., Susanto, A., and Dabiri, D., "Mode-ratio bootstrapping method for PIV outlier correction," *Measurement Science and Technology* (2007) 18: 3511-3522.
- <sup>227</sup> Sapkota, A. and Ohmi, K., "Error detection and performance analysis scheme for particle tracking velocimetry results using fuzzy logic," *International Journal of Innovative Computing, Information, and Control* (2009) 5: 4927-4934.
- <sup>228</sup> Westerweel, J. and Scarano, F., "Universal outlier detection for PIV data," *Experiments in Fluids* (2005) 39: 170-91.

- 
- <sup>229</sup> Duncan, J., Dabiri, D., Hove, J., Gharig, M., “Universal Outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data,” *Measurement Science and Technology* (2010) 21.
- <sup>230</sup> Masullo, A., Theunissen, R., “Adaptive vector validation in image velocimetry to minimize the influence of outlier clusters,” *Experiments in Fluids* (2016) 57: 33.
- <sup>231</sup> Guezennec, Y.G., Brodkey, R.S., Trigui, N., Kent, J.C., “Algorithms for fully automated three-dimensional particle tracking velocimetry,” *Experiments in Fluids* (1994) 17: 209-219.
- <sup>232</sup> Schanz, D., Schröder, A., Gesemann, S., Michaelis, D., Wieneke, B., "Shake The Box: A highly efficient and accurate Tomographic Particle Tracking Velocimetry (TOMO-PTV) method using prediction of particle positions," International Symposium on Particle Image Velocimetry (2013).
- <sup>233</sup> Schanz, D., Schröder, A., Gesemann, S., "Shake the Box - a 4D PTV algorithm: Accurate and ghostless reconstruction of Lagrangian tracks in densely seeded flows," *International Symposium on Applications of Laser Techniques to Fluid Mechanics* (2014).
- <sup>234</sup> Schanz, D., Gesemann, S., Schroder, A., “Shake-The-Box: Lagrangian particle tracking at high particle image densities,” *Experiments in fluids* (2016) 57: 70.
- <sup>235</sup> Cohn R.K., Koochesfahani, M.M. “The accuracy of remapping irregularly spaced velocity data onto a regular grid and the computation of vorticity,” *Experiments in Fluids* (2000) S61-S69.
- <sup>236</sup> Agui, J.C., Jimenez, J., “On the performance of particle tracking,” *Journal of Fluid Mechanics* (1987) 185: 447-468.
- <sup>237</sup> Malik, N.A. and Dracos, T. “Interpolation Schemes for Three-Dimensional Velocity Fields from Scattered Data using Taylor Expansions,” *Journal of Computational Physics* (1995) 119: 231-243.
- <sup>238</sup> Ido, T., Murai, Y., and Yamamoto, F., “Postprocessing algorithm for particle-tracking velocimetry based on ellipsoidal equations,” *Experiments in Fluids* (2002) 32: 326-336.
- <sup>239</sup> Ido, T. and Murai, Y., “A recursive interpolation algorithm for particle tracking velocimetry,” *Flow Measurement and Instrumentation* (2006) 17: 267-275.
- <sup>240</sup> Vedula, P. and Adrian, R.J., “Optimal solenoidal interpolation of turbulent vector fields: application to PTV and super-resolution PIV,” *Experiments in Fluids* (2005) 39: 213-221.
- <sup>241</sup> Casa, L.D.C. and Krueger, P.S., “Radial basis function interpolation of unstructured, three-dimensional, volumetric particle tracking velocimetry data,” *Measurements Science and Technology* (2013) 24 065304.
- <sup>242</sup> Gesemann, S., “From particle tracks to velocity and acceleration fields using B-splines and penalties,” (2015) arXiv 1510.09034.
- <sup>243</sup> Schneiders J.F.G., Dwight, R.P., Scarano, F., “Time-supersampling of 3D PIV measurements with vortex-in-cell simulation,” *Experiments in Fluids* (2014) 55: 1692.
- <sup>244</sup> Schneiders, J.F. and Scarano, F., “Dense velocity reconstruction from tomographic PTV with material derivatives,” *Experiments in Fluids* (2016) 57: 139.
- <sup>245</sup> van Oudheusden, B.W., “PIV-based pressure measurement,” *Meas Sci Technol* (2013) 24:032001.
- <sup>246</sup> Schneiders, J.F.G., Caridi, G.C.A., Sciacchitano, A., and Scarano, F., “Large-scale volumetric pressure from tomographic PTV with HFSB tracers,” *Experiments in Fluids* (2016) 10.1007.
- <sup>247</sup> Huhn, F., Schanz, D., Gesemann, S., and Schroder, A., “FFT integration of instantaneous 3D pressure gradient fields measured by Lagrangian particle tracking in turbulent flows,” *Experiments in Fluids* (2016) 57:151.

- 
- <sup>248</sup> Neeteson, N.J. and Rival D.E., “Pressure-field extraction on unstructured flow data using a Voronoi tessellation-based networking algorithm: a proof-of-principle study,” *Experiments in Fluids* (2015) 56: 44.
- <sup>249</sup> Neeteson, N.J., Bhattacharya, S., Rival, D.E., Michaelis, D., Schanz, D., and Schroder, A., “Pressure-field extraction from Lagrangian flow measurements: first experiences with 4D-PTV data,” *Experiments in Fluids* (2016) 57: 102.
- <sup>250</sup> Van Gent, P.L., Michaelis, D., van Oudheusden, B.W., Weiss, P.E., de Kat, R., Laskari, A., Jeon, Y.J., David, L., Schanz, D., Huhn, F., Gesemann, S., Novara, M., McPhaden, C., Neeteson, N.J., Rival, D. E., Schneiders, J.F.G., Schrijer, F.F.J., “Comparative assessment of pressure field reconstructions from particle image velocimetry measurements and Lagrangian particle tracking,” *Experiments in Fluids* (2017) 58: 33.