Implementation of 3DPTV for turbulence analysis and subgrid-scale stress model testing of a backward-facing step flow

Nicholas Dona

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2015

Reading Committee:

Dana Dabiri, Chair

James Riley

Program Authorized to Offer Degree:
UW Aeronautics and Astronautics

University of Washington

Abstract

Implementation of 3DPTV for turbulence analysis and subgrid-scale stress model testing of a backward-facing step flow

Nicholas Dona

Chair of the Supervisory Committee:
Associate Professor Dana Dabiri
UW Aeronautics and Astronautics

Large-Eddy Simulation (LES) is a constantly-expanding field with many new applications and models being introduced on a regular basis. The active nature of this field establishes the need for high-resolution, 3-dimensional experimental data for assessment and development of Subgrid-Scale (SGS) Models. An experimental technique well-suited to this application is Three-Dimensional Particle Tracking Velocimetry (3DPTV) utilizing the epipolar line triangulation method is an accepted method of quantifying volumes of 3-Dimension 3-Component (3D3C) velocity vector fields. This study adapted the epipolar line search triangulation methodology, as it was applied in micro-scale systems, to a backward-facing step flow in a small-scale water tunnel testing facility. The camera system, consisting of three 4008 $\times$ 2672 CCDs, was aligned and calibrated using a custom grid and dot target plate mounted on a purpose-built rig, containing a precision single-axis translation stage. Dual-pulsed, Nd:YAG lasers at 532 nm, 120 $^{\text{mJ}}/_{\text{pulse}}$, illuminated the 28 mm $\times$ 18 mm $\times$ 4.5 mm volume of interest, located downstream of a 2.858 cm step in a 15.24 cm $\times$ 30.48 cm cross-section water tunnel. The turbulent flow, u_{in} =22 $^{\text{cm}}/_{\text{s}}$, $Re_h = 6274$, $R_\lambda \approx 130$, and $ER = 1.208$ was seeded with TiO_2 particles, $<5\mu$m diameter, to maintain one-way coupling. The re-

sulting 3DPTV system was shown to have uncertainty comparable to that of previous experimentation.

Utilizing the acquired data, *a priori* testing of universally notable LES SGS Models, including the Smagorinsky, Similarity, Mixed, Coherent Structures, and Dynamic Models was accomplished and results are presented and discussed. This application of 3DPTV to a turbulent, backward-facing step flow and the results presented herein not only establish the technique as a promising source of experimental data in the development of LES, it lays a foundation for future study of the phenomena-rich backward-facing step flow and the testing and development of new LES SGS models.

TABLE OF CONTENTS

Page

List of Figures iv

List of Tables viii

Glossary ix

Chapter 1: Introduction 1

Chapter 2: Background and Theory 3
2.1 Backward-Facing Step Flow 3
2.1.1 Separated Shear Layer 3
2.1.2 Recirculating Regions and Reattachment Region 5
2.2 Direct Numerical Simulation 7
2.3 Large-Eddy Simulation 8
2.3.1 Subgrid-Scale Models 10
2.3.2 Model Appraisal 18
2.4 Reynolds-Averaged Navier-Stokes 21
2.5 Experimental Approaches 22
2.6 3-DμPTV Method 23

Chapter 3: Experimental Setup and Methodology 26
3.1 Flow Facility 26
3.2 Volume of Interest 29
3.3 Data Acquisition System 30
3.4 Camera System 31
3.4.1 CCDs 32

3.4.2 Lenses . . . 33
3.4.3 Water-Filled Prism . . . 36
3.5 Calibration Configuration . . . 38
3.5.1 Calibration Rig . . . 39
3.5.2 Calibration Target: Custom Grid Dot Pattern . . . 40
3.5.3 Prism Platform . . . 41
3.5.4 Illumination . . . 43
3.6 Acquiring Calibration Data . . . 45
3.7 Data Collection Configuration . . . 47
3.7.1 Laser Illumination . . . 47
3.7.2 Timing . . . 49
3.7.3 Tracer Particles . . . 51
3.8 Acquiring Experimental Data . . . 60

Chapter 4: Data Analysis . . . 61
4.1 Calibration Data Processing . . . 61
4.1.1 Target Plate Line Removal . . . 62
4.1.2 Calibration RBF Development . . . 63
4.2 Experimental Data Analysis . . . 65
4.2.1 Image Processing . . . 65
4.2.2 Peak Finding . . . 68
4.2.3 Triplet Matching and Particle Location . . . 73
4.2.4 Particle Tracking . . . 74
4.2.5 Unstructured Interpolation . . . 83
4.2.6 *A Priori* Testing of SGS Stress Models . . . 84

Chapter 5: Results and Discussion . . . 87
5.1 3DPTV Results . . . 87
5.2 Uncertainty Analysis . . . 90
5.3 SGS Stress Model Study . . . 92
5.3.1 SGS Stress Contours . . . 93
5.3.2 SGS Dissipation Contours . . . 94

5.3.3 Accumulation of Statistics 95
5.3.4 Summary 101

Chapter 6: Conclusions and Future Study 103
6.1 Conclusions 103
6.2 Future Study 105
6.2.1 Data Processing and Analysis 105
6.2.2 Experimental Hardware 108

Bibliography 110

Appendix A: 3DPTV Software Manual 118
A.1 Introduction 119
A.2 Getting Started 119
A.3 Calibration Process 120
A.3.1 Matching Mapping Points and Building RBF 157
A.4 Data Reduction 160
A.5 Batch Data Processing 171
A.6 Particle Tracking 172
A.7 Outlier Detection 173
A.8 SVD Interpolation 173
A.9 Turbulence Analysis 173

Appendix B: Additional Plots 174
B.1 Correlation Coefficients 174
B.1.1 $\Delta = 10\delta$ 174
B.1.2 $\Delta = 5\delta$ 179
B.1.3 $\Delta = 2\delta$ 185

LIST OF FIGURES

Figure Number Page

2.1 Backward-Facing Step Flow Phenomena 4
2.2 Expansion Ratio Dimensions 7
2.3 Graphical representation of LES *a priori* testing 19
2.4 DDPIV Defocusing Concept 23
2.5 3-Dμ-Channel PTV Setup 24

3.1 Water Tunnel 27
3.2 Backward-Facing Step Water Tunnel Section 28
3.3 VOI Placement 29
3.4 VOI Geometry 30
3.5 Data Acquisition Diagram 32
3.6 Previous camera design 33
3.7 CCDs 34
3.8 Schneider-Kreuznach Lens Performance Data 35
3.9 Lens Components 35
3.10 Astigmatism 36
3.11 Water-Filled Prism Ray Trace 37
3.12 Water-Filled Prism and Jigs 38
3.13 Calibration Configuration 39
3.14 Calibration Rig 40
3.15 Translation Stage 41
3.16 Calibration Pattern 42
3.17 Distortion Types 42
3.18 Prism Platform 43
3.19 LED and Diffusor Plate 44

3.20 Calibration Illumination 44
3.21 Data Collection Configuration 48
3.22 Halo Particle Distortion Comparison 49
3.23 Laser Optical Assembly 50
3.24 Timing Diagram 51
3.25 Particle-Turbulence Modulation 55
3.26 Multiphase Coupling Map 56
3.27 Particle Seeding Diagram 59

4.1 Sample Calibration Images 63
4.2 Epipolar Line Calibration Theory 64
4.3 Pre-Processing 67
4.4 Post-Processing 68
4.5 Examples of Image Histogram Distributions 70
4.6 Rosin Thresholding Process 71
4.7 Otsu's Method versus Rosin's Method 72
4.8 Epipolar Line Search Illustration 73
4.9 Example of Vision Association 75
4.10 3DPTV Tracking Program Flow 77
4.11 Turbulent Tracking Validation 82

5.1 Sample Unstructured 3DPTV Results 88
5.2 Sample Interpolated 3DPTV Results 89
5.3 Contour of Spanwise Vorticity 89
5.4 Comparison of Position Uncertainty 91
5.5 Comparison of Displacement Uncertainty 92
5.6 Sample Correlation Convergence 96
5.7 Comparison of SGS Stress Tensor Element Contours 97
5.8 Comparison of SGS Dissipation Contours 98

A.1 Color Separation GUI 121
A.5 Color Channel Crosstalk 126
A.12 Peak Finding GUI 135

A.15 Calibration Peak Finding Results 142
A.16 Calibration GUI 143
A.32 Peak Finding Results 164
A.33 Triplet Matching (PA Locate) GUI 165

B.1 Smagorinsky Stress Tensor Element Correlation Coefficients 175
B.2 Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$ 176
B.3 Mixed Stress Tensor Element Correlation Coefficients 176
B.4 Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$ 177
B.5 CSM Stress Tensor Element Correlation Coefficients 177
B.6 SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$ 178
B.7 SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$ 178
B.8 Smagorinsky Stress Tensor Element Correlation Coefficients 179
B.9 Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$ 179
B.10 Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 2$ 180
B.11 Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 1$ 180
B.12 Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 2$ 181
B.13 Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$ 181
B.14 Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 2$ 182
B.15 CSM Stress Tensor Element Correlation Coefficients 182
B.16 SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$ 183
B.17 SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$ 183
B.18 SGS Force Correlation for all models,$\gamma = 2$,$\alpha = 2$ 184
B.19 SGS Dissipation Correlation for all models,$\gamma = 2$,$\alpha = 2$ 184
B.20 Smagorinsky Stress Tensor Element Correlation Coefficients 186
B.21 Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$ 187
B.22 Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 2$ 187
B.23 Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 1$ 188
B.24 Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 2$ 188
B.25 Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$ 189
B.26 Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 2$ 189
B.27 CSM Stress Tensor Element Correlation Coefficients 190

B.28 SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$ 190
B.29 SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$ 191
B.30 SGS Force Correlation for all models,$\gamma = 2$,$\alpha = 2$ 191
B.31 SGS Dissipation Correlation for all models,$\gamma = 2$,$\alpha = 2$ 192

LIST OF TABLES

Table Number Page

3.1 Calibration Plane Locations 46
3.2 Laser Specifications 49
3.3 Tracer Particle Candidates 57
3.4 Tracer Particle Comparison 58

4.1 Results for 3DPTV tracking algorithm turbulence validation 81

5.1 Correlation Coefficients of Modeled Deviatoric Stress Tensor Elements 100
5.2 Correlation Coefficients of Modeled SGS Force 101
5.3 Correlation Coefficients of Modeled SGS Dissipation 102

B.1 Correlation Coefficients of Modeled Deviatoric Stress Tensor Elements 174
B.2 Correlation Coefficients of Modeled SGS Force 174
B.3 Correlation Coefficients of Modeled SGS Dissipation 175
B.4 Correlation Coefficients of Modeled Deviatoric Stress Tensor Elements 185
B.5 Correlation Coefficients of Modeled SGS Force 185
B.6 Correlation Coefficients of Modeled SGS Dissipation 186

GLOSSARY

LES: Large Eddy Simulation

RANS: Reynolds-Averaged Navier-Stokes

DNS: Direct Numerical Simulation

SGS: Sub-Grid Scale

X_R: Reattachment Length

RE_H: Step Height Reynolds Number

CTR: Center for Turbulence Research, Stanford University

CSM: Coherent Structure Model

GS: Grid-Scale

PIV: Particle Image Velocimetry

HPIV: Holographic Particle Image Velocimetry

3DμPTV: Three-Dimensional Micro Particle Tracking Velocimetry

DDPIV: Digital Defocused Particle Image Velocimetry

CCM: Cascade Correlation Method

SVD: Singular Value Decomposition

3DPTV: Three-Dimensional Particle Tracking Velocimetry

LED: Light-Emitting Diode

RPM: Revolutions per Minute

CCD: Charge-Coupled Device

VOI: Volume of Interest

3D3C: Three-Dimensional Three-Component

DVR: Digital Video Recorder

TDE: Triggered Double Exposure

ND:YAG: Neodymium-doped Yttrium Aluminum Garnet Laser

RBF: Radial Basis Function

PCA: Principal Component Analysis

VOXEL: Volume Element

JHTDB: Johns Hopkins Turbulence Database

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to the University of Washington and the members of the Department of Aeronautics and Astronautics for allowing me to work alongside such professional and knowledgeable engineers, to take part in this challenging course of study, and to pursue this esteemed degree. First, I would like to thank Professor Dana Dabiri, my Principle Investigator, for his constant guidance, creative solutions, and motivations for the research herein. Also, Professor James Riley, Mechanical Engineering, was instrumental to this work due to his esteemed expertise and consistent consultation in the areas of Large-Eddy Simulation and Turbulence; I thank him for his mentorship and instruction. For orientation, guidance, and the very basis for this work, I extend my thanks and gratitude to Professor Wei Tien, National Taiwan University of Science and Technology. I would also like to thank Professor Chris Bretherton, Atmospheric Sciences, and Professor Steven Brunton, Mechanical Engineering, for their consultations on scientific computing. Additionally, I would like to thank Ed Connery, Graduate Advising, for his support and consideration, and Dzung Tran for his kind advice and critical machining resources. Finally, I would like to thank my family and friends for their endless love and support through this daunting journey.

Chapter 1

INTRODUCTION

"Turbulence is the most important unsolved problem of classical physics."

Richard Feynman

Backward-facing step flow is extensively studied in fluid mechanics and is a widely applicable fluid flow across science and engineering disciplines. The results of such studies may be applied to a vast range of flows that experience instantaneous changes in cross-sectional area.[1] While geometrically simple, these flows have several distinct phases. Each phase in the streamwise evolution of a backward-facing step flow has unique characteristics and relevance to many applications.

In a typical realization, previously-attached flow separates into a shear layer concomitant with the backward-facing step. Immediately downstream, a recirculation region, bounded by the detached shear layer, develops. Favoring the low-velocity region of the flow, the shear layer curves and rolls downward, ultimately impinging on the bottom of the channel. Turbulent reattachment occurs, encapsulating the recirculation region. The elegant, multi-faceted, yet elusive nature of this flow makes it a rich problem for turbulence studies.[2]

Turbulence remains one of the most difficult phenomena to study both experimentally and computationally due to the chaotic nature of turbulence and non-linearity of the Navier-Stokes Equations. These challenges result in prohibitive computational costs and excessive resolution requirements for experimentation.

Thus, tantamount to the experimental studies on turbulence are attempts at com-

putationally modeling turbulent flows. Common computational approaches include Reynolds-Averaged Navier-Stokes (RANS), Large-Eddy Simulation (LES), and Direct Numerical Simulation (DNS). In situations where the small-scale (high-wavenumber) turbulent features are not resolved computationally, as they are in DNS, Sub-Grid Scale (SGS) models are required for the completion of the conservation equations along with system closure.

In order to further the understanding of turbulent flows, experimental measurements must be taken to both quantify flows and appraise existing and emerging SGS turbulence models. The experimental data required for such verification often exceed the spatial and temporal resolution of conventional measurement devices. These requirements drive the need for a high-resolution 3-dimensional flow measurement technique such as 3DPTV.

Chapter 2

BACKGROUND AND THEORY

"It doesn't matter how beautiful your theory is, it doesn't matter how smart you are. If it doesn't agree with experiment, it's wrong."

Richard Feynman

In this chapter, a breakdown of the backward-facing step flow is shown. A brief review of turbulence modeling is offered. As LES is a central focus of this study, it is described in detail along with a selection of classical SGS stress models and model testing approaches. Finally, a selection of experimental approaches that provide laboratory data for LES assessment is presented.

2.1 Backward-Facing Step Flow

Regions of separated flow have remained popular regions of study throughout the history of fluid research. Within the realm of this flow type, the backward-facing step flow has been the most commonly-studied flow due to its consistency of separation and wealth of unsteady behaviors.[3] A typical evolution of the backward-facing step flow, including the inflow, separated shear layer, recirculation regions, and reattachment point, is shown in Figure 2.1. The relevant phenomena of the backward-facing step flow are described in detail in Sections 2.1.1 - 2.1.2.

2.1.1 Separated Shear Layer

The separated shear layer initially consists of the boundary-layer flow from the floor of the inlet. At the edge of the step, a consistent and distinct separation point is formed.

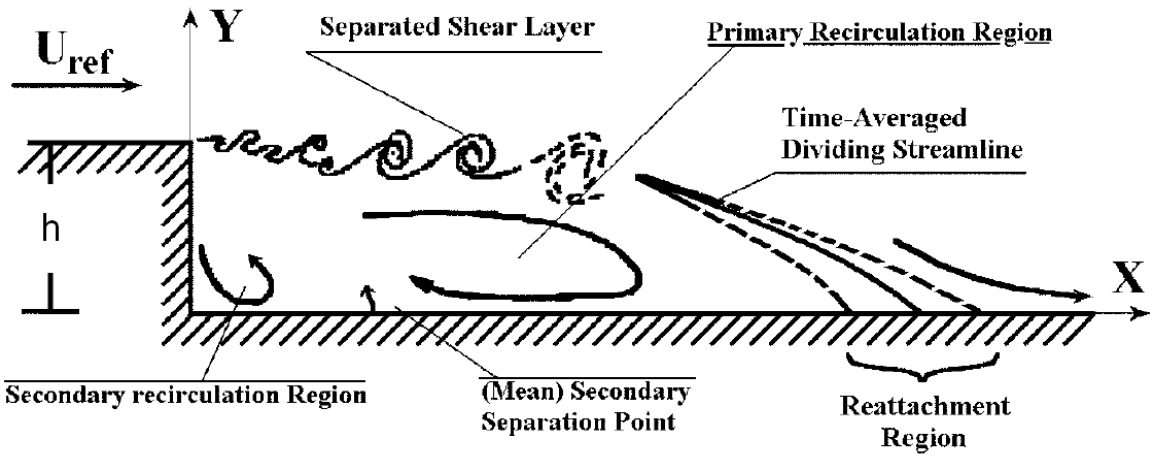


Figure 2.1: Schematic showing relevant phenomena in a typical instance of a backward-facing step flow.[3]

Just after this separation point and reasonably far away from the walls, the flow would be analogous to a free-shear flow, specifically the plane mixing layer. A plane mixing layer is a turbulent layer that forms at the interface of uniform, approximately parallel flows of disparate velocity.[4] In fact, it has been shown experimentally by Troutt *et al.*[5] that structures of span-wise vorticity throughout the separated shear layer of the backward-facing step flow show similarity to those of a mixing free shear layer. As is typical with these flows, simple observation shows that the separated shear layer region of the backward-facing step flow favors the lower-velocity flow (recirculating region) and in this case curves towards and impinges on the tunnel floor.

The separated shear layer, unlike in the case of the developed plane mixing layer, is fundamentally spatially limited, having limited statistical homogeneity due to its proximity to the reattachment region and resulting lack of opportunity to become a fully-developed flow. In addition, this separated shear layer is a vortex sheet that experiences the Kelvin-Helmholtz instability, rolls up, and causes vortex shedding.[3]

Due to the unsteady and inconsistent behavior of this separated shear layer region, considerable attempts have been made to characterize a wide range of individual vortex structures and how they impact behavior of the flow downstream of the region,[6] including behaviors of the reattachment region and reattached turbulent flow.

This unsteady behavior and complex nature make the task of modeling LES SGS stresses a difficult one. Models such as the Dynamic Smagorinsky Model rely upon the ability to average optimized Smagorinsky Coefficients along a direction of statistical homogeneity in order to prevent negative coefficient values that cause computational instabilities.[7] To this end, countless new models have been proposed to allow for the implementation of dynamic models for statistically non-homogeneous and complex flows. One such model involves locally defining coefficients based on coherent structures in the flow.[8,9] Another more general model typically used for complex flows is the Langrangian Dynamic model. The model uses path lines in the flow to acquire the correct coefficients via backward-time integration with decreasing weighting for past events.[10] Models such as these highlight the potential for innovation in the field of LES SGS stress modeling and the need for testing on a variety of flows and high-Reynolds number experimental data. It is this innovation that also drives a widespread need for high-resolution experimental data to enable further *a-priori* testing, described in Section 2.3.2, of newly-proposed models.

2.1.2 Recirculating Regions and Reattachment Region

The separated shear layer is not the only area of interest in the backward-facing step flow. Another area of significant study is the recirculating region contained by the separated shear layer. A vast quantity of research has been dedicated to evaluating the mean flow field of backward-facing step flows in an attempt to establish a mean location for the reattachment region, X_r, for a wide range of step height Reynolds

numbers, Re_h Equation 2.1, by countless experimentalists from Armaly (1983)[1] to Barri (2010),[11] and more recently for a range of channel expansion ratios, ER or the ratio of the post-step channel height to the inflow height Equation 2.2 and Figure 2.2, as well by Nadge *et al.* (2014).[12] Additionally, Kobayashi *et al.* (1992) accomplished LES, using the Smagorinsky Model, to resolve physics of the eddies in the recirculation region in an effort to ultimately determine the location of the reattachment region. In the study, the Smagorinsky Model resulted in a significantly insufficient amount of turbulence in the stream-wise direction,[13] which likely resulted in an overestimated step-normalized reattachment length, X_r/h, of between 9 and 10.

$$Re_h = \frac{\rho u_{in} h}{\mu} \tag{2.1}$$

$$ER = \frac{H + h}{H} \tag{2.2}$$

Closely associated with and encapsulating the recirculating regions is the reattachment region. Having been the focus of many past studies, the reattachment point is an important characteristic of the backward-facing step flow. The location dictates the X-position at which the nature of the boundary layer becomes highly turbulent again and has clear significance in aeronautical applications. While many studies have been done on the effects of Re_h and ER on X_r, the effects of the relationship between the state of the separated shear layer and X_r have proven to manifest themselves even while not the focus of the study.[14]

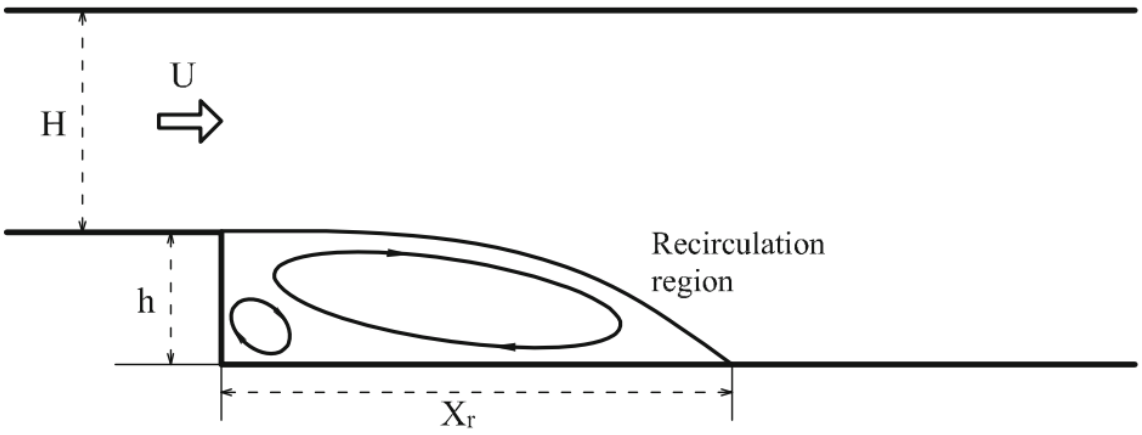


Figure 2.2: Dimensions of expansion ratio labeled with recirculation region and X_r shown.[12]

2.2 Direct Numerical Simulation

Achieving a high-resolution data set with which one may gain insight on the flow phenomena of complex flows, such as the backward facing step flow, is a difficult task. With the development of computational technology, however, DNS has become increasingly feasible. DNS is the process by which the Navier-Stokes Equations are solved in an iterative fashion in order to resolve all scales of motion, from the largest energy-containing motions to nearly the smallest of dissipative motions, the Kolmogorov Scale. DNS is widely regarded as an exceptionally accurate method with unrivaled simplicity.

However, each completed simulation merely corresponds to one instance or realization of the flow field being resolved. Thus, resolving a time series of a flow is an incredibly computationally-expensive process. The computational resources required for DNS are a well-quantified cost. According to Pope (2000),[4] the computational cost of DNS evolves according to Equation 2.3, where T_G is the estimated number

of days for a computing rate of 1 GFLOP to complete a simulation of N^3 total grid nodes and M time steps. This value scales with Reynolds number cubed, Re_L^3, which is approximately equal to the Taylor Microscale Reynolds number to the sixth power, R_λ^6.

$$T_G = \frac{10^3 N^3 M}{10^9 \times 60 \times 60 \times 24} \sim \left(\frac{Re_L}{800}\right)^3 \approx \left(\frac{R_\lambda}{70}\right)^6 \tag{2.3}$$

While the $1\,{}^{\text{GFLOP}}/_{\text{day}}$ metric is certainly obsolete by modern standards, with high-end Intel® Core™i7 Desktop Processors exceeding 350 GFLOPS base speeds, it is most important to note the steep computational cost increase with Reynolds number, which remains a limitation for complex turbulent simulations to this day. It is also important to note that approximately 99% of the computational load is weighted towards resolving the dissipation scale of a turbulent flow, explaining the high Reynolds Number dependence.[4]

This prohibitive computational trend has inspired departures from solving the Navier-Stokes Equations directly. These departures or alternative approaches include solving the large flow scales directly while modeling the more computationally demanding small scales (LES) or simply solving the mean flow equations without the simulation of the dissipative motions (RANS).

2.3 Large-Eddy Simulation

Stemming from the widespread desire to study turbulent flows that have notable engineering significance, LES enables researchers to use a DNS approach on the large-scale motions of a flow while modeling the behavior of the small-scale motions to avoid the associated high computational cost. LES was originally motivated by the meteorological field as a method by which to study turbulent, large-scale atmospheric flows by modeling small-scale motions where the resolution was not sufficient. Pioneered

by Smagorinsky[15] and Lilly,[16] the LES process is a simple one commonly examined in four parts. First, a filter is used to categorize motions as large-scale or small-scale. Next, the Navier-Stokes Equations are filtered with the previously-defined convolution. An additional term, Subgrid-Scale (SGS) Stress, is then constructed and added to the Navier-Stokes Equations to achieve closure. The filtered, modeled Navier-Stokes Equations have been closed and may finally be numerically solved for the large-scale motions of the flow.[4]

$$\widetilde{\mathbf{u}}(\mathbf{x},t) = \int G_{\Delta}(\mathbf{r},\mathbf{x})\mathbf{u}(\mathbf{x}-\mathbf{r},t)d\mathbf{r} \tag{2.4}$$

The initial filtering step may be described in its general form as in Equation 2.4, first proposed by Leonard (1974).[17] In this operation, $G(\mathbf{r},\mathbf{x})$ is the convolution kernel by which the field, in this specific example the velocity field $\mathbf{u}(\mathbf{x},t)$, is convolved at each fine-scale field location, $\mathbf{x}$. This operation yields a filtered field, denoted henceforth by the tilde symbol $\widetilde{(\ )}$, with a coarse, LES grid with spacing Δ. A commonly-used convolution of this type, and the one used in this study, is the Gaussian Filter featuring the Gaussian kernel $G_{\Delta}^{\mathrm{Gaus}}(\mathbf{x})$ in Equation 2.5.[7] This filter type, as with any LES filter, is a function of the LES coarse grid scale, Δ. The standard LES Gaussian filter has a variance of $\sigma^2 = \frac{1}{12}\Delta^2$, as seen in Equation 2.5.[4] Other filters commonly used in LES include the box or tophat filter and the spectral cutoff filter.

$$G_{\Delta}^{\mathrm{Gaus}}(\mathbf{x}) = \left(\frac{6}{\pi\Delta^2}\right)^{3/2}\exp(-6x^2/\Delta^2) \tag{2.5}$$

Next, the filtering operation may be applied to the Navier-Stokes Equations resulting in the filtered form, Equation 2.6.[7]

$$\partial_t\widetilde{\mathbf{u}} + \widetilde{\mathbf{u}}\cdot\nabla\widetilde{\mathbf{u}} = -\frac{1}{\rho}\nabla\widetilde{p} + \nu\nabla^2\widetilde{\mathbf{u}} - \nabla\cdot\boldsymbol{\tau}^{\Delta},\ \nabla\cdot\widetilde{\mathbf{u}} = 0 \tag{2.6}$$

From Equation 2.6, it is apparent that all quantities are filtered (resolved) quantities except for the SGS stress term, τ^{Δ}, which is a residual quantity that must be accounted for in order to allow for closure of the filtered Navier-Stokes Equations. The large-scale motions, greater than the filter scale, Δ, are simulated quantities, exactly as they would be in DNS, by the very nature of LES. It is important to note that in a full application of LES, no filter operation is explicitly applied; the large scales are quantified via the numerical solving of the Navier-Stokes Equations over a coarse grid of spatial resolution Δ, an inherently more efficient process than full DNS. Filtering is only a necessary step in the conceptual development and testing of the LES process and its SGS models. As the SGS motions are not being resolved in LES, this is the step in the process at which modeling of the subgrid-scale stresses, τ^{Δ}, must occur using the modeled SGS stress result, $\tau^{\Delta,\mathrm{mod}}$, of one of the many available models or a newly-designed model. The calculation for SGS stress of a filtered, high-resolution velocity field is shown in Equation 2.7. Calculated SGS stress, τ^{Δ}, is defined as the difference between the filtered stress tensor field, $\widetilde{u_i u_j}$, and the stress tensor field of the filtered velocities, $\widetilde{u}_i\widetilde{u}_j$. The models tested in this study include the Smagorinsky Model,[15,16] Similarity Model,[18] Mixed Model,[18] the Dynamic Model,[19,20] and Coherent Structures Model,[8] all of which are described in the following subsections.

$$\tau^{\Delta} = \widetilde{u_i u_j} - \widetilde{u}_i\widetilde{u}_j \approx \tau^{\Delta,\mathrm{mod}} \tag{2.7}$$

2.3.1 Subgrid-Scale Models

The concept that researchers explore that allows for relating certain large scale features to a limited number of smaller scale properties within a flow is called scale similarity. Conducting a Kolmogorov dimensional analysis on the stress tensor, τ, shows that for Δ in the inertial range, which would be an ideal condition, some

properties of $\xi^{-2/3}\tau^{\gamma\Delta}$ are invariant under the mapping $\Delta \to \xi\Delta$,[7] such that ξ is an arbitrary multiple establishing the lower end of the wavenumber range, within a full energy spectrum, across which scale similarity is assessed. Sometimes these features hold in an average sense or for certain tensor elements only depending on the model in question and the application. In the following subsections, the models utilized in this study to calculate $\tau^{\Delta,\text{mod}}$ are presented with brief explanations and basic historical performance trends.

Smagorinsky Model

One of the most renowned and widely-applied model types is the Eddy-Viscosity model, of which the Smagorinsky Model was an original and notable form. The general Eddy-Viscosity model, Equation 2.8, ultimately provides only the modeled anisotropic or deviatoric portion of the subgrid-scale stress, shown in Equation 2.9 as the traceless SGS stress tensor. The process of removing the trace removes the isotropic stresses leaving only the deviatoric stresses. The model suggests that the deviatoric portion of the modeled SGS stresses would be approximately equal to the calculated SGS stress, $\tau^{\Delta,\text{smag}} \approx \tau^{\Delta,\text{d}}$. The Eddy-Viscosity model consists of the eddy viscosity, ν_T, and the filtered velocity rate of strain tensor, $\widetilde{S}_{ij}$.

$$\tau_{ij}^{\Delta,\text{smag}} = -2\nu_T \widetilde{S}_{ij} \tag{2.8}$$

$$\tau_{ij}^{\Delta,\text{d}} = \tau_{ij}^{\Delta} - \frac{1}{3}\tau_{kk}^{\Delta}\delta_{ij} \tag{2.9}$$

The velocity rate of strain tensor is formed after taking the gradient of the filtered velocity field, $\widetilde{u}(\mathbf{x}, t)$, and applying the resulting gradient and its transpose in

Equation 2.10.

$$\widetilde{S}_{ij} = \frac{1}{2}\left(\frac{\partial \widetilde{u}_j}{\partial x_i} + \frac{\partial \widetilde{u}_i}{\partial x_j}\right) \tag{2.10}$$

In the Smagorinsky Model, the eddy viscosity scalar is approximated by the Smagorinsky length scale, ℓ_S, and the characteristic velocity rate of strain, $\widetilde{\mathcal{S}}$. The Smagorinsky length scale is formed by way of the Smagorinsky coefficient, C_S, and the LES filter width, Δ, Equation 2.11. The characteristic velocity rate of strain is found by way of the Frobenius norm, defined by Equation 2.12.

$$\begin{aligned} \nu_T &= \ell_S^2 \widetilde{\mathcal{S}} \\ &= (C_S \Delta)^2 \widetilde{\mathcal{S}} \end{aligned} \tag{2.11}$$

$$\widetilde{\mathcal{S}} \equiv \sqrt{2\widetilde{S}_{ij}\widetilde{S}_{ij}} \tag{2.12}$$

In this fashion, the deviatoric SGS stress tensor is directly calculated from the resolved velocity field. The standard Smagorinsky Model is a computationally cheap method by which to achieve an estimation for SGS stresses. There have been countless thorough assessments of Smagorinsky Model performance. It is apparent that the Smagorinsky Model is relatively simple, stable, and computationally efficient, which has led to its widespread application. Due to its simplicity and constant coefficient, C_S, however, the model is especially poorly-suited for complex flows. The consensus of these studies is that the Smagorinsky Model is not an adequate model to accurately replicate SGS physics of a turbulent flow.[7] It is implemented in this study as a means for comparison to other models.

Similarity Model

The Similarity Model, initially proposed by Bardina *et al.* (1980),[18] takes the scale-similarity concept and applies it in a very literal sense. In the application of this LES SGS model, a filter is applied to the resolved, coarse grid scale. In the development and testing of this model, two filters are applied as shown in Equation 2.13. This similarity filter, represented by overlines $\overline{(\ )}$ in Equation 2.13, is a process by which the SGS-range stress behavior is assumed to be directly related to that of the similarity filter range.

$$\tau_{ij}^{\Delta,\mathrm{sim}} = C_{\mathrm{sim}} \left(\overline{\tilde{u}_i \tilde{u}_j} - \overline{\tilde{u}}_i \overline{\tilde{u}}_j \right) \tag{2.13}$$

A typical value for C_{sim} is ≈ 1. Similarity filter length scales, Δ^{sim}, may be defined as $\Delta^{\mathrm{sim}} = \gamma\Delta$ such that γ may vary between 1 and 2, depending on the application and individual study. In this study, $\gamma = 1$, the trace of $\tau_{ij}^{\Delta,\mathrm{sim}}$ is not subtracted, and the second filter, Δ^{sim}, is also a Gaussian convolution, identical to the first.

During *a-priori* testing of the Similarity Model by Bardina *et al.* (1980), SGS stress correlation with real values, given by Equation 2.7, was as high as 0.8.[18] It is important to note that some portion of correlation is likely to stem from the filters' spectral overlap, in that some data may be common between modeled and real stresses. This overlap has been studied and found to only contribute a small portion (~ 0.2) of the Similarity Model's typically high stress correlation coefficients.[21] It was also noted, however, that the Similarity Model did not result in adequate levels of SGS dissipation, a certain drawback of the model[18] as this would lead to the improper simulation of the energy-containing and inertial physics.

Mixed Models

In an effort to increase the SGS energy dissipation while not diminishing the high correlation coefficients of the Similarity Model, Bardina *et al.* (1980) proposed the Mixed Model, a simple superposition of the Similarity Model and the Smagorinsky Model, Equation 2.14.

$$\begin{aligned}\tau_{ij}^{\Delta,\mathrm{mix}} &= \tau_{ij}^{\Delta,\mathrm{sim}} + \tau_{ij}^{\Delta,\mathrm{smag}} \\ &= C_{\mathrm{sim}}\left(\overline{\widetilde{u}_i\widetilde{u}_j} - \overline{\widetilde{u}}_i\overline{\widetilde{u}}_j\right) - 2(C_S\Delta)^2\widetilde{\mathcal{S}}\widetilde{S}_{ij}\end{aligned} \tag{2.14}$$

This model largely functioned as intended, as the magnitude of the Similarity Model's contribution to SGS stress is generally much greater than that of the Smagorinsky Model's. Thus, the addition of the Smagorinsky Model provides additional dissipation without diminishing the high correlation coefficient for SGS stresses provided by the Similarity Model.[22] This suggests that the Mixed Model performs better than either of its constituent models.

Dynamic Models

The Smagorinsky Model's weakness of implementing a single global coefficient, in other words its lack of locality, led to Germano's introduction of the Dynamic Model.[19] The Dynamic Model uses the same Eddy-Viscosity form as the standard Smagorinsky Model, but with dynamically determined Smagorinsky coefficients, C_S. The model uses the resolved scales just above the DNS filter scale in a second filtering operation, analogous to the Similarity Model, in order to adjust the Smagorinsky Coefficient locally. In this case, the second filter operation in Equation 2.13 may be seen as a test filter equal to the term $\mathcal{L}_{ij}$, or Germano's Identity, restated by Equation 2.15. The test filter is defined relative to the LES filter such that $\Delta^{\mathrm{test}} \equiv \alpha\Delta$ where the

condition $\alpha = 2$ is typical. From Germano's Identity, the traceless or deviatoric $\mathcal{L}_{ij}$ may be related to C_S by way of the tensor M_{ij} as in Equation 2.16.[7]

$$\mathcal{L}_{ij} \equiv \overline{\widetilde{u}_i \widetilde{u}_j} - \overline{\widetilde{u}}_i \overline{\widetilde{u}}_j \tag{2.15}$$

$$\mathcal{L}_{ij} - \frac{1}{3}\mathcal{L}_{kk}\delta_{ij} = (C_S)^2 M_{ij} \tag{2.16}$$

$$\begin{aligned} M_{ij} &= -2\Delta^2 \left[\alpha^2 \left(\frac{C_S^\alpha}{C_S} \right) \overline{\widetilde{\mathcal{S}}}\ \overline{\widetilde{S}}_{ij} - \overline{\widetilde{\mathcal{S}}\widetilde{S}_{ij}} \right] \text{ where} \\ \Delta^{\text{test}} &\equiv \alpha\Delta \end{aligned} \tag{2.17}$$

Equation 2.17 restates M_{ij} in an alternative fashion in terms of the characteristic rate of strain, $\widetilde{\mathcal{S}}$, and the rate of strain tensor, $\widetilde{S}_{ij}$, with various test filter operations applied, again, denoted by $\bar{(\ )}$. In practice, $\alpha = 2$ and the assumption of scale invariance is applied, resulting in equivalent Smagorinsky coefficients at the different scales, $C_S^\alpha = C_S$, or a ratio of 1. Lilly (1992) introduced the most widely-used version of the Dynamic Model, a square error minimization between modeled and test-range stresses of Equations 2.15 - 2.17,that led to the form in Equation 2.18, in terms of the Leonard Stress Tensor and the tensor M_{ij}.[20]

$$(C_S)^2 = \frac{\langle \mathcal{L}_{ij} M_{ij} \rangle}{\langle M_{ij} M_{ij} \rangle} \tag{2.18}$$

As the coefficients produced by Equation 2.18 are highly variable, the ensemble average, <>, was added to indicate averaging of the coefficients in a single spatial direction, one of statistical homogeneity. This prevents the computational instability in the application of the Dynamic Model that would occur due to fluctuating or negative coefficients, but maintains a 2-dimensional distribution of localized,

dynamically-determined Smagorinsky coefficients for a turbulent flow. These coefficients are directly applied in the Eddy-Viscosity Model, Equation 2.8, as a 2-D field, neglecting the dimension of statistical homogeneity. This dynamically-determined model is highly computationally intensive due to its double-filtering operations and spatially variable coefficients. Additionally, in flows with complex geometries, the lack of statistical homogeneity is prohibitive, giving rise to many alternative models and forms of the Dynamic Model.

Dynamic Mixed Models

In the many applications of the Dynamic Model, countless variations have been proposed. One such variation is the Dynamic Mixed Model, in which the Similarity Model is added to the Dynamic Smagorinsky Model to achieve the benefits of a locally-tuned coefficient, sufficient dissipation, and high correlation coefficients of SGS stress tensor elements. Additionally, in some studies two-coefficient Dynamic Models have been developed to locally determine C_{sim} in addition to C_S. The result of these models in *a-priori* multi-model comparisons have indicated that the Dynamic Mixed Model is superior to both the Dynamic Model and the Mixed Model, but the addition of a second dynamic coefficient tended to diminish stress correlations.[7] Therefore, in this study the two-coefficient Dynamic Mixed Model is not considered.

Coherent Structures Model

Another development by the Center for Turbulence Research, CTR, was the Coherent Structures Model (CSM). A form of the Eddy Viscosity Model, Equation 2.8, the CSM, Equation 2.19, was introduced by Kobayashi (2005)[8] as a method to relate

regions of higher vorticity to higher eddy viscosity, ν_T, values.

$$\nu_T = C_{CS}\Delta^2\widetilde{\mathcal{S}} \tag{2.19}$$

The process utilizes Q, the full second scalar invariant of a full flow field. To adapt the full Q definition for LES, filtered values are substituted for their full counterparts without adjustment for the SGS, as shown in Equation 2.20. When positive, Q indicates greater rotation in a flow field than strain. When negative, the opposite is true. This indicates locations of vortex structures.[8]

$$Q = \frac{1}{2}\left(\widetilde{W}_{ij}\widetilde{W}_{ij} - \widetilde{S}_{ij}\widetilde{S}_{ij}\right) = -\frac{1}{2}\frac{\partial\widetilde{u}_j}{\partial x_i}\frac{\partial\widetilde{u}_i}{\partial x_j} \tag{2.20}$$

The determination of Q involves finding the difference between the magnitude of the filtered vorticity tensor, $\widetilde{W}_{ij}$ Equation 2.21, and the magnitude of the velocity rate of strain tensor, $\widetilde{S}_{ij}$ Equation 2.10.

$$\widetilde{W}_{ij} = \frac{1}{2}\left(\frac{\partial\widetilde{u}_j}{\partial x_i} - \frac{\partial\widetilde{u}_i}{\partial x_j}\right) \tag{2.21}$$

When the second invariant, Q, is normalized by the magnitude of the velocity gradient tensor, E, Equation 2.22, the coherent structure function, F_{CS} Equation 2.23, results.

$$E = \frac{1}{2}\left(\widetilde{W}_{ij}\widetilde{W}_{ij} + \widetilde{S}_{ij}\widetilde{S}_{ij}\right) = \frac{1}{2}\left(\frac{\partial\widetilde{u}_j}{\partial x_i}\right)^2 \tag{2.22}$$

$$F_{CS} = \frac{Q}{E} \tag{2.23}$$

In Equation 2.24, C_1 is given the optimized value based on empirical testing and the addition of F_Ω, the energy-decay suppression function.[9] Finally, the locally-

defined value of C_{CS} is used in the Eddy Viscosity form, Equation 2.19.

$$C_{CS} = C_1 |F_{CS}|^{3/2} F_\Omega \tag{2.24}$$

$$F_\Omega = 1 - F_{CS} \tag{2.25}$$

$$C_1 = \frac{1}{22} \tag{2.26}$$

The CSM was developed using a turbulent channel flow and has been since applied, in an *a-posteriori* sense, described in Section 2.3.2, to the LES of a variety of complex and non-homogeneous flows including a backward-facing step flow, asymmetric plane diffuser, and staggered jets in crossflow with better efficiency and numerical stability than the Dynamic Smagorinsky Model with comparable performance.[9]

2.3.2 Model Appraisal

The development and testing of new SGS stress models requires methods of comparison with experimental data and, depending on the model, application to a variety of flows before widespread usage. Within the realm of SGS model appraisal, there are two major philosophies of testing approaches called *a-priori* and *a posteriori* testing. Originally coined by Piomelli *et al.* (1988), the philosophies are described in further detail in the following sections along with SGS statistical correlations and energy dissipation matching.

A Priori Model Testing

The *a priori* testing approach requires a full data set of high-resolution flow field velocities. The source of these data may be DNS or high-resolution experimental

data. In the *a priori* testing approach, a full execution of LES is never actually completed. Instead, the high-resolution data are filtered using a convolution of the form in Equation 2.4 to achieve a coarse, large-scale grid of the type that would be directly simulated in the first portion of the LES process. Then, the model of interest is applied directly and SGS stresses are solved analytically for the coarse, Δ length scale velocity grid. An excellent visual representation of this process is presented by Liu *et al.* (1994) in the *a priori* testing of a turbulent jet flow field, Figure 2.3.[21]

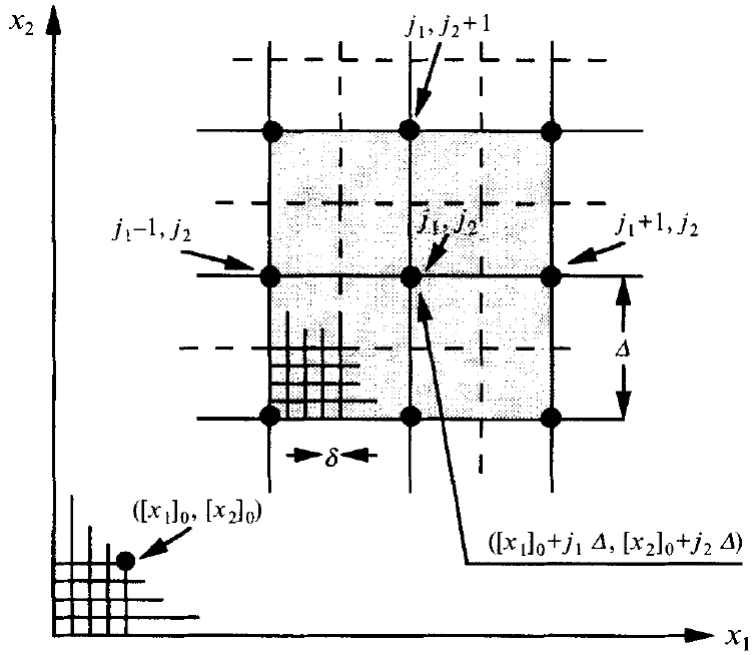


Figure 2.3: A graphical representation of LES *a priori* testing. δ is high-resolution data spacing. Δ is coarse, resolved (or fitered) LES scale data.[21]

The stresses may then be directly compared with the known values of the large-scale field, which are calculated from the high-resolution data set. This amounts to a direct comparison between $\tau^{\Delta}(\mathbf{x}, t)$, calculated by Equation 2.7, and $\tau^{\Delta,\mathrm{mod}}(\mathbf{x}, t)$,

approximated via a model of interest. The advantages of *a priori* testing mainly revolve around the practicality of the approach; it is possible to finely tune and apply a model repeatedly with limited computational expense and a short turnaround time. It also allows for direct comparison of the effects of the models on a small scale. If applied fully in an LES, it is less feasible to distinctly measure the ability of the SGS stress models to adequately model the stress on a fine level. This approach, however, is generally seen as less rigorous due to the fact that no standalone LES is actually being accomplished and the fact that the model is being optimized for the specific application at hand rather than universally.

A Posteriori Model Testing

The alternative and more stringent approach is *a posteriori* testing. In this approach, a full, high-resolution data set is selected only as a means for comparison to the LES results. A full LES is then accomplished with parameters that match those of the DNS or experimental results. The model for SGS stress is applied in the LES process and the end results are compared directly to the experimental or DNS results. This comparison is generally seen as a more ultimate assessment of the applicability of a model. Again, the drawback of this approach is the finer details of SGS qualities of a model may be buried in the overall LES execution. In either case, statistical methods must be applied to determine the quality of the simulation results.

SGS Correlations

Statistical linear correlation is used to measure how well the SGS stress models reproduce the SGS stress relative to the known quantities. Correlation coefficients, where ρ is the Pearson Correlation Coefficient, may be measured for individual SGS stress deviatoric tensor elements, for instance $\rho\left(\tau_{12}^{\Delta}, \tau_{12}^{\Delta,\mathrm{mod}}\right)$, the SGS stress force vector,

$\rho\left(\nabla\cdot\tau^{\Delta},\nabla\cdot\tau^{\Delta,\mathrm{mod}}\right)$, or for SGS energy dissipation, $\rho\left(\tau_{ij}^{\Delta}\widetilde{S}_{ij},\tau_{ij}^{\Delta,\mathrm{mod}}\widetilde{S}_{ij}\right)$. A combination of these statistical assessments have been applied in this study. Also presented are comparative contour plots of SGS stress deviatoric tensor elements from modeled and calculated SGS stresses.

Energy Dissipation Statistics

Another aspect of model assessment is how well the SGS model reproduces energy dissipation statistics. This flow aspect is generally seen as a minimum requirement that does not necessarily extend to a measure of merit. SGS energy dissipation may be tested in a local sense, as mentioned previously, or in a mean sense as in the evolution of mean kinetic energy, Equation 2.27.[23]

$$\frac{\partial\mathcal{K}}{\partial t}+\langle\widetilde{u}_i\rangle\frac{\partial\mathcal{K}}{\partial x_j}=-\frac{\partial A_j}{\partial x_j}-2\nu\left\langle\widetilde{S}_{ij}\widetilde{S}_{ij}\right\rangle-\Pi^{\Delta}+\left\langle\widetilde{f}_i\widetilde{u}_i\right\rangle \tag{2.27}$$

In the kinetic energy equation, mean kinetic energy is defined as $\mathcal{K}=\frac{1}{2}\langle\widetilde{u}_i\widetilde{u}_i\rangle$. The A_j term is a flux of grid-scale velocity third-order moments, pressure, SGS stress, and transport of viscous stress.[7] The kinematic viscosity term is the molecular dissipation of resolved energy, a measure that should be negligible for Δ placed squarely in the inertial range of a well-developed turbulent energy spectrum. The final term is body forces, and $\Pi^{\Delta}=-\left\langle\tau_{ij}^{\Delta}\widetilde{S}_{ij}\right\rangle$ is a sink of GS kinetic energy, which should be directly equal to the energy dissipation rate, ε, when Δ is in the inertial range and SGS kinetic energy is at an equilibrium state.[7] This assessment, presented as a basic requirement, was established by Piomelli *et al.* (1991).

2.4 Reynolds-Averaged Navier-Stokes

The RANS process is a method by which the mean velocity field of a flow may be solved by way of solving the Reynolds equations for the Reynolds stresses using a

turbulence model. While the RANS method may provide some useful insight to the mean features of a flow, such as the mean reattachment length, X_r, of a backward-facing step flow, it is insufficient to determine fine-scale turbulent qualities of a flow at a given instant due to the lack of full simulation of the flow field.

2.5 Experimental Approaches

To allow for validation and development of LES SGS models, high-resolution experimental or DNS data are required. While DNS is an excellent source for complete, high-resolution data at lower Reynolds numbers and simpler geometries, experimentation allows for higher Reynolds numbers in more complex flows.[7] The switch to experimental data, however, may come with the trade off of less complete data or, in some cases, fewer spatial dimensions.

In the past studies, many studies using various experimental approaches to test LES SGS models have been accomplished. Many notable contributions to the LES modeling field have come from the groups of Professors Katz and Meneveau, Johns Hopkins University. In Liu *et al.* (1994) and in another study by the same in 1995, 2-D Particle Image Velocimetry (PIV) was applied to the computationally complex turbulent jet flow at a Reynolds number of $R_\lambda \approx 310$, where R_λ is the Taylor-Scale Reynolds number given by Equation 2.28.[21,22]

$$R_\lambda \equiv \frac{u'\lambda}{\nu} \tag{2.28}$$

$$\lambda \approx u'\sqrt{\frac{15\nu}{\varepsilon}} \approx \sqrt{\frac{15\nu l}{u'}} \tag{2.29}$$

The Taylor Microscale, λ, is approximated prior to data acquisition by the estimated velocity fluctuations, u', characteristic length scale, l, and kinematic viscosity, ν. Later, Holographic Particle Image Velocimetry (HPIV) of a square turbulent duct

flow was accomplished by Tao *et al.* (2002) for an *a priori* test of LES SGS models on relatively isotropic turbulence.[24] The trends from the experimental data acquired in these studies showed reasonable agreement with those exhibited by lower Reynolds number DNS data, suggesting experimental data are a feasible and even preferred alternative to DNS data.

2.6 3-DμPTV Method

The high-resolution 3-Dimensional experimental method used to acquire turbulent backward-facing step flow data in this study is an adaptation of the 3-DμPTV method using the triangulation approach, developed by Tien.[25,26] The original system development is a variation of the Digital Defocused Particle Image Velocimetry, DDPIV, approach described in full detail by Pereira *et al.* (2000). The defocusing concept may be seen in Figure 2.4, which contains the DDPIV system schematic.

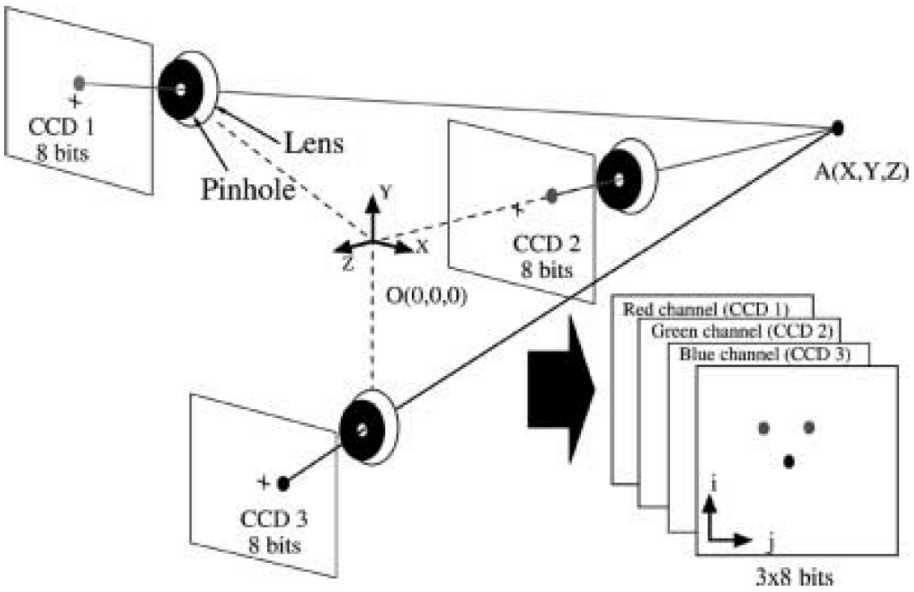


Figure 2.4: The DDPIV defocusing concept for a 3-CCD camera.[27]

The central theory of the DDPIV concept is the defocusing of a particle image onto the imaging plane, a phenomenon dependent on the Z-distance away from the focal plane of the camera system. When using a 3-pinhole plate or a 3-CCD and 3-aperture setup, as in Figure 2.4, a particle casts a unique image pattern on the resulting 3-channel image generated when the gray-level outputs of each CCD are superimposed to create a color, RGB image. The non-uniqueness of particles, especially problematic in flows with high particle densities such as the one featured in this study, prohibits the use of a 2-camera defocusing system.[25,28] The full 3-DμPTV channel setup developed by Tien is shown in Figure 2.5 with a system schematic.

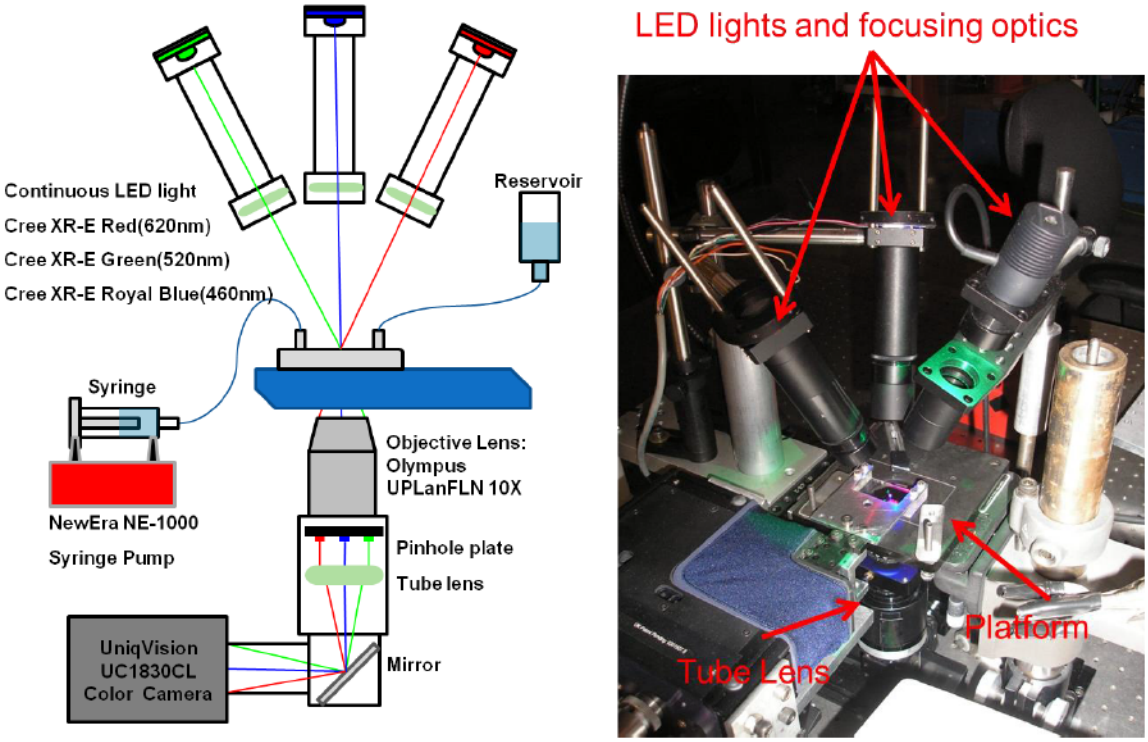


Figure 2.5: Full 3-DμPTV channel flow setup by Tien with system schematic[25]

Once defocused particle images have been acquired and post-processed, prices, sub-pixel peak locations are found using the Cascade Correlation Method (CCM) originally developed by Angarita-Jaimes *et al.* (2009)[29] and refined by Tien (2014) as the Modified CCM.[26] Once peak locations are found in image coordinates, triangulation utilizing a calibration mapping data set is accomplished to map each unique particle image triplet from image coordinates to physical coordinates. From the two sets of physical particle locations, tracking by computer vision utilizing the Singular Value Decomposition (SVD) is used as developed by Scott and Longuet-Higgins (1991).[30] Finally, tracking outliers are found and removed by a an unstructured Median Outlier filtering method in 3-dimensions.[25] The full method described here was adapted to 3DPTV of larger-scale flows for use in a 15.24 cm $\times$ 30.48 cm cross-section water tunnel. The data analysis techniques used in this study, including modifications and details unique to this application, are described in further detail beginning in Section 4.2.

Chapter 3

EXPERIMENTAL SETUP AND METHODOLOGY

"Science, my boy, is made up of mistakes, but they are mistakes which it is useful to make, because they lead little by little to the truth."

Jules Verne

Details of the experimental setup used in this study are described in both calibration and full data acquisition configurations over the course of this chapter. Further methodology regarding the post-processing and data analysis are developed in detail in Chapter 4.

3.1 Flow Facility

The flow facility used in this study was an Engineering Laboratory Design, Inc. 15.24cm × 30.48cm design test section water tunnel, pictured in Figure 3.1. The velocity range was continuously variable with a theoretical operational maximum of $76\,^{\mathrm{cm}}/_{\mathrm{s}}$ and a variation over the cross-section of $\pm 1\%$ of the mean velocity.

The standard constant rectangular test section was replaced with a backward-facing step flow test section. The backward-facing step designed for previous experimentation was altered by removing the copper heating plate from the section just downstream of the step. This panel was replaced with a precision-milled acrylic section for optical purposes, allowing for LED back-lighting during calibration. A side-by-side comparison of the new and old backward-facing step tunnel sections are pictured in Figure 3.2.

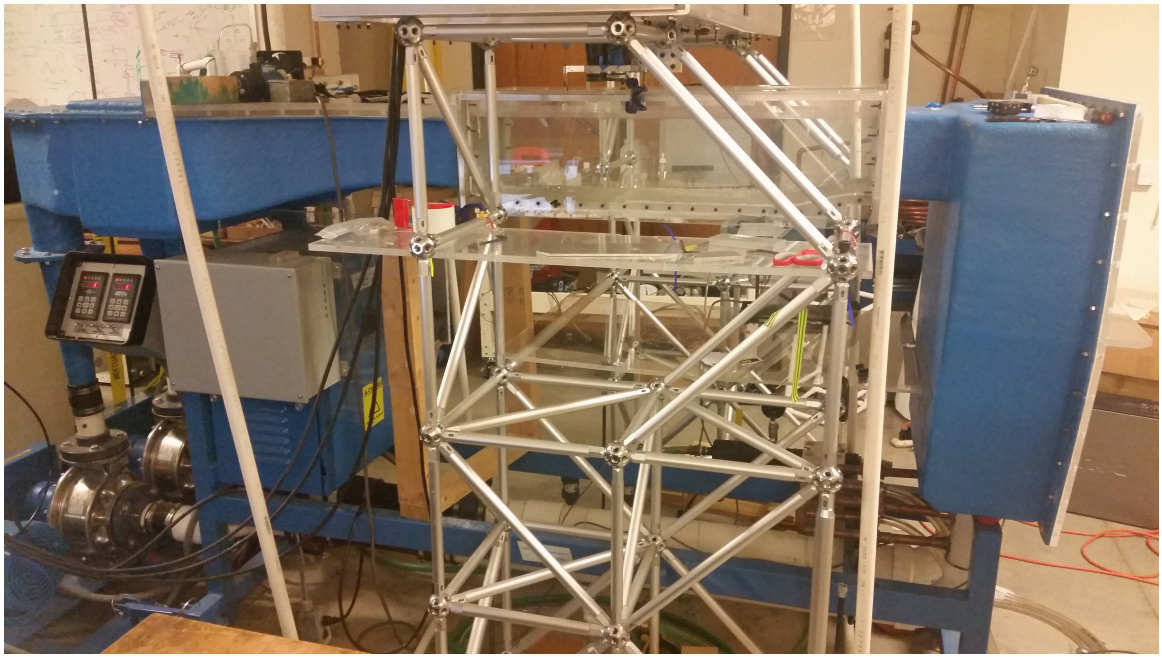


Figure 3.1: Water tunnel used in the study

The expansion ratio of the backward-facing step section is 1.1875, as defined by Equation 2.2 and Figure 2.2, with $H = 15.24$cm and $h = 2.8575$cm. While the maximum pump RPM setting for each of the 1.5 HP motors was 1770, the pump setting slope was calibrated for the operating range in this study. A 60 cm length of the test section was used in conjunction with repeated dye visualization at the channel center to quantify the mean tunnel velocity. The resulting operating slope was $15.5\,^{\mathrm{RPM}}/_{\mathrm{cm/sec}}$ $\pm 5\%$ for both pumps set in following mode.

The velocity used in this study was $22\,^{\mathrm{cm}}/_{\mathrm{sec}}$, pump setting of 341RPM, resulting in a Reynolds number of $Re_h = 6274$, as defined by Equation 2.1, and a Taylor scale Reynolds number of $R_\lambda = 130$, as approximated by Equation 2.28. For this study, velocity fluctuations, u', were predicted at 18% of the mean flow velocity, a typical value for a free-shear flow according to Pope (2000).[4] The Kolmogorov scale,

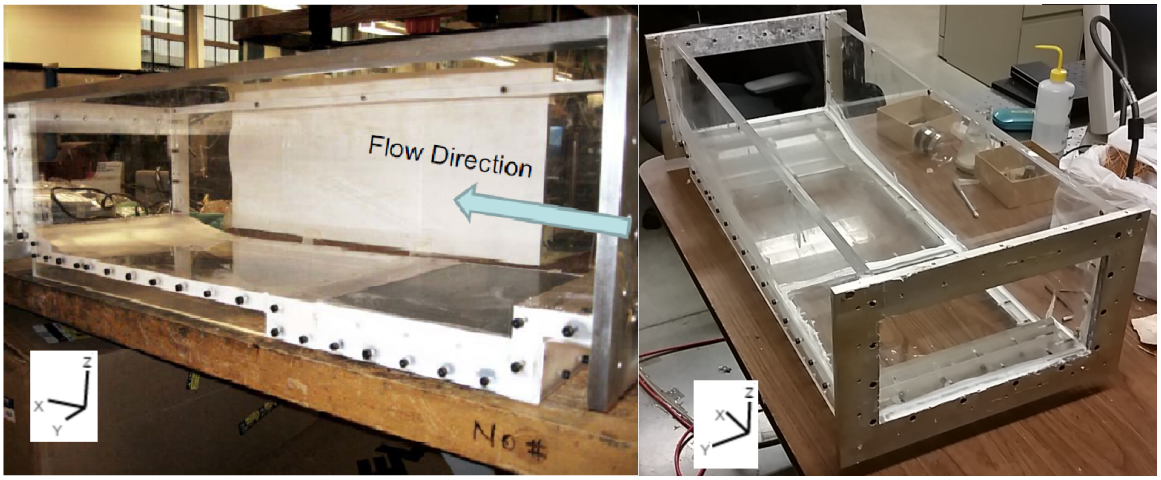


Figure 3.2: Water tunnel test section containing backward-facing step. Heated copper plate (left) from previous experimentation[2] was replaced with a clear acrylic plate (right) for optical purposes. Coordinate system shown relative to step.

the length scale corresponding to the smallest eddies in the flow, was calculated to be $\eta \approx 147\mu$m according to the approximation in Equation 3.1,[4] where ν is the kinematic viscosity of water, and $l = h$ as a characteristic length of the flow geometry. The coordinate system used in the study may be seen in Figure 3.2 and was a right-hand coordinate system aligned with the X-axis in the direction of the inflow, the Y-axis along the edge of the step, and the Z-axis vertically upward.

$$\eta \approx \left(\frac{\nu^3 l}{u'^3}\right)^{1/4} \tag{3.1}$$

The velocities and Reynolds numbers used were dictated by the spatial resolution attainable by the system and data reduction methods, described in further detail over the course of the next two chapters.

3.2 Volume of Interest

The Volume of Interest, VOI, in this study was positioned at $6.5h$ downstream of the step, 13mm above the tunnel floor, and centered in the Y-dimension, as pictured in Figure 3.3, to enable study of the separated shear layer just prior to reattachment, discussed in Section 2.1.1. The size of the VOI was determined such that the camera system would be capable of resolving motions near the Kolmogorov scale of the flow, $\eta \approx 147\mu$m, at a comparable level to that of DNS approaches, which have finest spatial resolutions set relative to the Kolmogorov scale at $\delta = 2.1\eta$.[4] The high-resolution interline transfer Charge-Coupled Devices, CCDs, discussed in further detail in Section 3.4.1, enable high-resolution imaging at these length scales at the design Reynolds numbers, however there is room for great improvement in Reynolds numbers, as discussed in Chapter 6.

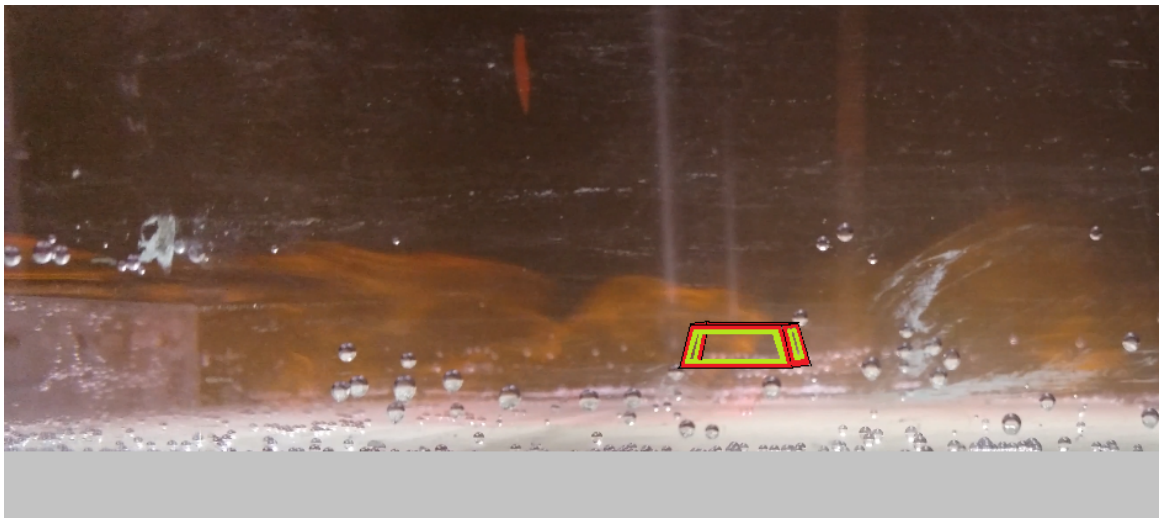
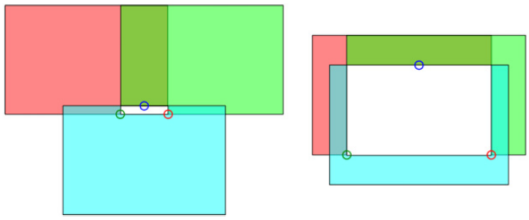
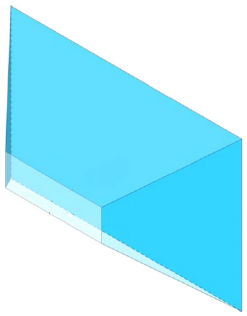


Figure 3.3: Approximate placement of VOI, outlined in black, red, and green, with dye visualization of the backward-facing step flow.

The shape of the experimental VOI was a skewed rectangular prism, based on the imaging volume definition developed by Grothe and Dabiri (2008), as shown in Figure 3.7b, which were derived based on the field of view of each CCD within the full camera, shown in Figure 3.7a.[31] While not as exaggerated as pictured in Figure 3.7b, the shape of the VOI was a skewed rectangular prism with the top of the VOI slightly smaller than the bottom. The Z-depth of the VOI was 4.5mm with a bottom plane dimension of 30.49mm × 20.30mm and an upper plane dimension of 29.71mm × 19.81mm. These measurements were determined via the precision calibration plate described in Section 3.5.2. The mean system magnification through the VOI was determined to be approximately 1.20.



(a) Camera's valid overlap range of CCDs, top (left) and bottom (right)



(b) Skewed rectangular prism shape of camera's resulting VOI

Figure 3.4: VOI geometry based on CCD overlap dimensions[31]

3.3 Data Acquisition System

The data acquisition system used in this study consisted of a computer with Digital Video Recorder (DVR) hardware, a signal generator board, and an external, independent hard disk array. The DVR computer was designed and built by IO Industries

and included three DVR Express CLFC Dual-Base boards, each with two Camera Link Cable interfaces. These boards are also linked to the internal signal generator and are capable of precise synchronization with the laser illumination source via four BNC outputs, two flashlamp and two Q-switch outs with independent settings. The camera exposure and gain settings were adjusted using the XMV Camera Control Application version 2. The cameras and laser timings were set using the DVR Express CLFC Control Signal Manager module from within Streams 5, which controlled the master signal generator board, to which the DVR Express and BNC output boards were slaved. The hard disk array contains eight additional drives, linked by interlink adapters, to supplement the four on-board drives in the DVR computer. Each of the 12 hard disks is an identical Seagate drive with over 128GB of formatted storage space.

A diagram of the complete acquisition system, including the signal generator, DVR boards, hard disks, cameras, and laser is presented in Figure 3.5. The DVR boards are slave to the on-board signal generator internally and transmit the timing signal via the Camera Link Cables to the CCDs. The signal generator also synchronizes the timing signal with the BNC outputs to the laser's flash lamps and q-switch delays. The DVR boards capture the images from the CCDs and write to either the internal disks directly or the external chassis disks via interlink connectors.

3.4 Camera System

The camera system in this study was re-designed from previous experimentation. Originally, the camera system consisted of six CCDs, three for the 3-Dimensional velocity measurement and three for the liquid crystal thermometry color camera, as seen in Figure 3.6. The camera system used in this study did not contain a center color camera or lens. Each CCD was still mounted on a high-precision 6-axis translation

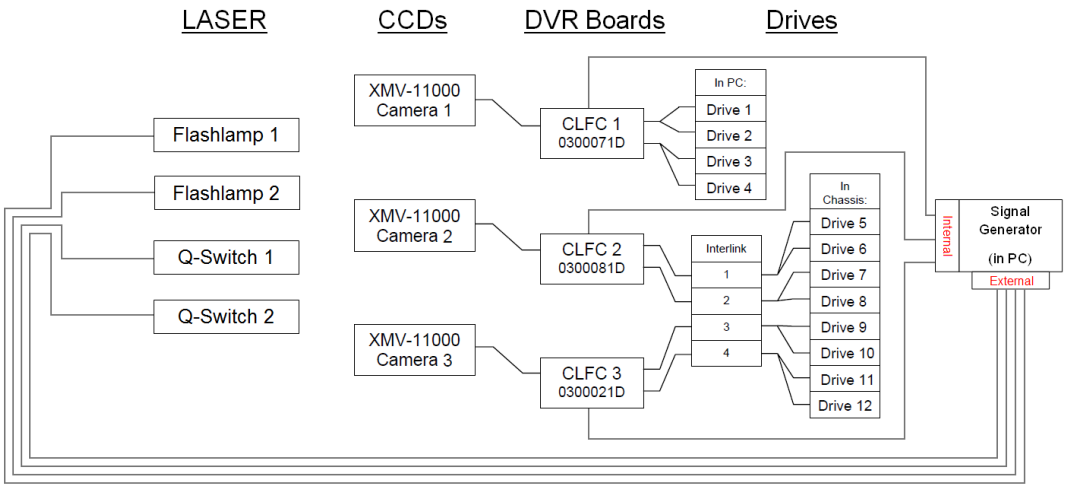


Figure 3.5: Data acquisition equipment control diagram. The DVR boards are slave to the signal generator internally, which also controls the laser's flash lamp and q-switch timing externally.

stage from Thorlabs. As the camera did not have a central color camera, all CCDs were more compact, forming an equilateral triangle with a shorter side length of 12cm. The same was true for the lens assembly, which had an equilateral triangle side length of 6cm. Each CCD was shrouded to its respective lens with black plastic sheeting.

3.4.1 CCDs

The camera system used in this study contained three high-resolution area scan Illunis XMV-11000 grayscale cameras. The cameras contained Kodak KAI-11000 interline transfer CCDs with 4008(h) $\times$ 2672(v) resolution consisting of 9.0 μm square pixels, a well depth of 60,000 electrons, and a signal to noise ratio of 30dB.[32] Maximum frame rate was 3.8 hz, however the cameras were manufactured with a special Triggered Double Exposure (TDE) mode for PIV-type applications. Operating in this mode allowed the cameras to take two consecutive exposures with as little as 800 ns interval.

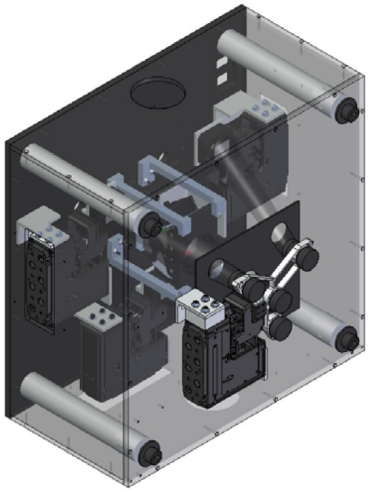
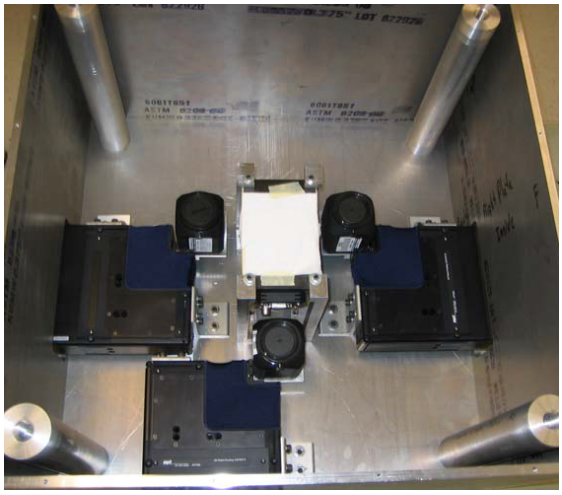


Figure 3.6: Previous camera design for 3D velocity and temperature measurement. The camera used in this study was of a similar form without a central color camera or central lens.[2]

This minuscule time interval would allow for an image pair Δt well below those used in this study, making imaging and analysis of much faster flows a possibility. In theory, the system could image a flow with peak velocities of 90 $^{m}/_{s}$ with a reasonable (10 pixel) maximum displacement.

3.4.2 Lenses

Critical to the 3DPTV camera system were the three precision-manufactured Schneider-Kreuznach Optics Macro-Symmar HM 120mm f/5.6 lenses. The lenses were a macro variation with high-modulation glass components. Performance data for these lenses are shown in Figure 3.8. This lens had been selected due to the highly precise focal length, 119.9mm ±1%, high transmittance, > 90% for a majority of the visible spectrum in Figure 3.8a, and minimal edge distortion, < 0.5% in Figure 3.8b, all resulting in a low level of error propagation to the remainder of the system.[33] Addi-



(a) CCDs on 6-axis translation stages[33]

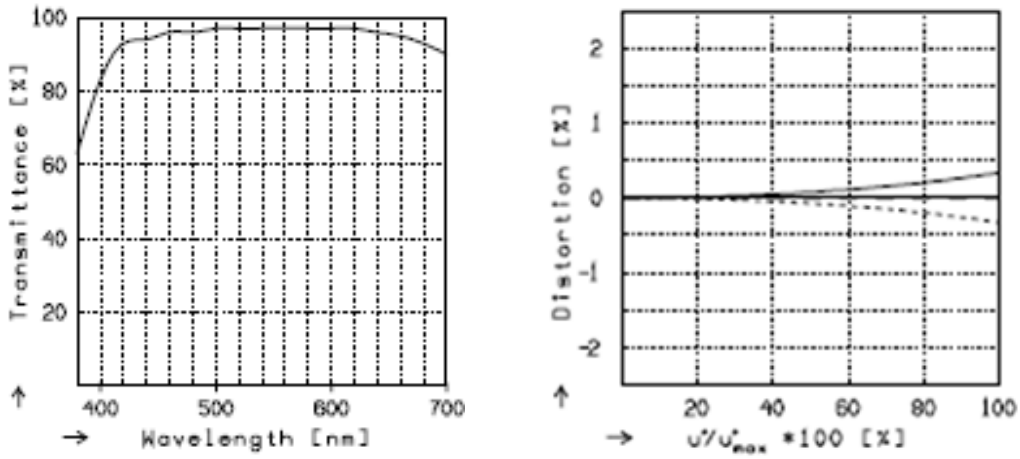
(b) The Illunis XMV-11000 CCD[32]

Figure 3.7: CCDs used in the study. CCD equilateral triangle side length reduced to 12cm for this study.

tionally, the lenses had a relatively high F-number, approximately 3.7, as calculated by Equation 3.2,[33] or simply focal length normalized by lens diameter.

$$F = \frac{f}{D} \tag{3.2}$$

It is important to note that high F-number lenses have proven effective in reducing spherical aberrations that usually cause particle triplet patterns to have undesirable skewness or asymmetry.[34] These triplet errors would result in decreased particle location accuracy, a measure of merit that is critical for systems with high spatial resolution such as the one in this study. A cross-sectional view of the lens design, dimensions, and its components is shown in Figure 3.9.



(a) Lens transmittance over the visible spectrum

(b) Lens distortion versus radial position (percent radius)

Figure 3.8: Schneider-Kreuznach Optics Macro-Symmar HM 120mm f/5.6 lense performance data.[33] (Courtesy of Schneider Optics)

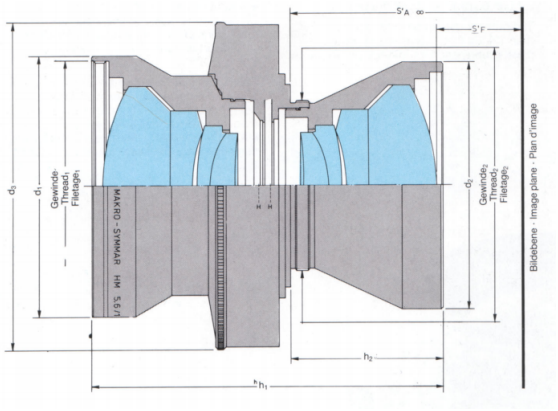


Figure 3.9: Makro-Symmar HM Lens components cross-section view[35]

3.4.3 Water-Filled Prism

Due to the fact that the camera in this study contained CCDs that were off of the optical axis, significant refraction through the acrylic water tunnel lid and water within would result in astigmatism and warped particle images. It has been well-established by Schmitt (2007)[33] and Grothe (2008)[2] through experimentation and ray-tracing theory that a water-filled prism was a necessity for applications such as the one featured in this study. A simple ray tracing diagram exemplifying this minimization of refraction provided by a water-filled prism is also shown in Figure 3.11.

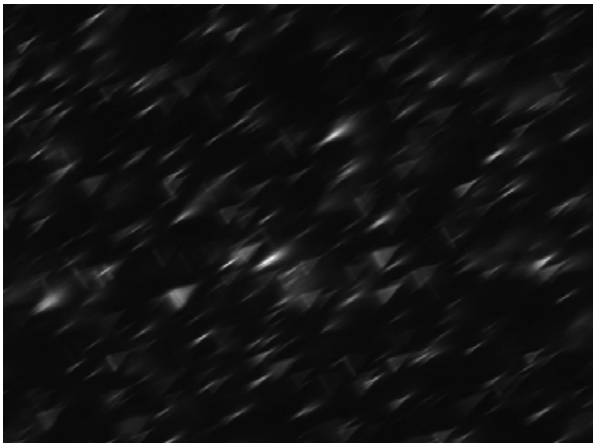


Figure 3.10: Astigmatism caused by disparate refractive indices of air, acrylic, and water. The problem is greatly exacerbated by the fact that the CCDs used in this study were offset from the central optical axis.[33]

The water-filled prism, Figure 3.12a, had been custom-designed and manufactured out of acrylic with refillable water chambers using a Luer Plug system. Prior to each use, each of the three chambers was filled fully with deionized water using a syringe such that there were no bubbles in the chambers. Any air bubbles in the water-filled prism would lead to undesirable distortion caused by refraction in the

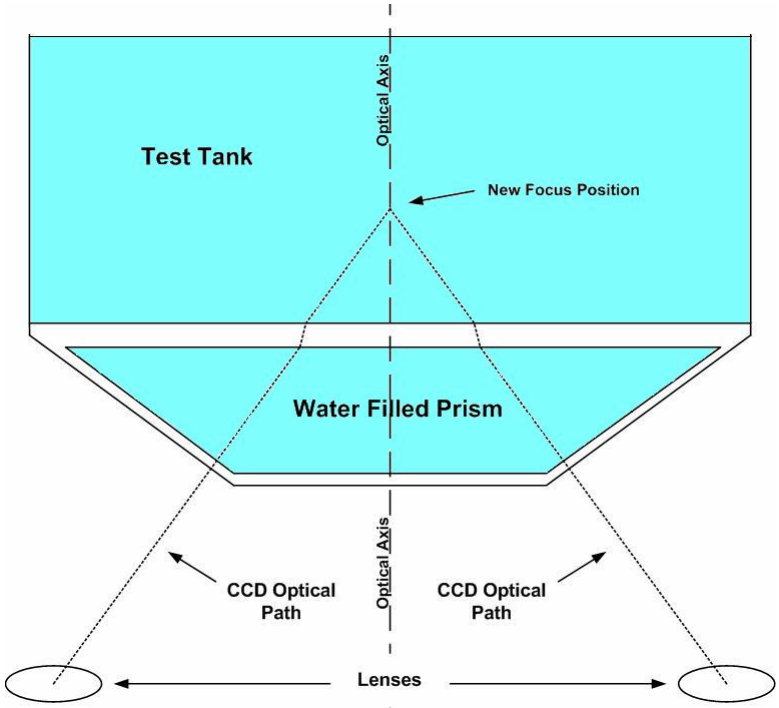
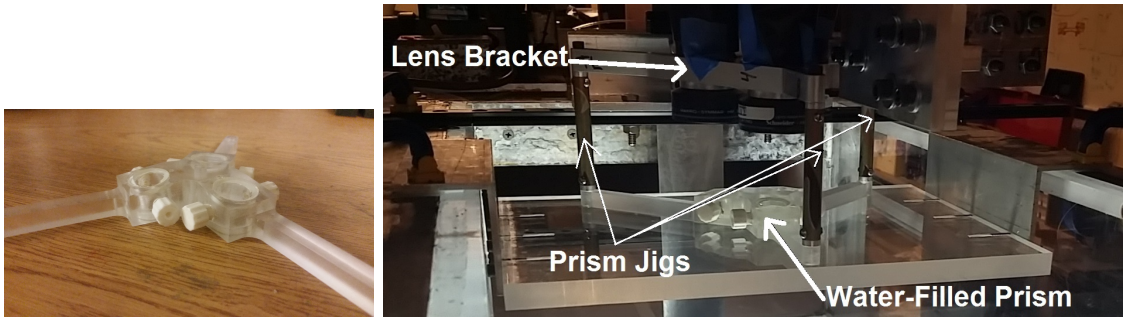


Figure 3.11: A ray-tracing diagram demonstrating the refraction minimized by a water-filled prism[33]

resulting particle images. The water-filled prism was then either placed directly on the acrylic calibration platform, as in Figure 3.12b, for use during calibration or directly on the acrylic tunnel lid for use during the data collection process. In order to insure consistent placement of the prism jig between calibration and data collection, aluminum jig shafts were screwed into the lens bracket and fit the prism's legs in a fixed location to within $\approx 50\mu$m in each direction, as seen in Figure 3.12b.



(a) Water-filled prism. (b) Water-filled prism precisely positioned using jig.

Figure 3.12: Water-filled prism with mounting on lens bracket with jigs.

3.5 Calibration Configuration

In order to allow for physical location mapping of the particle image triplets, calibration data were taken throughout the full volume of interest in the configuration shown in Figure 3.13. To accomplish this, a calibration target plate was mounted on a calibration rig, developed for use in this study, featuring a single-axis high-precision translation stage and displaced to 11 Z-locations and imaged at each plane. The images were taken using continuous back-lighting generated by an LED array and a diffusion plate, both assembled directly below the acrylic floor of the water tunnel test section. The water tunnel was filled, but did not have the lid on in order to allow for access to the calibration rig's translation stage. In order to prevent a free surface and to match all system indices of refraction that would occur during data acquisition, an acrylic prism platform was designed and fabricated for this application of 3DPTV.

While Figure 3.13 provides a general overview of the entire calibration configuration, detailed descriptions and explanations of the design and use of each part are presented over the course of the following subsections. The data reduction methods used to process the calibration data are discussed in Chapter 4.

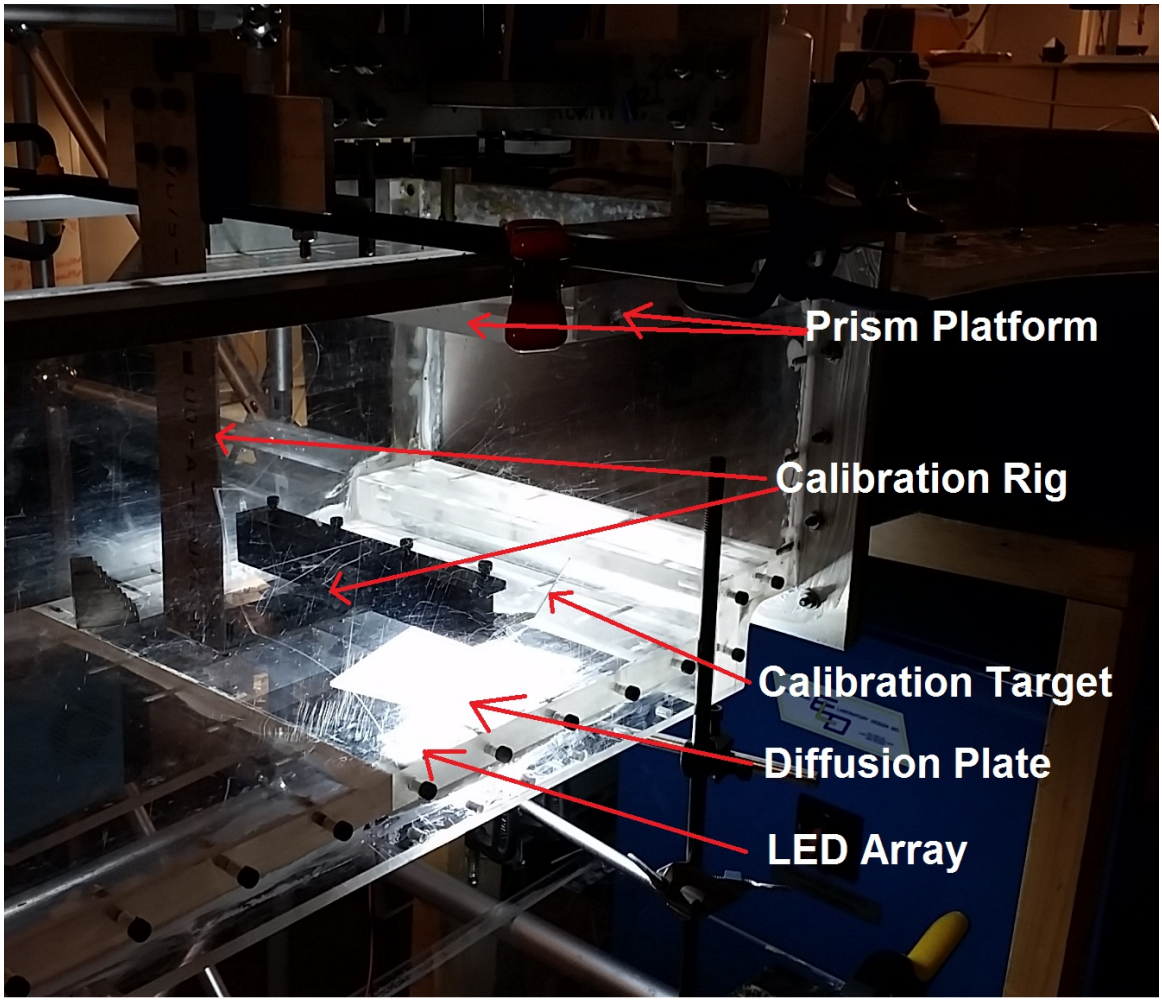


Figure 3.13: A full view of the calibration configuration.

3.5.1 Calibration Rig

The calibration rig, pictured in Figure 3.14, was designed to span the top of the water tunnel test section. Clamps were used to secure the beam to the tunnel flanges. A Mitutoyo 0.5in single-axis linear translation stage, Figure 3.15, with 0.001in increments was fixed to the beam at 90^o. An L-shaped extension was screwed into the translation stage and attached to an adjustable mount with screws and rubber strips to fix the

calibration target plate to the rig without damaging the glass. This setup allowed the target plate to be accurately placed at each Z-location for calibration imaging. The design of the L-shaped metal extension allowed for use of the translation stage with the target plate fully submerged, reaching the full extent of the VOI and extending directly beneath the camera system and prism platform just above the tunnel floor as seen in Figure 3.13.

Figure 3.14: The calibration rig used in this study consisted of a single-axis translation stage and calibration target plate.

3.5.2 Calibration Target: Custom Grid Dot Pattern

The calibration target used in the study was a 1.59mm thick glass plate with a 40.5mm $\times$ 40.5mm regular, repeating etched grid and dot pattern of the configuration

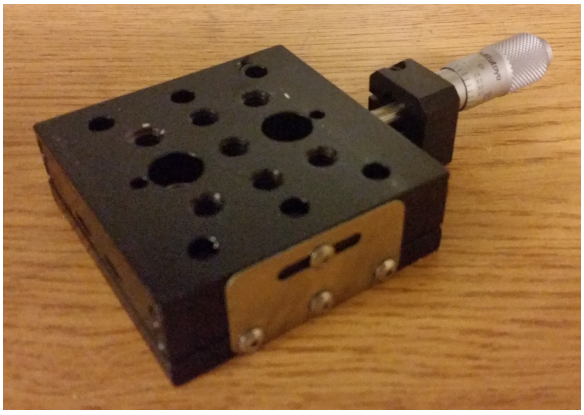


Figure 3.15: The single-axis translation stage featured in the calibration rig

and dimensions shown in Figure 3.16. Each grid line and dot were spaced at a constant interval of 526.2μm $\pm$0.02μm. Each line had a thickness of 26.2μm $\pm$2μm and each dot had a diameter of 78.5μm $\pm$2μm.

The lines that composed the calibration target aided in preventing distortions that would stem from camera misalignment as seen in Figure 3.17. The dots of the calibration target provided discrete physical locations with known X, Y, Z coordinates to allow for the building of a calibration mapping linking to image coordinates, x, y, that resulted from each of the three cameras. The calibration process is described in further detail in Chapter 4.

3.5.3 Prism Platform

Open access to the tunnel test section for translation of the calibration target drove the need for a platform on which to stage the water-filled prism. This platform allowed the calibration configuration to match the indices of refraction experienced in the data collection configuration while preventing a free-surface above the VOI. A stock piece of acrylic was machined to a width that allowed for the full support of the

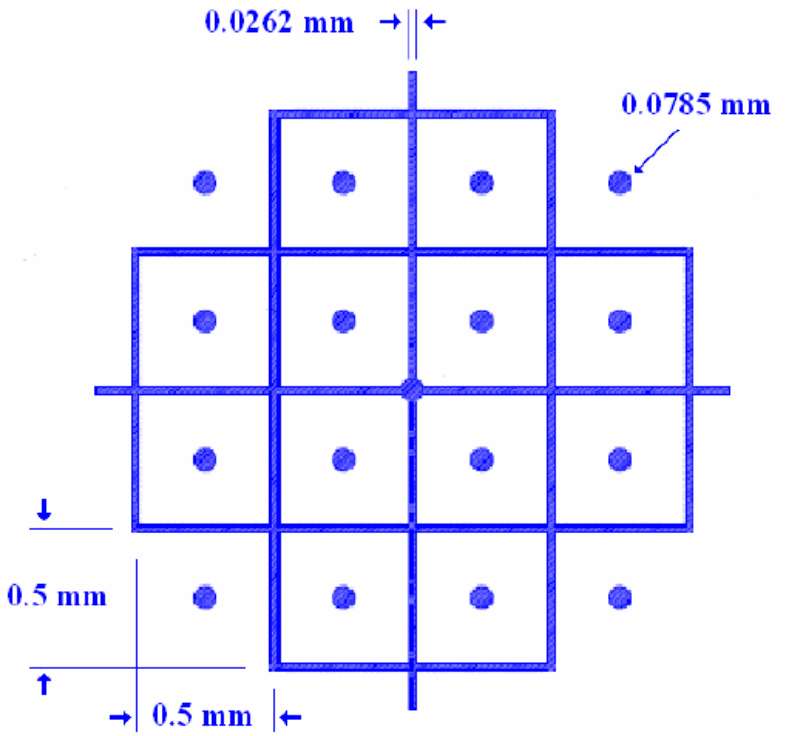


Figure 3.16: Calibration plate's grid and dot pattern with dimensions[33]

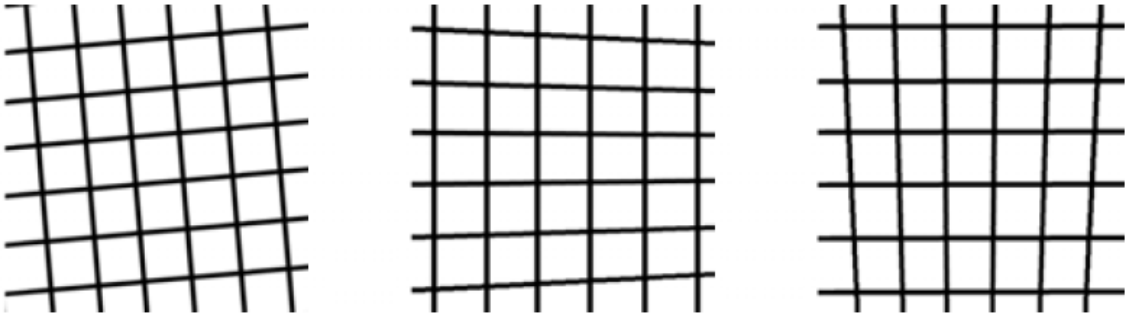


Figure 3.17: Distortions that result from camera misalignment in each of three rotational axes

water-filled prism, a length that spanned the width of the water tunnel test section, and a thickness that matched that of the water tunnel lid. This resulted in a piece of acrylic with dimensions of $11\,3/8\text{in} \times 8\text{in} \times 0.5\text{in}$. The platform, pictured in Figure 3.18 featured 90^o metal brackets for clamping the platform to the edge of the water tunnel at the same height as the water tunnel lid.

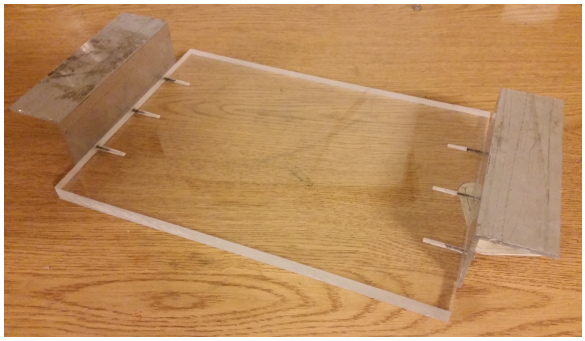


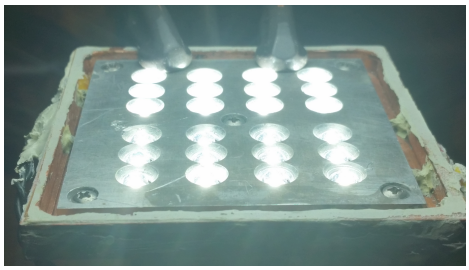
Figure 3.18: Platform to stage the water-filled prism during the calibration process designed to match indices of refraction.

The approach of using a translation stage, calibration target, and matching the indices of refraction was inspired by the work of Tien *et al.* (2014) in which an analogous scheme was used on a micro scale.[26]

3.5.4 Illumination

For the calibration process, continuous white background illumination was used to image the target plate. This illumination scheme was chosen to provide high contrast for the target plate's black grid and dots, as seen in Figure 4.1a. A Light-Emitting Diode (LED) array that consisted of 24 total diodes was used as the illumination source, Figure 3.19a. The array was powered by a Tektronix PS280 Laboratory DC Power Supply at 38V, $\sim$0.5A. A frosted glass diffusion plate was mounted above the

array with sufficient distance to smooth the inherently discrete light pattern produced by the array into an even light distribution, as seen in Figure 3.19b. The LED array and diffusion plate assembly was positioned directly below the water tunnel and aligned with the VOI, as seen in Figure 3.20.



(a) LED Plate

(b) Diffusor plate used to smooth illumination distribution

Figure 3.19: LED and Diffusor Plate

Figure 3.20: Illumination setup for the calibration configuration

3.6 Acquiring Calibration Data

In order to acquire the necessary data for a full calibration of the 3DPTV system, the setup thoroughly described over the course of the previous section was achieved with clamps at appropriate locations to prevent the movement of any calibration parts. The water tunnel was filled such that full contact with the prism platform's underside was achieved. This prevented any free optical surfaces from interfering with the calibration. Next, the calibration rig was positioned near the VOI such that the laser's illumination volume completely encompassed the desired Z-range.

Once the calibration rig was appropriately positioned with the correct Z-range of $\approx$ 4.5mm, the grid lines on the calibration target plate were used to insure that no distortions of the form shown in Figure 3.17 were present. If such distortions were apparent on the pixel-level, the six-axis translation stages were used to remove any distortions. Once the cameras were appropriately aligned, a small reference object was placed on the calibration plate to cast a shadow on each of the CCDs. To insure a relatively equilateral configuration of the cameras, the three linear axes of the translation stages were used to make the shadow appear at equal distances in a combined 3-channel image. This was accomplished at multiple Z-locations throughout the VOI to insure consistent epipolar lines.

Once the system, including the CCDs, calibration rig, and laser, was in perfect alignment, calibration planes could be taken. In order to accomplish this, the Illunis XMV Camera Control Application was used to control the exposure and gain settings of the cameras while Streams 5 was used to control timing, capture, and storage of the calibration images. A new movie was created for each calibration and a new scene was labeled to describe and differentiate the Z-location of each calibration plane. For accuracy purposes, calibration data were taken at 11 Z-locations throughout the VOI, enumerated Table 3.1, which were referenced from the bottom of the VOI. As

previously described, the VOI was located approximately 13mm above the floor of the water tunnel's test section.

Table 3.1: Calibration plane locations relative to the bottom of the VOI.

Plane	Z-Location (μm)
Z_1	0
Z_2	635
Z_3	1270
Z_4	1905
Z_5	2540
Z_6	3175
Z_7	3810
Z_8	4445
Z_9	5715
Z_{10}	6985
Z_{11}	7620

To capture images at each Z-location, the calibration rig was carefully set to the desired height using the linear translation stage. After a sufficient waiting period to allow for any induced motions to dampen beyond detection on the live video feed, 5-10 still images were captured. These images were averaged and the resulting image was used for calibration in order to remove any moving debris that could potentially be moving through the VOI. In addition to these Z-planes, intermediate Z-planes were also captured for uncertainty analysis purposes. After capturing images of all calibration planes, additional translation stages were added to the calibration rig to allow for translation in the X and Y axes as well as the Z. This allowed for the

capture of in-plane position and displacement uncertainty data. The methods used in the post-processing of the calibration images and the building of the calibration RBF are described in a theoretical sense in Chapter 4. A practical tutorial on the use of the 3DPTV software for calibration may be found in Appendix A.

3.7 Data Collection Configuration

The configuration used for data collection was significantly different than the one used for calibration. Laser illumination was used in place of the LED illumination, the calibration rig was removed from the tunnel, the full tunnel lid was used in place of the prism platform, and double-exposure images were taken, precisely timed with the laser pulses, in place of single images. It is also important to note that the entire camera system, as it was fixed on top of a scaffolding structure, did not move whatsoever between calibration and data collection. The full data collection configuration is pictured and labeled in Figure 3.21.

3.7.1 Laser Illumination

While continuous white light was used for background illumination during calibration, such lighting was not feasible for data collection. To provide sharp, live images of a turbulent flow with a high signal-to-noise ratio, pulsed side illumination was used. A New Wave Research Solo PIV 120XT dual-pulsed Neodymium-doped Yttrium Aluminum Garnet (Nd:YAG) laser system was used to provide illumination for all 3DPTV experimentation. The Solo 120XT emits visible light at the 532nm wavelength (green). Maximum energy level was $120\,{}^{\mathrm{mJ}}/_{\mathrm{pulse}}$. Full specifications can be found in Table 3.2.

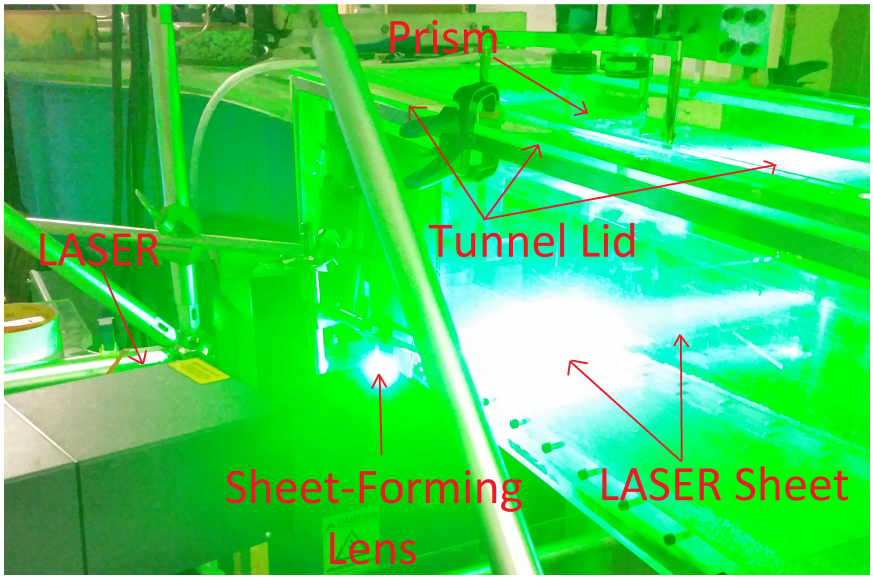


Figure 3.21: Configuration used to collect data.

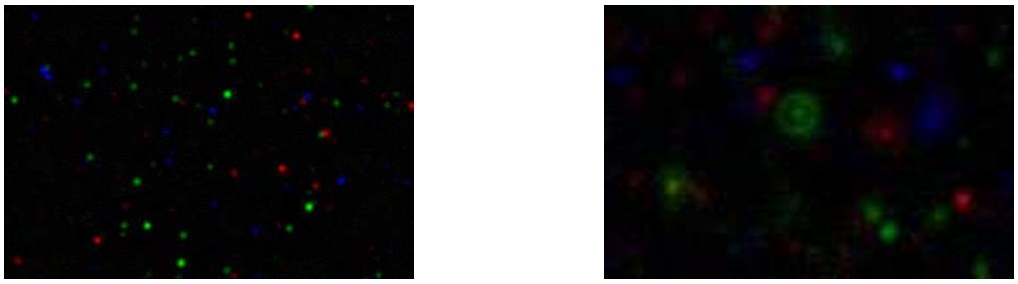
Optics

To achieve complete, controlled illumination of the VOI, a CVI Laser Optics 532nm coated, -12.7mm focal length, plano-concave cylindrical sheet-forming lens was used. The full laser assembly can be seen in Figure 3.23. Extra care was taken to insure accurate laser alignment, such that the resulting laser sheet remained level, prevented reflections off of the tunnel's acrylic floor that caused undesirable halo patterns in particle images. These halo patterns would have prevented accurate peak finding and location mapping of individual tracer particles. A side-by-side comparison of particle images with and without this effect can be seen in Figure 3.22.

The previously-described illumination scheme resulted in a laser sheet that, at its maximum, was approximately 4.5mm thick, the waist of the original beam, and sufficiently wide as to insure an adequate Z-depth of illumination throughout the entire VOI.

Table 3.2: Solo PIV 120XT Laser Specifications[36]

Repetition Rate (Hz)	15
Energy (mJ)	120
Energy Stability (±%)	4
Beam Diameter (mm)	4.5
Pulse Width (ns)	3-5
Divergence (mrad)	<3
Beam Pointing Stability (μrad)	<100
Jitter (±ns)	1



(a) Typical particle image without halo issue

(b) Closeup of particle halo illumination, an undesirable phenomenon

Figure 3.22: Side-by-side comparison of an unaffected particle image and a particle image with halo distortions surrounding particles.

3.7.2 Timing

As in any particle velocimetry application, the precise control of system timing was critical to data accuracy. The architecture of the timing control is described in Figure 3.5 and a basic timing diagram showing the relative timing of the trigger signal

Figure 3.23: Laser assembly with concave cylindrical lens

generator, SG1, the CCD double exposures, and the laser pulses is presented in Figure 3.24.

As seen in the diagram, the master trigger was the data acquisition computer's internal signal generator board, SG1. This signal was a 1.9hz signal at a 50% duty cycle. Relative to the signal, the three CCDs were programmed to begin their TDE function synchronized with the trigger signal's rising edges. This resulted in a camera frame rate of 3.8hz, the design maximum of the CCDs. The time between exposures in TDE mode was determined to be ≈800ns, negligible in comparison to the Δt desired in this study and was therefore not considered in the timing of the laser pulses. Laser 1 was timed to pulse at a time $\frac{1}{2}\Delta t$ prior to the end of Exposure 1, and laser 2 was timed to pulse at a time $\frac{1}{2}\Delta t$ after the end of Exposure 1. To optimize energy per

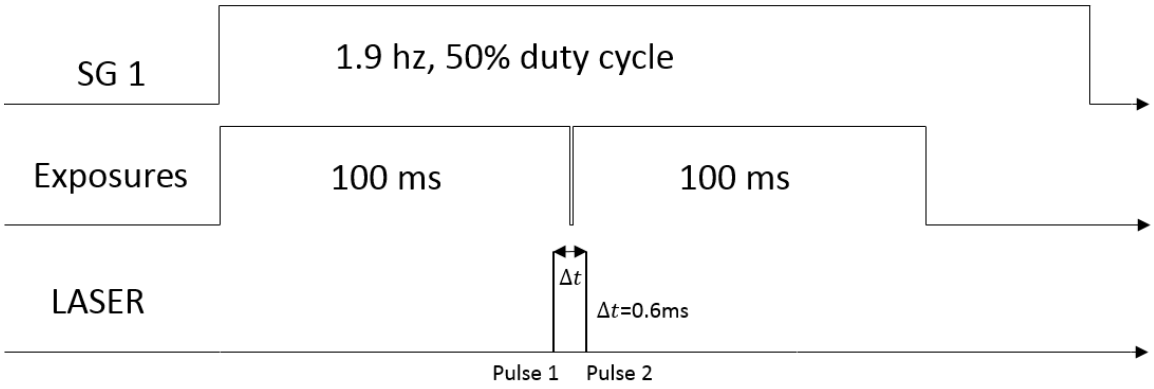


Figure 3.24: Timing diagram for trigger signal, CCD double exposure, and laser pulses. **Note:** Timing is not to scale.

laser pulse, the flash lamp was triggered 190μs prior to the Q-switch. Each laser pulse duration was $3 - 5$ns.

3.7.3 Tracer Particles

Choosing an appropriate particle is a critical step in the development of any PTV application. Candidate particles were assessed based on many criteria including particle diameter, specific gravity, imaged size, imaged intensity, and response time. In addition to the intrinsic importance of these measures, they also determined the non-dimensional quantities of Stokes Number, Reynolds Number, and volume fraction for each particle. These non-dimensional numbers have significant implications for accurately quantifying a flow without significantly altering its nature. A summary of candidate particles and their specifications may be found in Table 3.3. A comparison of the calculated non-dimensional numbers is presented Table 3.4. Finally, the method by which the flow is seeded with these particles must not impact the flow as the flow passes through the VOI. The seeding method and location are also described.

Stokes Number

The Stokes Number of a particle encountering a given obstacle is given by Equation 3.3, where τ_{p} is the particle's response time relative to the carrier flow as defined in Equation 3.4, U_o is freestream velocity, and d_c is the characteristic length of the obstacle encountered by the flow. In the calculation of τ_{p}, Equation 3.4, ρ_{p} is the particle density, d_{p} is the particle diameter, and μ_g is the dynamic viscosity of the carrier fluid. The Stokes Number provides an important incite as to whether or not a particle will accurately follow the streamlines of a flow when the flow encounters the obstacle. If the particle's Stokes Number is high, $St > 1$, the particle's inertial forces will outweigh the carrier fluid's viscous forces and the particle will deviate from the streamline. As St decreases however, the particle's tracking error also decreases. It has been shown that for $St \ll 0.1$, tracking errors become less than 1%.[37]

$$St = \frac{\tau_{\mathrm{p}} U_o}{d_c} \tag{3.3}$$

$$\tau_{\mathrm{p}} = \frac{\rho_{\mathrm{p}} d_{\mathrm{p}}^2}{18 \mu_g} \tag{3.4}$$

While a tracer particle's ability to follow the flow past an obstacle certainly retains its importance in a backward-facing step flow, additional consideration for turbulent structures within the flow were required. Turbulent structures within a flow may also be seen as obstacles around which a tracer must travel. Therefore, the Stokes Number relative to the smallest turbulent structures in the flow, St_η, must be determined as in Equation 3.5,[38] where m_{p} is the particle mass, R is the particle radius, and τ_η is the Kolmogorov timescale as defined by Equation 3.6.[4]

$$St_\eta = \frac{m_{\mathrm{p}}}{6\pi R \mu_g \tau_\eta} = \frac{\tau_{\mathrm{p}}}{\tau_\eta} \tag{3.5}$$

$$\tau_\eta \equiv \left(\frac{\nu}{\varepsilon}\right)^{\frac{1}{2}} \tag{3.6}$$

In Equation 3.6, ν was the kinematic viscosity of the carrier fluid. However, as the tracers were selected prior to the culmination of data analysis, the dissipation rate, ε, of the flow was not a known quantity. Therefore, an estimation using dimensional analysis was obtained by Equation 3.7, as in a similar approach used by Tao *et al.*. (2002).[24]

$$\tau_\eta \approx \left(\frac{\nu l}{u'^3}\right)^{\frac{1}{2}} \tag{3.7}$$

In this instance, the characteristic length scale is equal to the step height, $l = h$, and u' is the root mean square velocity fluctuation. As these values were required for particle selection prior to the results of the study, the value of u' was assumed to be 18% of the inflow velocity, a standard value for a typical free-shear flow.[4] While the separated shear layer of a backward-facing step flow is not a free-shear flow, it was reasonable to assume that the velocity fluctuations were of a similar order of magnitude just beyond the step and reasonably away from any walls. The values for St_η have been determined for all tracer particle candidates and are summarized in Table 3.4.

Reynolds Number

Next, the particle Reynolds Number, Re_p, is determined by Equation 3.8, where all terms are defined using properties of the fluid except for U_p, which is the velocity of the particle used to define the velocity of the particle relative to the fluid flow. For all particles in this study, the condition $St \ll 0.1$ was satisfied meaning the tracing errors were negligible, $< 1\%$. This resulted in trivially small particle Reynolds Numbers and

limited the relevant portion of Figure 3.26 to the extreme left side, along the y-axis.

$$Re_{\mathrm{p}} = \frac{\rho_{\mathrm{f}}|U_{\mathrm{f}} - U_{\mathrm{p}}|d_{\mathrm{p}}}{\mu_g} \tag{3.8}$$

Volume Fraction

A final non-dimensional number critical to the suitability of a tracer particle for a given flow is the volume fraction, Φ_{p}, Equation 3.9, which is the ratio of the volume of particles in a representative volume of the flow to the volume of the fluid in the same volume. In this equation, N is the number of particles in the representative flow volume, V_{p} is the volume of an individual, representative particle, and V is the size of the representative volume such that $V - NV_{\mathrm{p}}$ yields the amount of fluid in the representative volume.

$$\Phi_{\mathrm{p}} = \frac{NV_{\mathrm{p}}}{V - NV_{\mathrm{p}}} \tag{3.9}$$

In the present study, the target seeding density used to calculate Φ_{p} was approximately 33.6 $^{\mathrm{particles}}/_{\mathrm{mm}^3}$ while the individual particle volume, V_{p} was calculated using the average particle diameters presented in Table 3.3. The resulting Volume Fraction for each particle may be found in Table 3.4. These resulting volume fractions were analyzed using Figures 3.25 and 3.26, along with Stokes Number and particle Reynolds Number, in order to determine what level of multiphase coupling was present. The summarized results of this analysis are also presented in Table 3.4.

Particle Selection

The tables featured in this section include particle information such as the type of particle tested, the manufacturer, material number, average diameter, specific gravity, average imaged diameter, Stokes Number, volume fraction, and coupling type of the

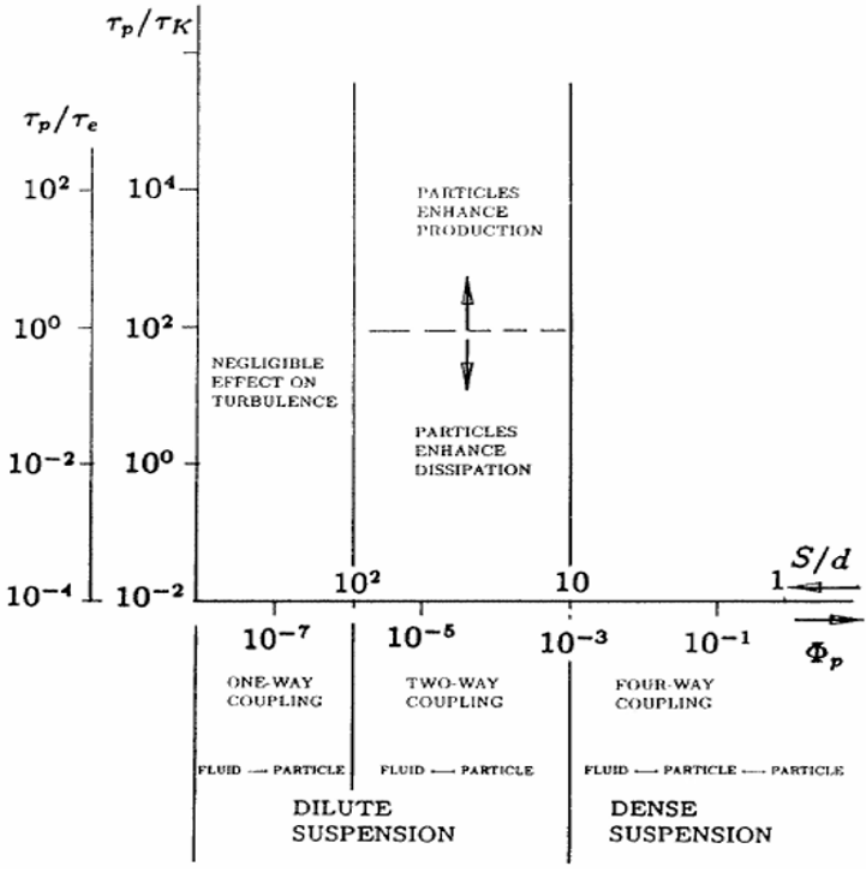


Figure 3.25: A map for particle-turbulence modulation. Ratio of particle time constant to turbulence time constant, τ_p/τ_e, is plotted versus volume fraction, Φ_p.[39,40]

resulting multi-phase flow. Ultimately, the particle selection was made to achieve the maximum number of particles per image while minimizing the volume fraction, Φ_p, in order to eliminate two-way coupling. To this end, the imaged particle size was an important measure of merit as this value determined how many imaged particles could theoretically be captured by the resolution of a CCD used in the study. A combination of this value and seeding characteristics led to the selection of TiO_2

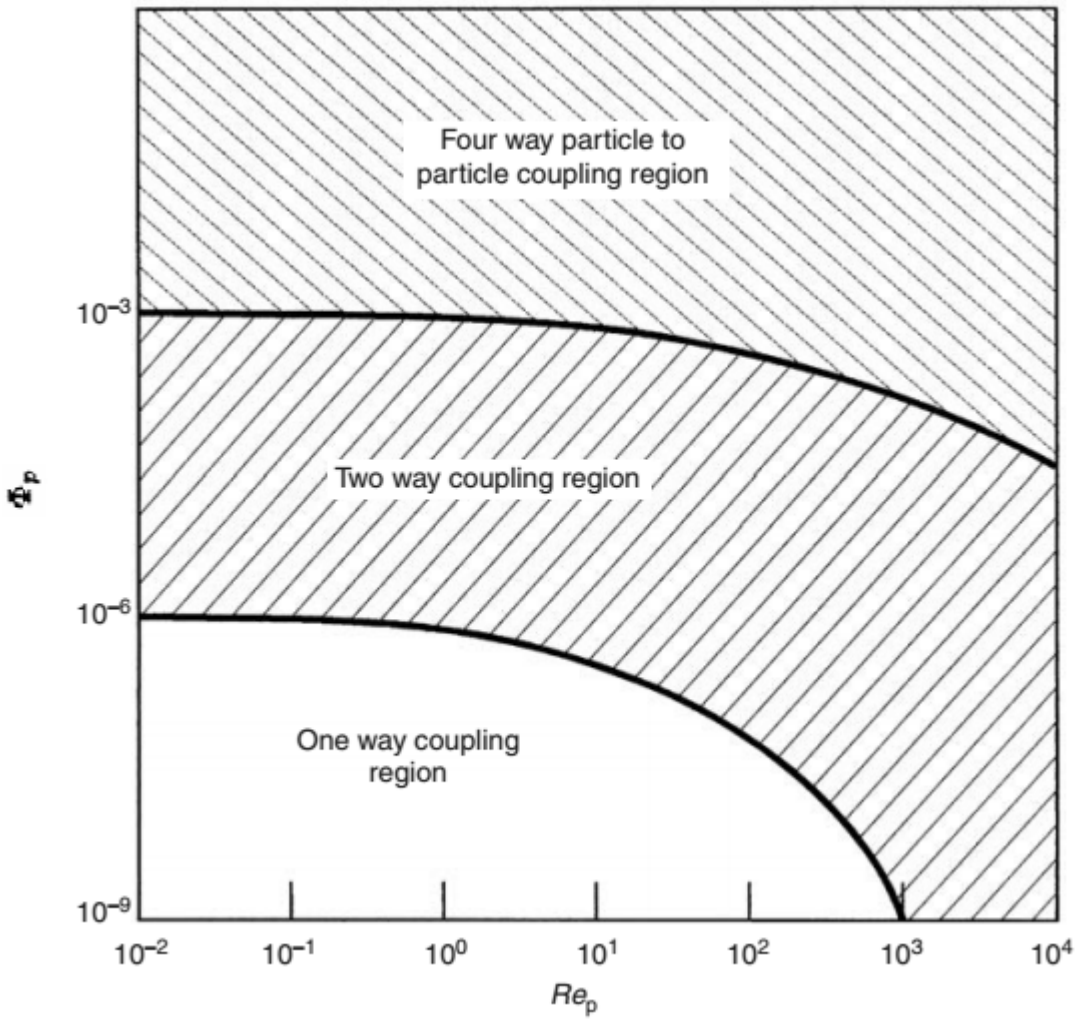


Figure 3.26: Map of multiphase coupling regions. Volume fraction, Φ_{p}, is plotted versus particle Reynolds number, Re_{p}.[41]

Powder, Rutile from Sigma Aldrich, as the primary particle used to seed the flow in this study. As the Silver-Coated Thin Glass Flakes (5μm) and Silver-Coated Glass Spheres (2μm) were new arrivals at the end of the study, they were only tested in a limited capacity. As their performance was promising, it is recommended that they are used in place of TiO_2 powder in future studies.

Table 3.3: Candidate tracer particles considered in the study

Particle	Manufacturer	Material No.	Avg Diameter	Avg SG
Silver-Coated Hollow Glass	Potters Ind. Inc.	SH400S20	13 μm	1.6
Hollow Glass Spheres	Potters Ind. Inc.	110P8	10 μm	1.10
Silver-Coated Thin Glass Flakes*	Potters Ind. Inc.	SG05TF40	5 μm	$\sim$3.5
TiO_2 Powder, Rutile	Sigma Aldrich	224227	$<$5 μm	4.17
Ceramic Microspheres	3M	W-210	3 μm	2.4
Silver-Coated Glass Spheres*	Potters Ind. Inc.	SG02S40	2 μm	$\sim$3.5
TiO_2 Nanopowder, Rutile	US Nano	US3548	500 nm	4.23

*Samples courtesy of Potters Industries, Inc.

Seeding Method

The seeding method used in this study was designed to deliver a concentrated stream of particles, with a target concentration of $\approx$33.6 $^{\text{particles}}/_{\text{mm}^3}$, while not adding turbulence to the flow. To this end the particle seeding wand was placed far upstream of

Table 3.4: Performance comparison of tracer particles in the study

Particle	Avg Imaged Diameter (px)	St_η (-)	Φ_p (-)	Coupling
Silver-Coated Hollow Glass	~ 10	4.69×10^{-4}	3.87×10^{-5}	Two-Way (Enhanced Dissipation)
Hollow Glass Spheres	< 1	1.91×10^{-4}	1.76×10^{-5}	Two-Way (Enhanced Dissipation)
Silver-Coated Thin Glass Flakes	$6 - 7$	$< 1.52 \times 10^{-4}$	$< 2.20 \times 10^{-6}$	One-Way
TiO_2 Powder, Rutile	$7 - 8$	$< 1.81 \times 10^{-4}$	$< 2.20 \times 10^{-6}$	One-Way
Ceramic Microspheres	< 1	3.74×10^{-5}	4.75×10^{-7}	One-Way
Silver-Coated Glass Spheres	$6 - 7$	2.43×10^{-5}	1.41×10^{-7}	One-Way
TiO_2 Nanopowder, Rutile	~ 7	1.83×10^{-6}	2.20×10^{-9}	One-Way

the water tunnel test section and backward-facing step. The wand was positioned at the center of the stilling chamber section of the water tunnel approximately $33.78h$ upstream of the step. Additionally, the wand was stationed at the perforated plate and upstream of the flow straighteners. This allowed for the breakup and attenuation

of any vortex structures added to the flow by the seeding wand prior to reaching the test section. A top-view diagram of the water tunnel with wand location is illustrated in Figure 3.27. As seen in the illustration, the stilling chamber of the water tunnel contained perforated plate and flow straightener sections. The perforated plates were stainless steel sections with 60% porosity. The flow straighteners consisted of an array of plastic honeycomb tubular cells.

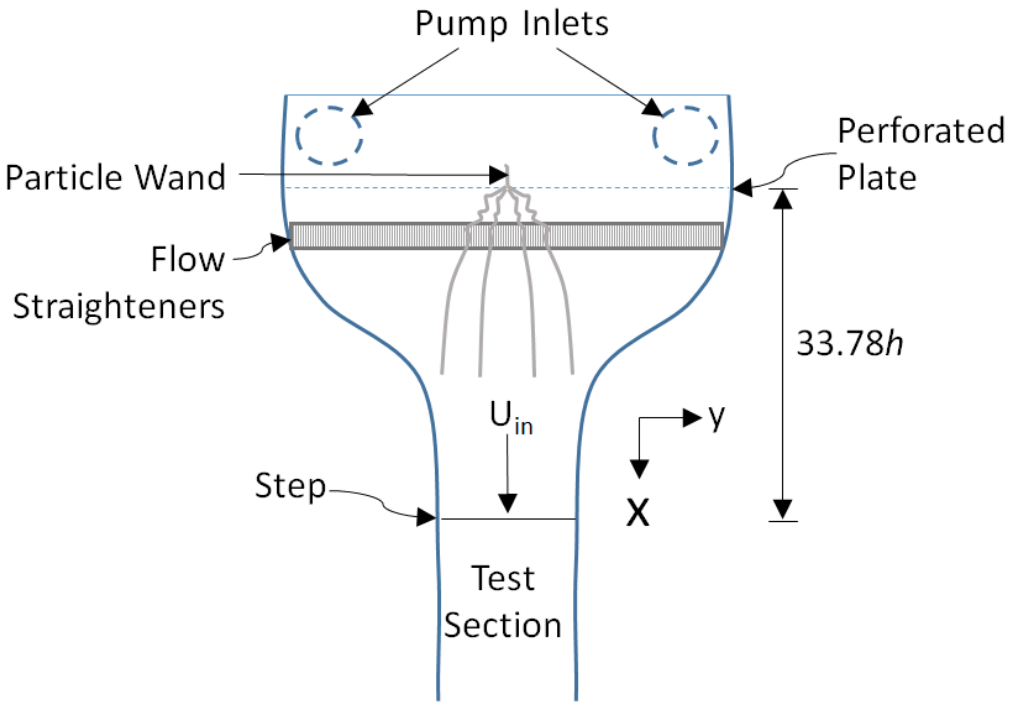


Figure 3.27: A top view drawn diagram of the particle seeding wand location relative to the backward-facing step and the tunnel test section.

3.8 Acquiring Experimental Data

To collect the data necessary for the *a priori* testing of LES SGS models, the setup described throughout the previous section was carefully achieved. CCD exposure timing was precisely matched in an iterative fashion using the Illunis XMV Camera Control Software. Streams 5 was used to precisely control the CCDs and laser pulses using the timing settings shown in Figure 3.24. Timing settings were repeatedly refined to insure that each frame in the TDE had adequate illumination by a laser pulse. The value for Δt was empirically determined such that the maximum displacements of pixels in any image pair were $\approx$ 10px. In this study, Δt was $\approx$ 0.6ms for an inflow velocity of 22 $^{\text{cm}}/_{\text{s}}$.

Once the appropriate timing settings were achieved, seeding particles were prepared. For $< 5\mu$m diameter Rutile TiO_2, a solution of 0.75-1.25g of TiO_2 powder with 0.25L water was prepared in a seeding bottle. The seeding bottle was attached to the seeding wand and placed above the water tunnel at a height of $\approx$ 65cm above the seeding location in order to allow for the gravity siphoning of particles into the water tunnel.

Once seeding began, the CCDs were placed into live mode with the laser running. Once the desired seeding concentration was achieved by visual assessment, image pairs were captured for a typical total of 2000 images, which would result in 1000 3D3C velocity vector fields. Given the mean frame rate of the CCDs used in this study, this resulted in a data rate of 1.9 $^{\text{frames}}/_{\text{s}}$. The Δt and water tunnel pump RPM setting for each data set were recorded.

Chapter 4

DATA ANALYSIS

"I am an old man now, and when I die and go to heaven there are two matters on which I hope for enlightenment. One is quantum electrodynamics, and the other is the turbulent motion of fluids. And about the former I am rather optimistic."

Horace Lamb

The data analysis portion of this study was an involved process that bridged the gap between particle images and LES SGS model studies. This chapter describes the techniques used in the reduction of experimental data, including calibration and flow data. Steps described in this chapter include image de-noising, removal of lines from calibration images, contrast adjustment, blob finding, peak location, Radial Basis Function (RBF) building, triplet matching, particle tracking, outlier removal, unstructured interpolation, LES filtering, and SGS model application. A full practical tutorial and guide to using the 3DPTV Software Suite used in this study is in Appendix A.

4.1 Calibration Data Processing

Once calibration data were acquired, as described in the previous chapter, an RBF interpolation was ultimately built. First, however, images were processed, peaks were located, and the RBF interpolation was built based on a mapping of the peak locations. As many of the parts of calibration processing were the same as data processing, only the steps unique to calibration data processing are discussed in this

section. Steps that were similar to those used in data reduction are described in Section 4.2.

As calibration planes were static images, any moving objects were removed from view via image averaging to form a single calibration image for each plane. Next, a de-noising step was accomplished as it was in flow data reduction, Section 4.2.1.

4.1.1 Target Plate Line Removal

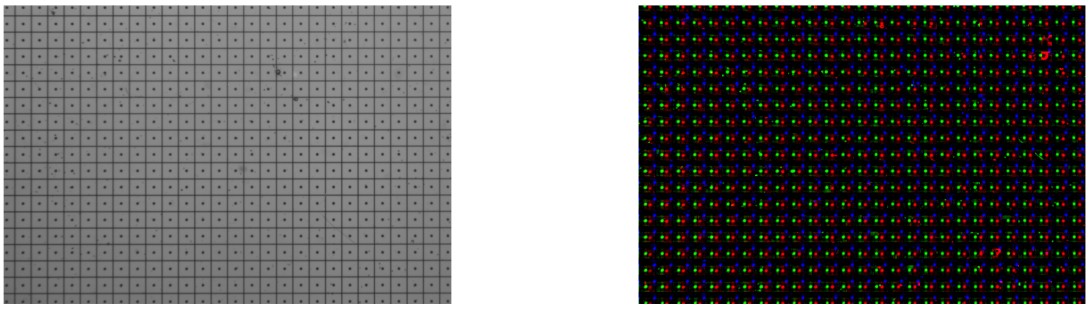
After the calibration images were de-noised, lines from the calibration target's grid and dot pattern remained, as seen in Figure 4.1a. These lines proved extremely problematic for the peak-finding software as they were 1-D Gaussian distributions in each image direction. To fix this issue, a computer vision technique was developed utilizing Principal Component Analysis (PCA) via a SVD.

To accomplish this, the image complement was calculated and adaptive histogram adjustment was accomplished to smooth the histogram over windowed areas of the image to fix spatial lighting variations that may skew statistics. Then, vertical lines of pixels from the original calibration image, $\mathbf{Im}$, and its transpose, $\mathbf{Im}^T$, were decomposed in MATLAB® using the economy SVD, Equation 4.1.

$$\mathbf{Im} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{4.1}$$

Next, the orthogonal unit vectors, right singular vectors, that made up $\mathbf{V}$ were extracted for the remainder of the process. The first component of all unit vectors of $\mathbf{V}$, Principal Component 1 (PC1) or the rows of the $\mathbf{V}$-matrix, were extracted and ranked. It was empirically determined that the lowest of the PC1 values corresponded to lines of resolution composing the grid lines of the images. The lowest 300-500 of these horizontal and vertical lines of resolution in each image channel were removed, set to 0 intensity. Contrast issues in the resulting images were fixed using

tophat spatial filtering. The final product of the de-noising, line removal, and contrast adjustment processes is pictured in Figure 4.1b.



(a) Sample region of target plate in a raw calibration image from one of three cameras

(b) Same image as Figure 4.1a after post-processing, line- removal, and 3-camera combination

Figure 4.1: Side-by-side comparison of raw and post-processed sample calibration images

4.1.2 Calibration RBF Development

After blob finding and peak location, described in Sections 4.2.2 and 4.2.2, respectively, were completed, the particle locations in image space with known physical Z locations, $P(x, y, Z)$, from each image channel were used to develop the calibration RBF by which particles were located. The process of building the 3-D calibration is described here as it was presented in the work of Tien (2013).[25]

The theory of the calibration process is pictured in Figure 4.2 using only two image planes as an introductory illustration as the system used in this study contained 3 CCDs. Considering the two-perspective approach, the peak locations resulting from the previously-described process were related to the projections in other image planes

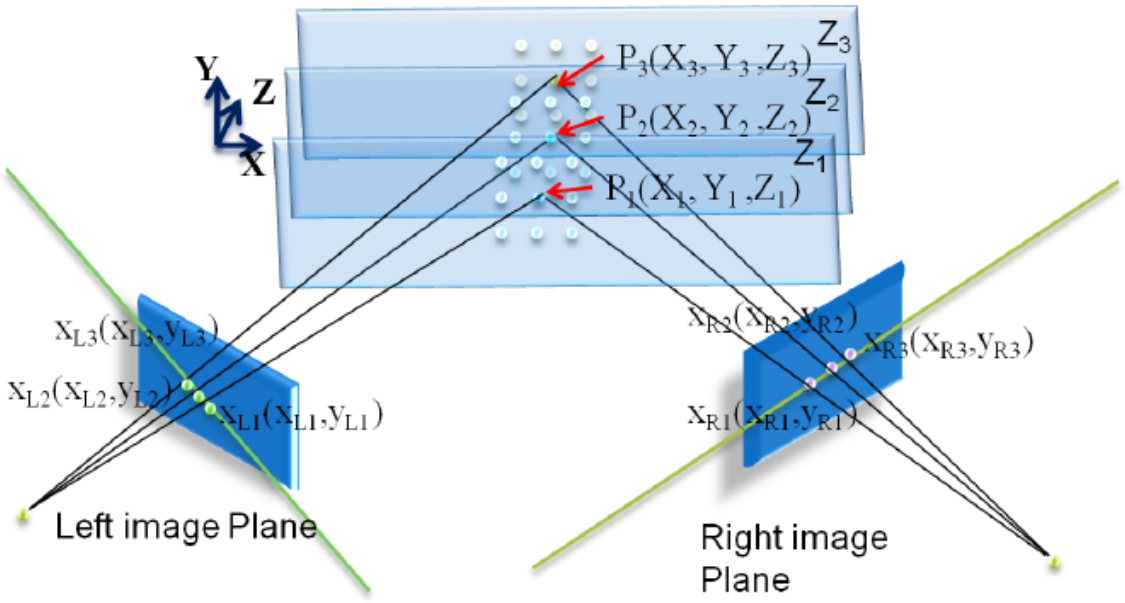


Figure 4.2: Epipolar line calibration approach used to build an RBF mapping of x, y image coordinates to X, Y, Z physical coordinates using calibration data, a simplified 2-perspective approach.[26]

by the known Z-locations using functions presented in Equation 4.2 and Equation 4.3. In these equations, X, Y, and Z correspond to the peak physical coordinates as known from the calibration location and geometry. The coordinates x_L, y_L, x_R, and y_R correspond to image coordinates from different image planes of Figure 4.2.

$$\begin{aligned} x_R &= f_{x,L\Rightarrow R}(x_L, y_L, Z) \quad y_R = f_{y,L\Rightarrow R}(x_L, y_L, Z) \\ x_L &= f_{x,R\Rightarrow L}(x_R, y_R, Z) \quad y_L = f_{y,R\Rightarrow L}(x_R, y_R, Z) \end{aligned} \tag{4.2}$$

$$\begin{aligned} X &= F_{X,L\Rightarrow XYZ}(x_L, y_L, Z) \quad Y = F_{Y,L\Rightarrow XYZ}(x_L, y_L, Z) \\ X &= F_{X,R\Rightarrow XYZ}(x_R, y_R, Z) \quad Y = F_{Y,R\Rightarrow XYZ}(x_R, y_R, Z) \end{aligned} \tag{4.3}$$

As these equations describe a 2-perspective system, the process was repeated to each of the 3 CCDs, top left to top right, top right to bottom, and bottom to top left, corresponding to each of the rgb image channels, instead of simply left to right and right to left. This approach helps limit incorrect particle mapping caused by the non-uniqueness of particles that has been well-documented in 3-D systems with fewer perspectives.[25] The functions were modeled in MATLAB® using an RBF interpolation routine.

When the calibration was applied in the analysis of flow data, the RBF mapping generated in this step used $(x_r, y_r), (x_g, y_g)$, and (x_b, y_b) image coordinates from each of the image channels in order to map particle locations to physical space, (X, Y, Z).

4.2 Experimental Data Analysis

The theory behind each step of the data analysis process is described over the course of this section. Any step that was also used in the processing of the calibration data is also described here in the context of experimental data. A practical guide to the use of data reduction software may be found in Appendix A.

4.2.1 Image Processing

All relevant image processing steps are discussed in the following subsections. It is important to note that no color separation was required in this study, as it was in the work of Tien, due to the use of three independent gray scale CCDs.

Background Subtraction

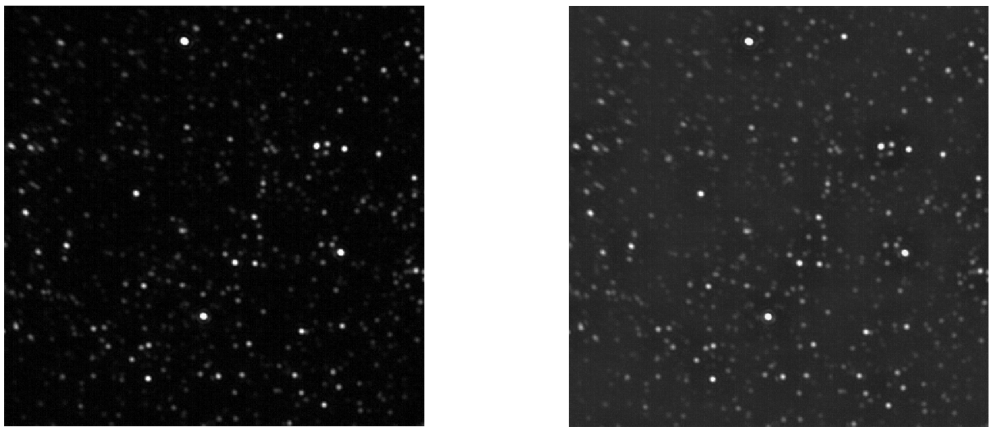
Reflections caused by the illumination and particle accumulation in the VOI often manifested themselves as constant background intensity offsets in particle images. Instead of using a time-averaged mean from the series of particle images, the minimum

pixel values from the time series were taken as the background image. This approach was noted to have improved performance over the time-averaged approach.[25] This background image was then subtracted from each frame in the entire particle image series. This meant that any constant image offsets caused by background particles or illumination were removed from the image series.

Pre-Processing

Even with background offsets removed, there were still spatial variations in intensity throughout the particle images related to different types of random noise. The intensity variations across the image were likely a result of the laser beam's uneven intensity distribution, a common attribute of high-powered laser systems. Additionally, noise associated with imaging sensors, as is typical in any digital imaging application, was an issue. Both of these manifestations of noise were problematic for blob and peak finding in the 3DPTV algorithm and were removed.

The approached used to remove both of these noise types was spatial band-pass filtering. Sensor noise was typically a fine-scale, high-frequency random variation that could be removed by high-pass filtering. Illumination variation was typically over a longer spatial scale, low-frequency random noise that was eliminated by low-pass filtering. To effectively remove both types of noise simultaneously, band-pass filtering was applied with settings adjustable to the specific application. A before and after example of pre-processing is shown in Figure 4.3.



(a) Sample region of a particle image after background subtraction and before pre-processing/de-noising step

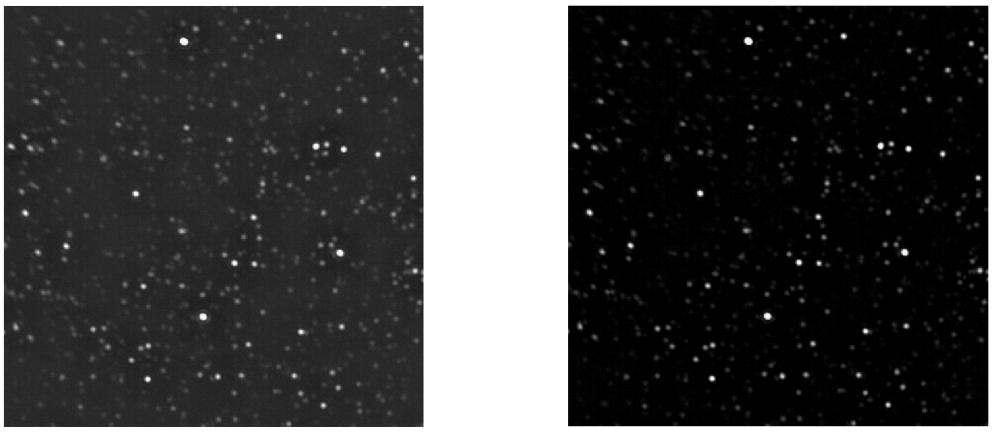
(b) Same image region after pre-processing/de-noising step

Figure 4.3: Side-by-side particle image comparison of before and after pre-processing step

Post-Processing

As the image contrast was typically lowered throughout the course of the previously-described processes, each image underwent a contrast adjustment. The contrast adjustment was a stretching and normalization of intensity ranges to insure all frames had the same resulting intensity range across each image. This allowed for more consistent results in the global threshold that would be applied during the blob finding step. This process was also important as it allowed for consistent intensity across all image channels because levels were typically not consistent between images from

different CCDs.



(a) Sample region of a particle image after pre-processing/de-noising and before post- processing/contrast adjustment step

(b) Same image region after post-processing/contrast adjustment step

Figure 4.4: Side-by-side particle image comparison of before and after post-processing step

4.2.2 Peak Finding

The ability to assess individual particle's peak locations from particle images is a key step in any PTV process. This was accomplished in a multi-step approach. Starting with global image thresholding, blobs, clusters of image intensity values, were isolated from the background of the image. In the step that followed, the Cascade Cross-Correlation Method, CCM, was accomplished on the isolated blobs from the previous

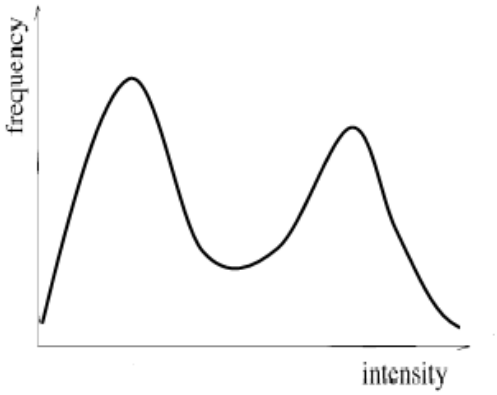
step. The resulting data were in the form of sub-pixel accurate peak values in image x-y coordinates.

Blob Finding

Prior to the determination of individual peak locations, blobs, Gaussian clusters of relatively bright pixels corresponding to particles, were isolated from the particle image background. This binarization is typically accomplished using an appropriate global threshold value of pixel intensity. The faulty determination of the blob threshold intensity value, caused by lighting variations, background noise, and insufficient focus, has been cited as a leading cause of particle loss in the 3DPTV process by Takehara.[42] Thus, optimal thresholding is a critical element of a procedure that is the focus of continuing study.

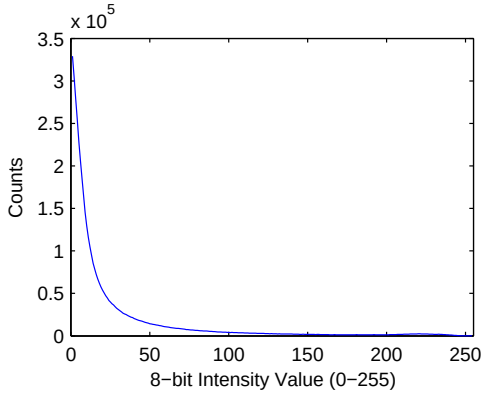
Many gray-level histogram-based thresholding methods have been developed to this end. Historically, histogram-based techniques have largely relied upon the classification of bimodal and multimodal distributions of image histograms as reviewed by Sahoo[44] and Glasbey.[45] This included Otsu's method (1979),[46] which is a traditional gray-level histogram-based image thresholding technique that has been shown as an effective technique in particle isolation in the 3DPTV process in many past studies.[47] Examples of these histogram types may be seen in Figure 4.5. Figure 4.5a is an example of the bimodal distribution assumed by Otsu's method. Figure 4.5b is an example of the unimodal distribution assumed by Rosin's method. Figure 4.5c is an empirical example of a unimodal intensity distribution from the particle images generated by this study.

Initially, Otsu's method of global gray-level thresholding was applied to the data in this study. The effectiveness of this blob isolation method was poor however, typically resulting in only $\approx$ 50% of peaks located in a given particle image. An iterative



(a) Theoretical Bimodal Distrubition of an Image Histogram

(b) Theoretical Unimodal Distrubition of an Image Histogram[43]



(c) Empirical Unimodal Distribution of a Particle Image Histogram

Figure 4.5: Examples of Various Image Histogram Distribution Types

technique was initially developed to delete processed blobs, allowing for dimmer blobs to be found in the next pass. However, after further inspection, the histogram of the particle images in this study were of the distribution shown in Figure 4.5c. Therefore, the assumption that allowed for the use of Otsu's method was not valid for the present study.

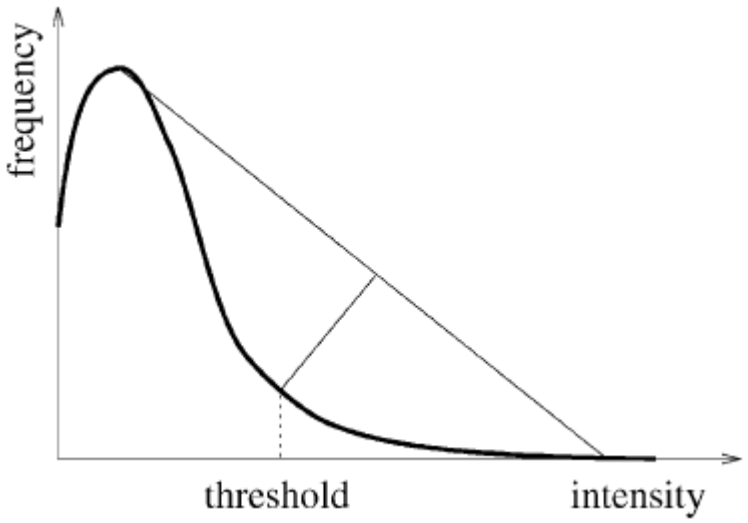
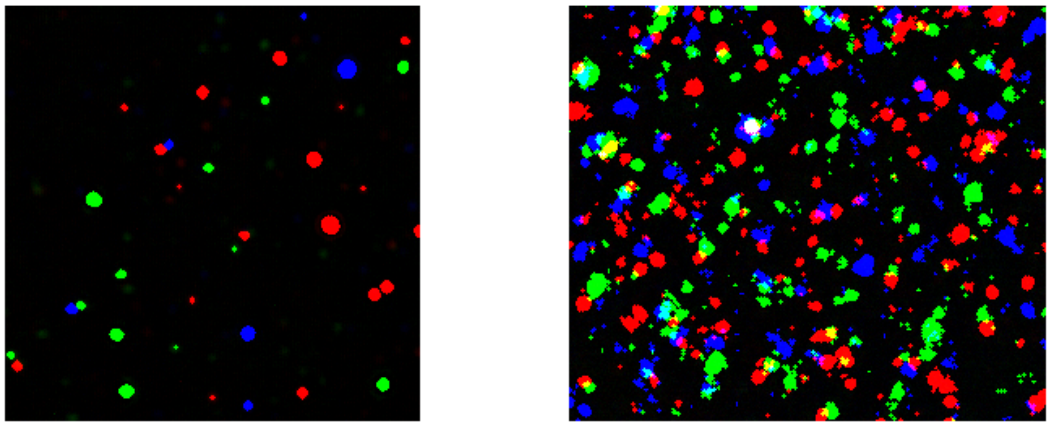


Figure 4.6: An illustration of the Rosin Thresholding process[43]

Instead, Rosin's method, illustrated in Figure 4.6, was adopted leading to the appropriate isolation of blobs. Rosin's method works by choosing an appropriate global threshold such that error is minimized when separating a smaller, subdued Gaussian distribution contained within the main unimodal distribution. A comparison of the performance of Otsu's Method versus the performance of Rosin's Method in finding blobs in a particle image is shown in Figure 4.7. The final result of the blob finding step is a structured array of binarized image sub-regions mapping the origin and pixels associated with each blob found. This blob array is typically immediately used by the Peak Finding step to find peak locations within each isolated blob.

Peak Finding

Tien *et al.* combined the CCM algorithm of Angarita-Jaimes*et al.* (2009)[29] with the Gaussian surface fit approach of Ponchaut (2005)[48] to develop the Modified CCM algorithm used in this study. The CCM algorithm used Gaussian masks of varying



(a) Sample region of a particle image with blobs found by Otsu's method and identified by regions of saturated pixels.

(b) Sample region of a particle image with blobs found by Rosin's method and identified by regions of saturated pixels

Figure 4.7: Side-by-side particle image comparison of Otsu's Method and Rosin's Method. Note the relatively dim particle images that were not found by Otsu's method.

radii and eroded particle images to ensure precise location of peaks over a range of sizes, including smaller peaks that normally have a lesser correlation coefficient.[25]

The multiple Gaussian surface fitting approach taken by Ponchaut allowed for the identification of heavily-overlapped particles, a critical feature for this study considering the high particle densities involved. The resulting combined technique found heavily-overlapped particles over a large size range. This approach has been shown to be very effective, up to nearly 60% overlap ratios of particles. For more information regarding the complete Modified CCM and its performance, see Tien (2013).[25] The final result of the peak finding step is a full list of image x and y sub-pixel accurate

coordinates for all peaks found in each particle image.

4.2.3 Triplet Matching and Particle Location

The Triplet Matching process used the resulting image coordinate sub-pixel accurate peak locations of the previous step from all three imaging planes. Using the epipolar geometry characterized by the RBF interpolation developed from the calibration process, triplets were found by searching for matching particles from each of the three imaging planes along epipolar lines. An illustration of this triangulation process is shown in Figure 4.8.[48]

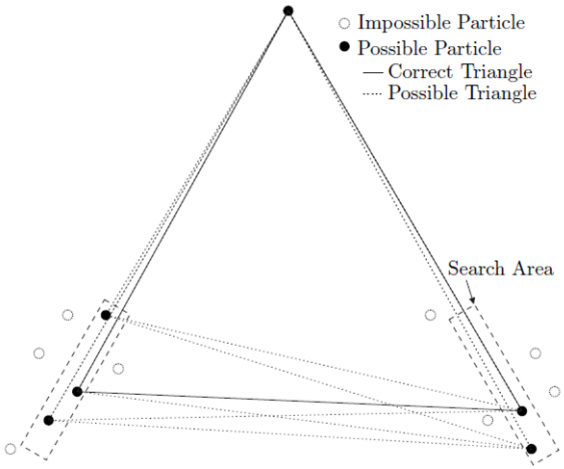


Figure 4.8: An illustration of the epipolar line search method for triplet matching by way of triangulation[48]

Once the peak locations were matched between image channels, the image to image mapping that was developed in the calibration process was used to determine the Z location most likely to produce the triplet pattern. Once the image x and y along with the physical Z locations were known, the image to physical mapping developed in the calibration process was used to map the x and y image coordinates to physical

coordinates, completing the 3-dimensional location of the particle. The result of this step was a list of 3-dimensional particle locations in X, Y, and Z physical coordinates for each image. For further details on the Triplet Matching and Particle Location process by way of the epipolar line search method and interpolation mapping, refer to Tien (2013).[25]

4.2.4 Particle Tracking

After particle locations in image space, $\mathbf{P}_{rgb}(\mathbf{x}, \mathbf{y}, t)$, have been mapped to 3-D physical space, $\mathbf{P}_i(\mathbf{X}, \mathbf{Y}, \mathbf{Z}, t)$ for all potential particles, i, and for each of rgb particles taken at time t_1 and t_2, the velocity field was to be determined. As all physical locations were known, tracking in the form of matching these locations between time steps t_1 and t_2 were accomplished and outliers were removed. All tracking in this 3DPTV application was accomplished using a Computer Vision method.[30] Outlier filtering was accomplished using a 3-D, unstructured Universal Outlier Detection algorithm developed by past members of the research group.[49]

3DPTV by Computer Vision

Automated matching of identified image features between distinct images has been a long-standing goal of the computer vision field. In many applications, distinct features are easily matched between images. However, when image features become non-unique, such as particle images, the matching process becomes more complex as simple matching is no longer a possibility.[25] Ignoring the image feature and using coordinates along with heuristic principles known as the "Principle of Proximity" and the "Principle of Exclusion," Scott and Longuet-Higgins have concluded that computers are able to accomplish many of the same associations that humans can form instinctively.[30] An example of this type of association is shown in Figure 4.9.[30]

The "Principle of Proximity" is the assumption that the closer the two features, the greater the likelihood that they are associated. This principle is analogous to the more primitive nearest-neighbor method of particle tracking. The "Principle of Exclusion" is the rule that dictates features should be matched one-to-one between frames. These principles were incorporated into the approach of Scott and Longuet-Higgins by first developing a rating of proximity in the matrix, $\mathbf{G}$, as shown in Equation 4.4.[25,30,50]

$$G_{ij} = e^{-r_{ij}^2/2\sigma^2} \tag{4.4}$$

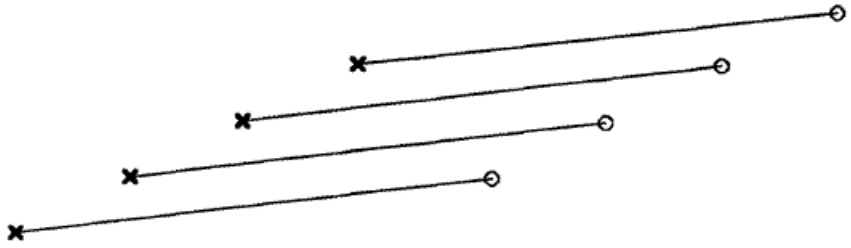


Figure 4.9: Example of Vision Association[30]

In this application, r_{ij} represents the radius between particles i at t_1 and particles j at t_2 and σ is a representative distance for the displacement of the features. From this proximity matrix, a pairing matrix, $\mathbf{P}$, must be developed such that the Frobenius inner product, $P_{ij}G_{ij} = tr(\mathbf{P}^T\mathbf{G})$, is maximized. This pairing matrix, $\mathbf{P}$, is generated by the Singular Value Decomposition, SVD, of the proximity matrix, $\mathbf{G}$, as shown in Equation 4.5.

$$\mathbf{G} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \tag{4.5}$$

Following the SVD, the diagonal $\mathbf{\Sigma}$ matrix is replaced by the identity matrix in order to form the pairing matrix, $\mathbf{P}$, as shown in Equation 4.6. Within $\mathbf{P}$, a

given element, P_{ij}, represents the pairing likelihood between element i from the t_1 image and element j from the t_2 image. Because $\mathbf{G}$ is a Gaussian weighting of the distance between features i and j, $\mathbf{P}$ may be seen as a least-squares solution of distance mapping between image features. By nature of the SVD, $\mathbf{U}$ and $\mathbf{V}^T$ are orthogonal as well as the resulting $\mathbf{P}$ matrix. This means that there is only one maximum per row and column. Thus, the least-squares distance mapping provides proximity, and the orthogonality of rows and columns provides exclusion. Therefore, both principles of vision association are satisfied by the SVD.[25,30]

$$\mathbf{P} = \mathbf{UIV^T} \tag{4.6}$$

Thus, any element of P_{ij} that is the maximum of both the row and column to which it belongs is considered a match between feature i from t_1 and feature j from t_2. This method is a simple implementation and only presents one issue for this application. The value for σ in Equation 4.4 must be defined as an average displacement in the flow. In the past, σ has been chosen by way of PIV results from the same flow.[25] When applying this to turbulent flows, however, this presents a significant issue as each realization of the flow has uniquely chaotic structures that may not be resolved by way of PIV. In lieu of this approach, the tracking algorithm has been altered by the author to accomplish an adaptive selection for the value of σ where each successive volume element, or voxel, of the flow uses the average displacement from the previous voxel as the representative σ value used for normalization in the Gaussian distance weighting, G_{ij}. The assumption is that, with sufficiently small voxels, the flow characteristics will be consistent enough that the mean displacement will be similar in successive voxels. A further discussion of computational voxels may be found in Section 4.2.4. The development of the iterative σ determination approach is presented in Section 4.2.4.

The other known issue regarding this method is a maximum angle of rotation of features between successive frames. This was found by Luo and Hancock (2002) to be approximately 20^o.[51] It was noted by Tien, however, that if PTV is correctly applied to even vortical flows, this should not be a source of uncertainty. The approach has evolved and was implemented in 2-D[50] and hybridized with PIV results.[25] Then, an outlier removal scheme, described in Section 4.2.4, was implemented to increase accuracy and robustness.[49] The algorithm was adapted for use with 3-D position data, but maintained its original program flow, presented in Figure 4.10.[25]

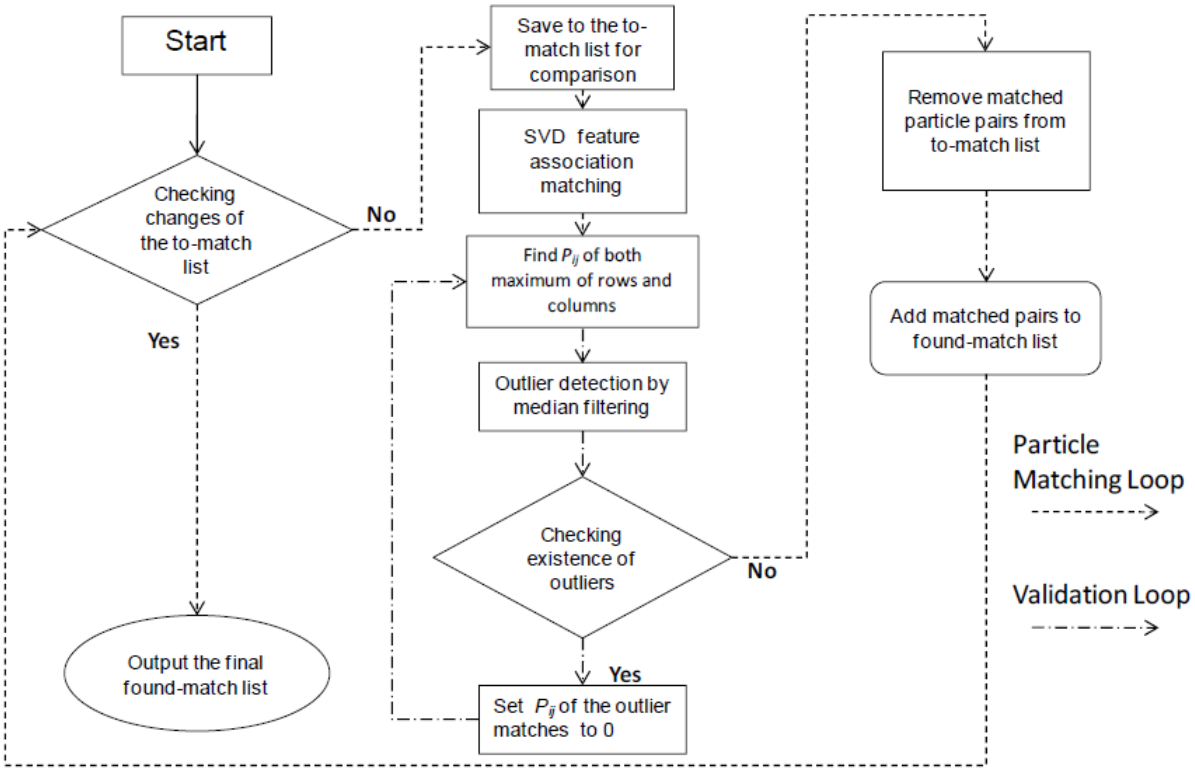


Figure 4.10: 3DPTV Tracking Program Flow[50]

While the application of the algorithm to 3DPTV was well-assessed by Tien,

who showed that the algorithm maintained over 90% reliability for even rotational flows with high displacements. There has not been, however, any assessment for applications to chaotic turbulent flows, to the author's knowledge. In this study, the application to turbulent flows was analyzed and is presented in Section 4.2.4.

Computational Voxels

The inputs of particle physical locations for the 3DPTV step were mapped into a 512× 512 computational domain and compartmentalized into volume elements, previously referred to as voxels, with a 50% overlap in each dimension. Re-mapping was intended to provided a consistent computational domain in order to normalize the physical dimensions of any potential application.

The distribution of the particle locations into individual voxels was implemented to increase computational efficiency. The SVD operator, described in Section 4.2.4, is a very computationally complex algorithm and there is a highly non-linear increase in computation time versus increase in **G** matrix size. Therefore, SVD accomplished on many smaller matrices is much more efficient than SVD accomplished on a single larger matrix. Finally, a 50% overlap was added to prevent voxel partitions from splitting up potential matches between voxels. Thus, accuracy was maintained while computational efficiency was greatly increased.[50]

In this study, the computational re-mapping of the experimental VOI was refined to insure the re-mapping only occurred over the physical range that included viable data. This prevented empty voxels and improved data distribution among the SVD operations. The computational voxel side lengths used in this study were 32×32 or 64×64 depending on the vector density of the data set.

Iterative σ Definition

As discussed by Tien (2013),[25] the vision-based feature association work originally accomplished by Scott and Longuet-Higgins (1991)[30] uses characteristic displacement, σ in Equation 4.4, to help build the Gaussian distribution of all possible displacements of particles located within each voxel. Scott and Longuet-Higgins also noted that when an overestimation of expected displacements was used for the σ value, better results were achieved. In the work of Tien (2013), interpolated PIV results were used to develop a guiding field of σ values. For turbulent flows, however, this approach would not be viable due to the fluctuating, unsteady nature of the flow through the experimental VOI.

Thus, a new approach was developed for the flow in this study. Instead of a pre-defined field of σ values, σ was initialized as the expected mean flow velocity magnitude, as an order of magnitude approximation. Then, after the determination of the velocity field within the first voxel, each of the following fields utilized the maximum velocity of the previously-solved voxel. Considering the 50% overlap of voxels in each spatial dimension along with the sufficiently small size of each voxel (50% overlap and a side length of 64/512 resulted in 4,096 total voxels), the assumption of similar vector magnitudes in adjacent voxels was a reasonable one. While this approach was used to achieve results in this study, further studies on various σ definitions versus accuracy of results are recommended.

Median Outlier Filtering

While the accuracy of the vision-based 3DPTV by feature association algorithm is sufficiently high, outlier detection and filtering is still a requirement. The algorithm can be sensitive to matching errors due to outliers, such as data dropout caused by incomplete triplets at the edges of an experimental VOI. This can cause a cascade effect

of incorrect matching of multiple vectors. The filtering method relies on comparison of each vector to the median value of all vectors within the surrounding neighborhood. This outlier detection approach may be placed within the tracking process, as shown in Figure 4.10, as well as after the completion of the tracking process. The Median Outlier Detection theory was extended from the 2-D structured version to the 3-D unstructured version by Duncan *et al.* (2010)[49] and Tien (2013),[25] who have fully described the development and extension of the method.

Application to Turbulent Flow

To validate the application of this 3DPTV code to turbulent flow, the Johns Hopkins Turbulence Database (JHTDB) was queried for DNS results in independent domains of 10,000, 25,000, and 50,000 points that were randomly-generated via MATLAB® . The data set was the result of a collaboration between the University of Texas and Johns Hopkins University using a DNS code provided by UT Austin. The domain of the simulation was $8\pi h_c \times 2h_c \times 3\pi h_c$ where h_c is the half-channel height (h_c=1 in dimensionless units). The grid was $2048 \times 512 \times 1536$ in physical space. The viscosity was $\nu = 5 \times 10^{-5}$ and center line Reynolds number was $Re = 2.264 \times 10^4$.[52]

The results granted by the JHTDB query were in the form of 3-dimensional velocity components. These velocity components were re-scaled to a volume of the same order of magnitude as the experimental VOI, treated as 3DPTV data, and used to displace the points originally submitted in the query, $\mathbf{P}_q$, as shown in Equations 4.7 - 4.9.

$$\Delta\mathbf{P} = (\Delta x, \Delta y, \Delta z) \tag{4.7}$$

$$\mathbf{P}_1 = \mathbf{P}_q - \frac{1}{2}\Delta\mathbf{P} \tag{4.8}$$

$$\mathbf{P}_2 = \mathbf{P}_q + \frac{1}{2}\Delta\mathbf{P} \tag{4.9}$$

This resulted in two distinct, randomly sorted sets of 3-D particle coordinates. This was the same format of data that typically resulted from the triplet matching set. These 3-D locations were fed into the 3DPTV algorithm to be matched. The algorithm paired the points and produced 3D3C results. These 3D3C results were then directly compared to the original JHTDB outputs to assess algorithm accuracy within a small tolerance for numerical error.

Table 4.1: Results for 3DPTV tracking algorithm turbulence validation

Vectors	Matching Accuracy
$10,000$	99.85%
$25,000$	99.92%
$50,000$	99.93%

While this was a simple proof-of-concept assessment, a more stringent analysis including the addition of noise and varying voxel side lengths of the type accomplished by Lei *et al.* (2012)[50] is recommended in future study. A sample successfully-tracked data cube consisting of 50,000 vectors from the turbulent channel flow DNS data is shown in Figure 4.11.

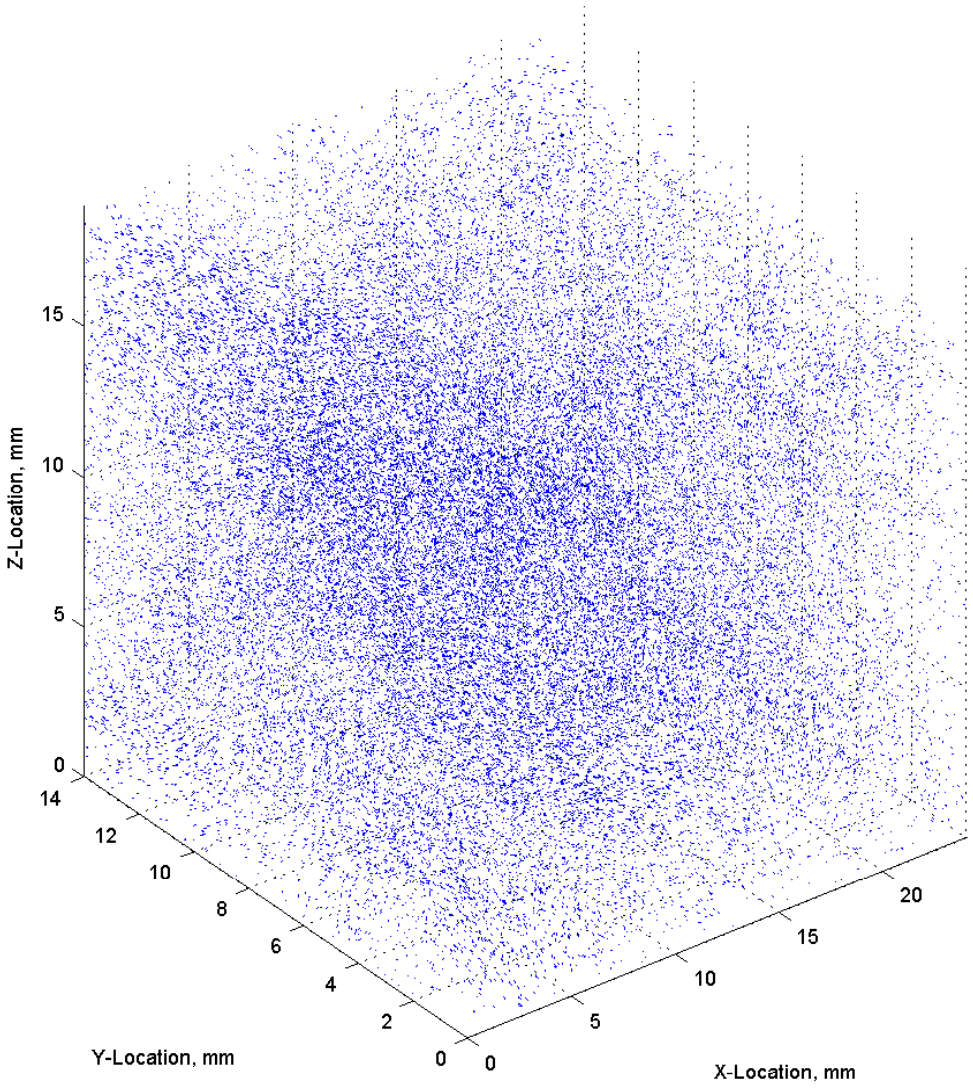


Figure 4.11: Turbulent channel flow tracking validation featuring flow data from the JHTDB, 50,000 vectors.[52,53,54]

4.2.5 Unstructured Interpolation

The result of the 3DPTV process is an unstructured 3-Dimensional velocity vector field. The LES *a priori* testing of SGS stress models is an inherently grid-based process. Thus, an effective and accurate unstructured interpolation method was required to map the 3D3C unstructured velocity vectors to a uniform 3-D grid. Traditional interpolation techniques, including linear, logarithmic, and nearest-neighbor, typically involve higher levels of numerical error, proving inadequate for application to turbulence analysis. In lieu of such methods, the author implemented a second-order Taylor series expansion and utilized the Moore-Penrose Pseudoinverse, a matrix inversion of SVD results, to directly solve for the structured grid values of velocity and its spatial gradients within ellipsoidal neighborhoods centered around each grid point. This method was described and developed for 3-D digitial in-line holography by Talapatra and Katz (2012)[55] and adapted for this study by the author in consultation with Professor Brunton and Professor Bretherton.

First, a standard, second-order Taylor Series Expansion was accomplished to expand the unknown velocities and gradients at grid locations, x^G, to each of n measured, unstructured vector locations, x^n, Equation 4.10, where i, j, and k indices each vary from one to three, representing the x_1, x_2, and x_3 orthonormal basis vectors.

$$u_i^n = u_i(x^n) = u_i(x^G) + \left.\frac{\partial u_i}{\partial x_j}\right|_{x^G}(x_j^n - x_j^G) + \left.\frac{\partial^2 u_i}{2\partial x_j \partial x_k}\right|_{x^G} \times (x_j^n - x_j^G)(x_k^n - x_k^G) \quad (4.10)$$

Then, the matrix on the right hand side of Equation 4.11 was simultaneously decomposed using the SVD and inverted to solve for the grid point data using the

Moore-Penrose Pseudoinverse MATLAB® command, "pinv."

$$\begin{bmatrix} u_1^1 \\ u_2^1 \\ u_3^1 \\ \cdots\cdots \\ \cdots\cdots \\ u_1^N \\ u_2^N \\ u_3^N \end{bmatrix} = \begin{bmatrix} 1\,0\,0\,\Delta x_1^1 & \Delta x_2^1 & \Delta x_3^1 & 0 & \cdots\cdots\ 0 & 0.5(\Delta x_1^1)^2 & \cdots\ 0 & \cdots\ 0 \\ 0\,1\,0\,0 & 0 & 0 & \Delta x_1^1 & \cdots\cdots\ 0 & 0 & \cdots\ 0.5(\Delta x_2^1)^2 & \cdots\ 0 \\ 0\,0\,1\,0 & 0 & 0 & 0 & \cdots\cdots\ \Delta x_3^1 & 0 & \cdots\ 0 & \cdots\ 0.5(\Delta x_3^1)^2 \\ \cdots\cdots & & & & & & & \\ \cdots\cdots\cdots & & & & & & & \\ 1\,0\,0\,\Delta x_1^N & \Delta x_2^N & \Delta x_3^N & 0 & \cdots\cdots\ 0 & 0.5(\Delta x_1^N)^2 & \cdots\ 0 & \cdots\ 0 \\ 0\,1\,0\,0 & 0 & 0 & \Delta x_1^N & \cdots\cdots\ 0 & 0 & \cdots\ 0.5(\Delta x_2^N)^2 & \cdots\ 0 \\ 0\,0\,1\,0 & 0 & 0 & 0 & \cdots\cdots\ \Delta x_3^N & 0 & \cdots\ 0 & \cdots\ 0.5(\Delta x_3^N)^2 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \partial u_1/\partial x_1 \\ \partial u_1/\partial x_2 \\ \partial u_1/\partial x_3 \\ \partial u_2/\partial x_1 \\ \cdots\cdots \\ \partial u_3/\partial x_3 \\ \partial^2 u_1/\partial x_1^2 \\ \cdots\cdots\cdots \\ \partial^2 u_2/\partial x_2^2 \\ \cdots\cdots\cdots \\ \partial^2 u_3/\partial x_3^2 \end{bmatrix}_{x=x^G} \tag{4.11}$$

As noted by Talapatra and Katz, the interpolation method was sufficiently accurate so long as a minimum of 70-80% of ellipsoidal interpolation neighborhoods contained at least 10 unstructured vectors, resulting in a determined or over-determined system of equations, but not under-determined. This quantity of unstructured vectors would result in a minimum of 30 velocity vector components used to determine velocity components, spatial velocity gradients, and second-order moments at a single grid point, a grand total of 30 terms.

The velocity vector fields in this study all contained a minimum of 70% ellipsoidal neighborhoods of at least 10 unstructured vectors each. The resulting fine-scale grid spacing, δ, of the vector fields was 300μm. Velocity vector fields before and after interpolation are presented in Chapter 5. It is recommended that in future study, the same interpolation performance assessment using a flow field divergence-based statistical analysis, as accomplished by Talapatra and Katz, is applied.

4.2.6 A Priori Testing of SGS Stress Models

Once the velocity vectors were interpolated to a uniform 3-Dimensional grid, filtering operations were applied to achieve coarse LES-scale grid spacings of length scale Δ

at various multiples of the fine grid scale δ. Once the velocity data were filtered to a coarse grid scale, the previously-described selection of LES SGS models was applied and stress values were compared to known values calculated from the fine-spacing δ SGS data.

Filtering Technique

Filtering operations were accomplished using the Gaussian Filter type shown in Equation 2.5 with a variance of $\sigma^2 = \frac{1}{12}\Delta^2$. This convolution was applied in the form shown in Equation 2.4. Practically speaking, a MATLAB® filter structure was built using the available command, "fspecial3." This type of structure was a 3-D extension of the 2-D version "fspecial." Once the Gaussian Filter structure was built, it was used in an integral convolution routine in MATLAB® using the available function, "ndnanfilter," which was an N-D extension of the "imfilter" command, filtering around any "NaN" values. This command was also analogous to the N-D convolution command, "convn," using the "same" option for the domain size, again filtering around any "NaN" values. As the results of the operation were smoothed, full-resolution data arrays, they were queried at new, coarse grid locations of spacing $\Delta = 2\delta$, $\Delta = 5\delta$, and $\Delta = 10\delta$ in independent study trials in order to test the effect of filter length on modeling results.

As for models utilizing a second filtering operation, the Similarity and Dynamic Models, the filtering process was repeated at the new filter scale without re-querying the data grid at a coarser spacing. This was to simulate the LES application of the models and to facilitate direct comparison to calculated SGS stress results.

Model Application

Once the filtering of the data was completed to achieve the coarse LES grid, the SGS models described earlier were applied in MATLAB® as developed in Section 2.3.1. Results from each model calculation were then compared directly to the known values of SGS stress calculated from the full-resolution data using Equation 2.7.

Chapter 5

RESULTS AND DISCUSSION

"but vether its worth goin through so much, to learn so little... is a matter o' taste."

Charles Dickens

In this chapter, results from the 3DPTV portion of the study are presented along with uncertainty analysis for position and displacement accuracy. Then, SGS model study results are shown including contour plots of SGS stress and energy dissipation or transfer as well as correlation coefficients at varying filter scales. This section displays and discusses a sampling of the full study results. Additional plots are listed in Appendix B.

5.1 3DPTV Results

Unstructured 3DPTV results were attained for a total of six vector fields in order to insure statistical convergence. The total vector count of these fields ranged from 20,000 to 25,000. A sample unstructured vector field with flow direction indicated is shown in Figure 5.1. As details of the flow field are difficult to discern in such a dense vector field, a magnified 3-D region of the flow is also shown to highlight fine details captured by the high-resolution 3DPTV method.

These unstructured vector fields were interpolated to a 3-D structured grid with 300μm spacing for a total of approximately 163,000 grid locations based on the approach developed by Talapatra and Katz (2012).[55] A sample result of the structured interpolation method is shown in Figure 5.2. As with the unstructured velocity vector

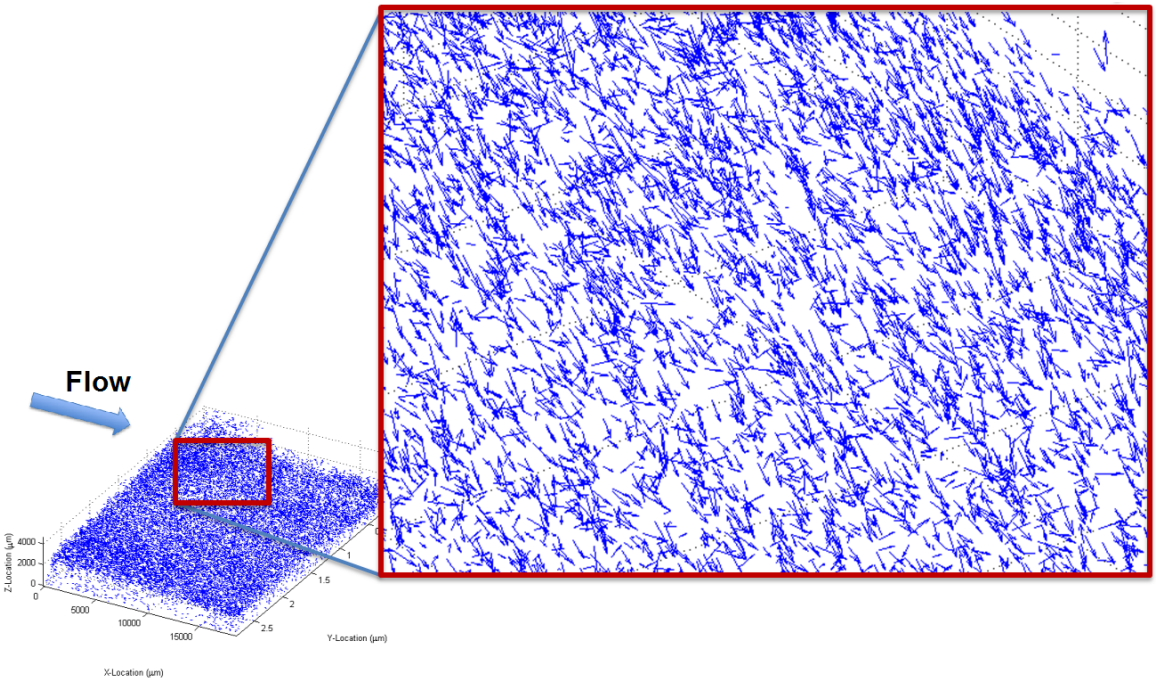


Figure 5.1: Sample 3DPTV results in the form of a 3D3C unstructured velocity vector field. Dimensions of the data volume were approximately 30mm×20mm×3.3mm. Incoming flow direction indicated.

field, a representative volume was magnified in order to show fine details of the flow field.

A final sampling of the data is a vorticity contour plot, Figure 5.3. This plot was generated by calculating the curl of the full 3-D velocity field and extracting the vorticity component in the Y-direction, ω_{xz}. To show the mean structure of the flow field in the XZ-plane, the ensemble average of the vorticity component was taken along the Y-direction and the resulting mean values were plotted. The resulting vorticity values in the vortex sheets were within $\pm 3u_{\text{in}}/h$ or three times the characteristic vorticity of the incoming flow over the backward-facing step.

As the illumination filled a volume larger than that of the VOI, the vector fields

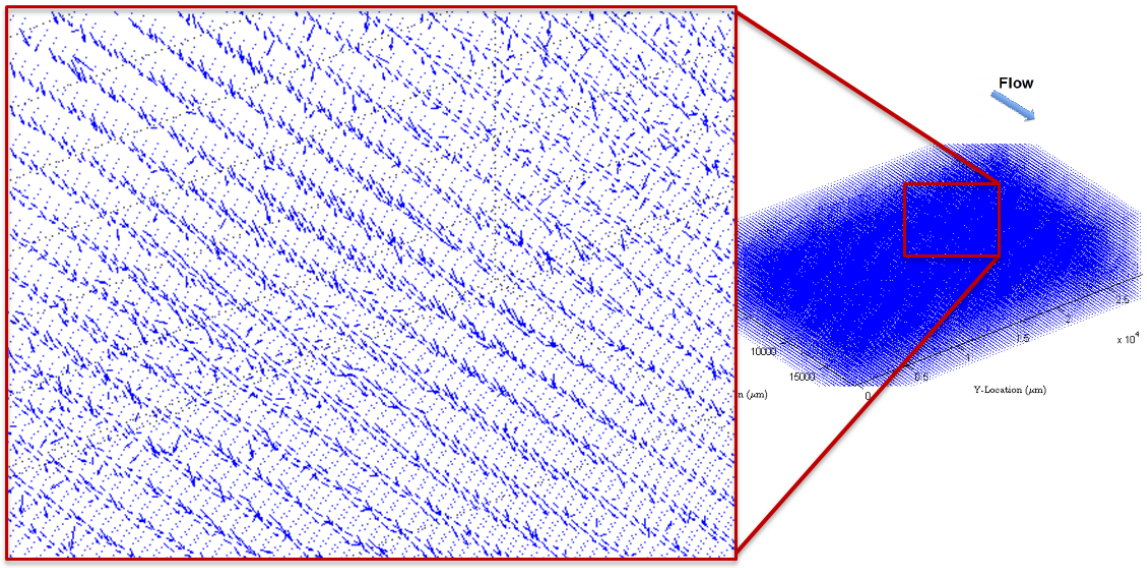


Figure 5.2: Sample 3DPTV results in the form of a 3D3C structured velocity vector field with grid spacing $\delta = 300\mu$m. Incoming flow direction indicated.

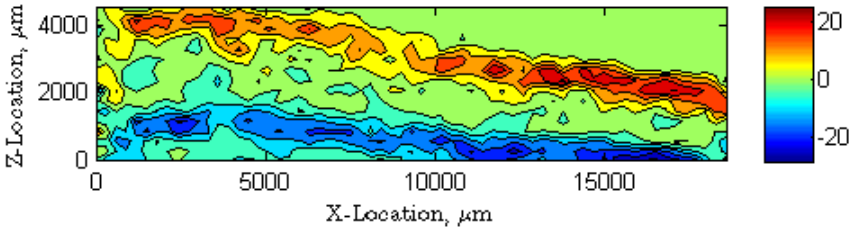


Figure 5.3: A contour plot of the mean vorticity component normal to the Y-axis, ω_{xz}, from a single vector field. Units are s^{-1}. Flow was along the positive X-direction.

were reduced to the Z-range with the greatest vector density to facilitate turbulence studies on the most valid regions of the interpolation (regions with most neighborhoods containing at least 10 unstructured vectors). This limited each Z-range to the final value of 3.3mm for a total of 67,000 gridded vectors ultimately considered in the

LES filtering process of each structured vector field.

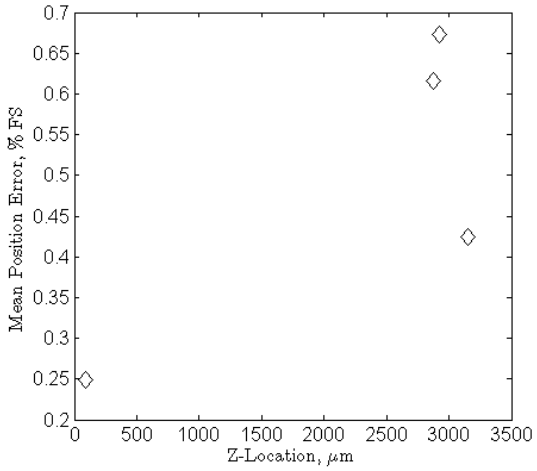
5.2 Uncertainty Analysis

The uncertainty analysis for the 3DPTV system developed in this study was accomplished using the same approach taken by Tien in the 3DμPTV system development[25] to assess position and displacement accuracy. The known, regular geometry of the calibration target plate and the precise control of the linear translation stages were used to collect uncertainty data with known position and displacement quantities.

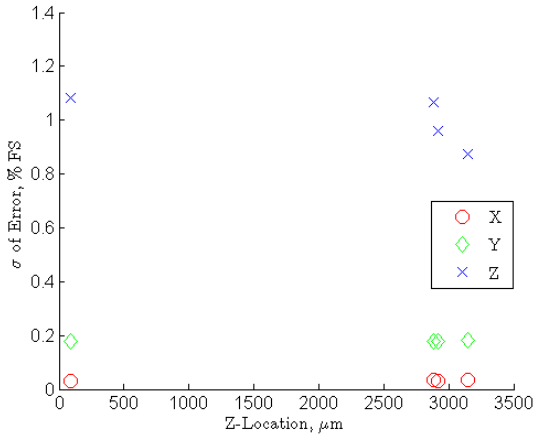
For position uncertainty, during the collection of calibration data two additional Z-planes at intermittent locations were imaged and not used to build the original calibration. These calibration planes were then analyzed using the full data reduction process and dots were located in 3-D using the previously-built calibration. The results were assessed for X, Y, and Z location accuracy relative to the known geometry. The results of the position uncertainty analysis are shown in Figure 5.4 in a side-by-side comparison to the uncertainty values for the 3DμPTV system developed by Tien (2013).[25] Position uncertainties in this study were on the same order of magnitude as those found for the 3DμPTV study.

In order to assess displacement uncertainties, additional translation stages were installed on the calibration rig to translate the plate by known displacements in the X and Y directions. These displacements were assessed using the calibration data and mean errors and standard deviations of errors were plotted in Figure 5.5 at two displacements, 25.4μm and 50.8μm, corresponding to 0.001" and 0.002" values, respectively, which were the finest increments of the linear translation stages. These displacement errors were compared side-by-side to those found in the 3DμPTV study.

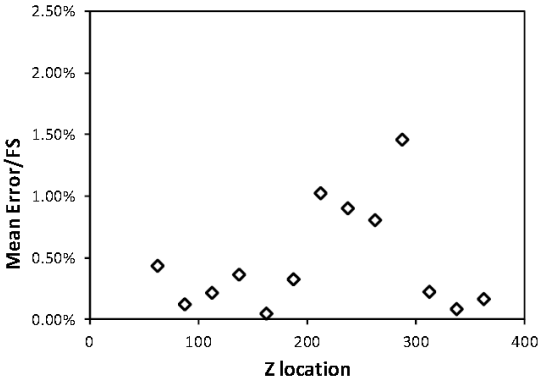
The displacement uncertainties analyzed in this study were on the same order of magnitude as those found in the work of Tien (2013).[25] Some possible sources



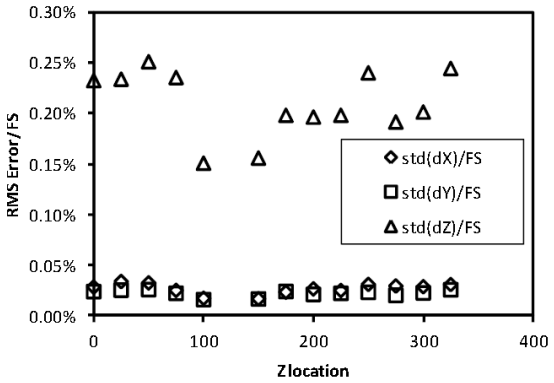
(a) Mean of position errors



(b) Standard deviations of position errors



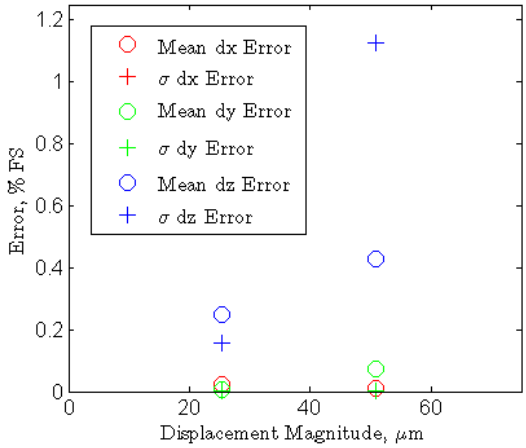
(c) Mean of position errors from the 3DμPTV study by Tien (2013)[25] (Z location in μm)



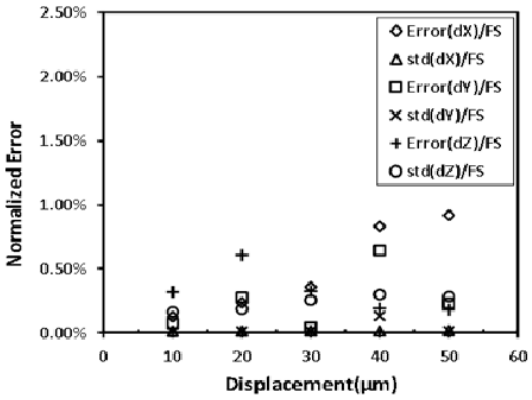
(d) Standard deviations of position errors from the 3DμPTV study by Tien (2013)[25] (Z location in μm)

Figure 5.4: Position uncertainty comparison between turbulent 3DPTV and 3DμPTV systems.

of error in this study are CCD sensor noise or lighting variations skewing the peaks of particle images, inaccuracy in the linear translation stage or alignment of the calibration rig, particle image saturation, and slight variation in the positioning of



(a) Means and standard deviations of displacement errors from the turbulent 3DPTV study.



(b) Means and standard deviations of displacement errors from the 3DμPTV study by Tien (2013)[25]

Figure 5.5: Displacement uncertainty comparison between turbulent 3DPTV and 3DμPTV systems.

the prism and tunnel lid. Additionally, the image processing steps designed to remove image noise and adjust contrast may have introduced peak shifts that resulted in mapping inaccuracies during the triplet matching and particle location steps. While only a limited selection of uncertainty testing was accomplished in this study, Tien (2013)[25] has extensively established the validity of the methods applied herein. As the uncertainty levels between studies were comparable, this application of the 3DPTV method is an accurate and valid extension.

5.3 SGS Stress Model Study

As previously mentioned, the Smagorinsky, Similarity, Mixed, Dynamic, and Coherent Structures Models were applied to the filtered velocity field. The velocity fields were

also filtered at varying LES length scales relative to the fine resolution scale $\Delta = 10\delta$, $\Delta = 5\delta$, and $\Delta = 2\delta$. Results presented in this section were filtered at a length scale of $\Delta = 5\delta$ unless otherwise noted.

As described in-depth by Meneveau and Katz (2000)[7] as well as in Chapter 2, there are many ways by which the performance of SGS stress models is assessed. In this section, values of modeled SGS stress tensor elements, $\tau_{ij}^{\Delta,\mathrm{mod}}$, and SGS dissipation or transfer of kinetic energy, $-\tau_{ij}^{\Delta,\mathrm{mod}}\widetilde{S}_{ij}$ are compared directly to their experimental counterparts, τ_{ij}^{Δ} and $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$, where experimental SGS stress is calculated by Equation 2.7, via contour plots. Methods of accumulating Pearson Correlation Coefficients for SGS stress tensor elements, SGS force, and SGS energy dissipation are discussed, and the corresponding results are compared.

5.3.1 SGS Stress Contours

Sample stress contour plots for the XY stress tensor element, τ_{12}^{Δ}, are shown in Figure 5.7*, for the experimental values as well as the values from each of the five stress models. While these are simply 2-D plots of a single tensor element in a single XY-plane, 3-D data are available for each of the three deviatoric stress tensor components, τ_{12}^{Δ}, τ_{23}^{Δ}, and τ_{13}^{Δ}.

Of all the models, the contour plots for the Similarity and Mixed Models most closely resemble the contour plot of the experimental values. This resemblance is also mirrored by the high correlation coefficients exhibited by the Similarity and Mixed Models, discussed in Section 5.3.3. The matching of the Similarity and Mixed Models follows logically from the Mixed Model's definition, where the Mixed Model is simply equal to the Similarity Model with the addition of the Smagorinsky Model, shown in

*The discussion of Figure 5.6 occurs in Section 5.3.3. The figure numbering sequence has been altered to accommodate complete pages for Figures 5.7 and 5.8.

Equation 2.13. This resemblance also reflects the results from Liu as presented by Meneveau and Katz (2000),[7] where the Similarity and Mixed Models' contour plots showed the best resemblance to that of the experimental. Additionally, in the plots presented by Meneveau and Katz,[7] the Similarity and Mixed Models' contours were nearly identical to one another; this correspondence can also be seen in Figures 5.7c and 5.7e of this study.

The Smagorinsky, Coherent Structures, and Dynamic Model contour plots show very little resemblance to the experimental plot. This trend is also reflected by the low correlation coefficient values for SGS stress tensor elements, discussed in Section 5.3.3. The Smagorinsky plot shares very little commonality with the experimental plot, as was the case in the study by Liu *et al.*[21] where regions of higher SGS stress in the Smagorinsky contours seemed nearly random in comparison to the experimental contours.

5.3.2 SGS Dissipation Contours

A selection of contour plots for the measure of local SGS dissipation, $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$, is shown in Figure 5.8, for experimental and modeled values. As with the stress contour plots, these are 2-D plots of a single XY plane, however, 3-D data are available for SGS energy dissipation.

It is apparent that some regions of the plots shown for the experimental, Similarity Model, and Mixed Model dissipation display negative values or what amounts to energy backscatter. This local backscatter has been regularly observed in past studies, seen by Piomelli (1991)[23] and Liu (1994).[21] Examination of the contour plots of local energy dissipation for the Smagorinsky, Coherent Structures, and Dynamic Models reveals that these models display no backscatter; in other words, they are purely dissipative in nature. This lack of local backscatter has been noted as a limitation

of the eddy viscosity model, which is the general form of the aforementioned models. This is an undesirable attribute of the eddy viscosity model form and contributes to its lower correlation values.

The results for SGS energy dissipation were similar to the results for the SGS stress elements. The Similarity and Mixed Model contours closely resemble the experimental contour plot, which gave way to higher correlation coefficient values, as summarized in Section 5.3.3. The contours of the Smagorinsky, Coherent Structures, and Dynamic Models, however, do not resemble the experimental contours and have negligible correlations. The contour plots for these three models, however, do show some resemblance to one another. This is likely due to the fact that these three models take the form of an Eddy-Viscosity Model while the Similarity Model does not.

5.3.3 Accumulation of Statistics

In order to accumulate a sufficient amount of statistics such that the correlation coefficients calculated in the study converged, as shown in Figure 5.6, six velocity vector fields were used. Edge quantities were ignored in the calculation of correlation coefficients in order to avoid any numerical errors resulting from filtering or calculating gradients near the edge. To amass more statistics and to allow for better convergence of correlations, the sampling locations for the coarse, LES grid were repeatedly rotated by a fine grid length scale, δ, for a full interval of the coarse length scale, Δ, and models were reapplied at each set of locations. These practices were adopted as described by Liu *et al.* (1994)[21] and again by Tao *et al.* (2002).[24]

The Pearson Correlation Coefficient was assessed between experimental and modeled values of SGS dissipation, SGS force, and SGS stress tensor elements or $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$, $\nabla \cdot \tau^{\Delta}$, and τ_{ij}^{Δ}, respectively. The correlation coefficient computations were accom-

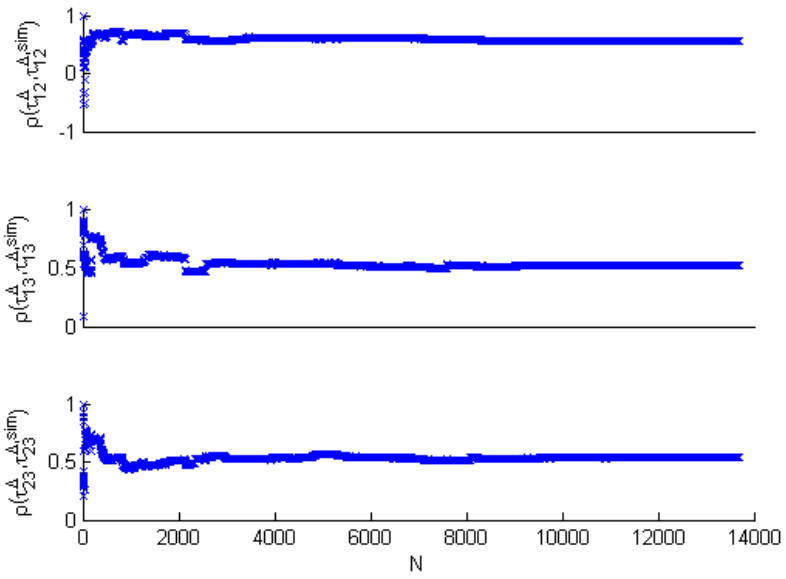
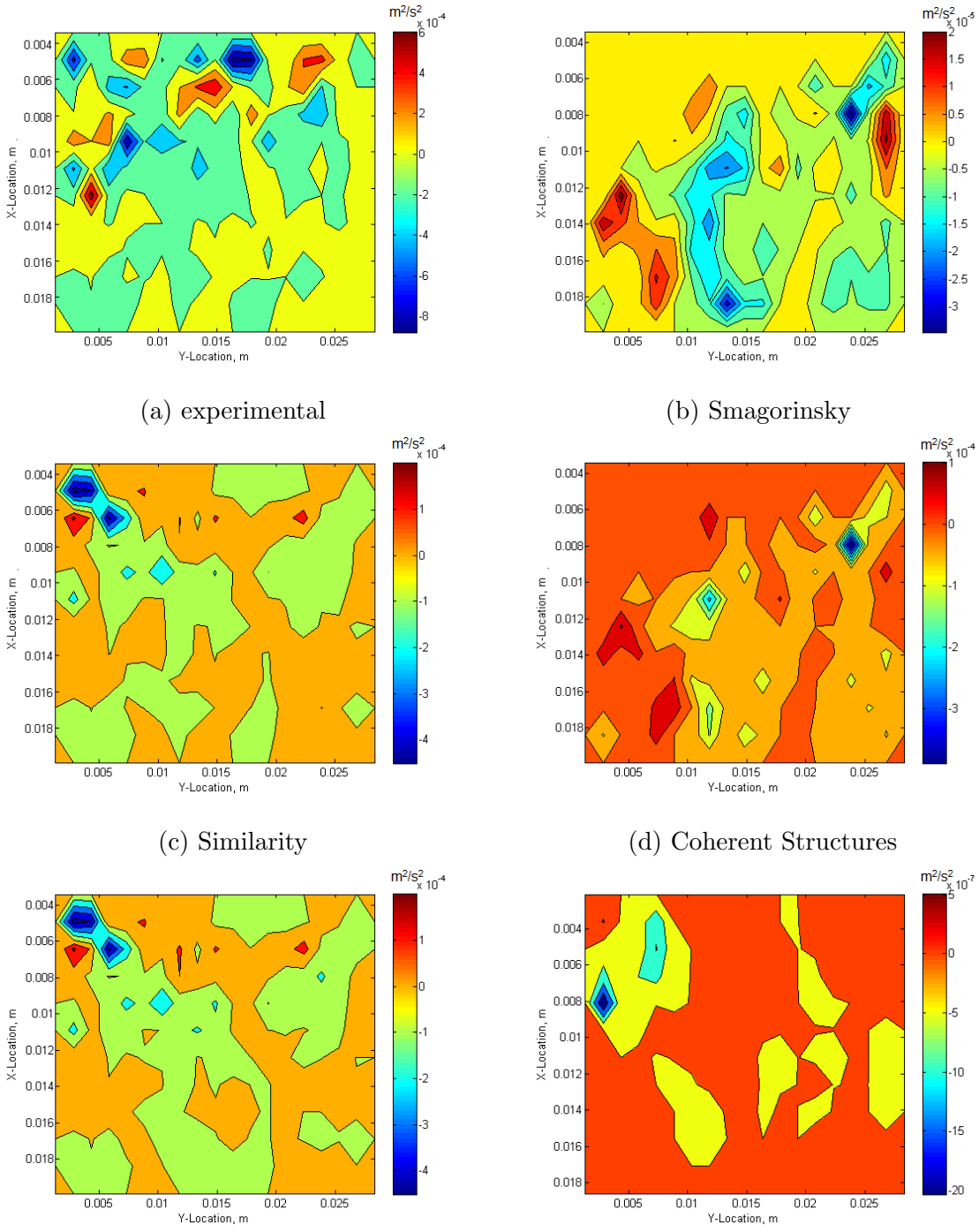


Figure 5.6: A correlation convergence plot showing the correlation between experimental and Similarity Modeled SGS stress tensor elements, $\rho(\boldsymbol{\tau}_{ij}^{\Delta}, \boldsymbol{\tau}_{ij}^{\Delta,\mathrm{Sim}})$.

plished cumulatively, utilizing increasing portions of the full data sets in order to plot and check for convergence. The results of each of these correlations are summarized over the following subsections. Coefficient values are also compared to values summarized in the work of Meneveau and Katz (2000).[7] Cumulative plots for each of the correlation quantities at each filter length scale can be found in Appendix B.



(a) experimental

(b) Smagorinsky

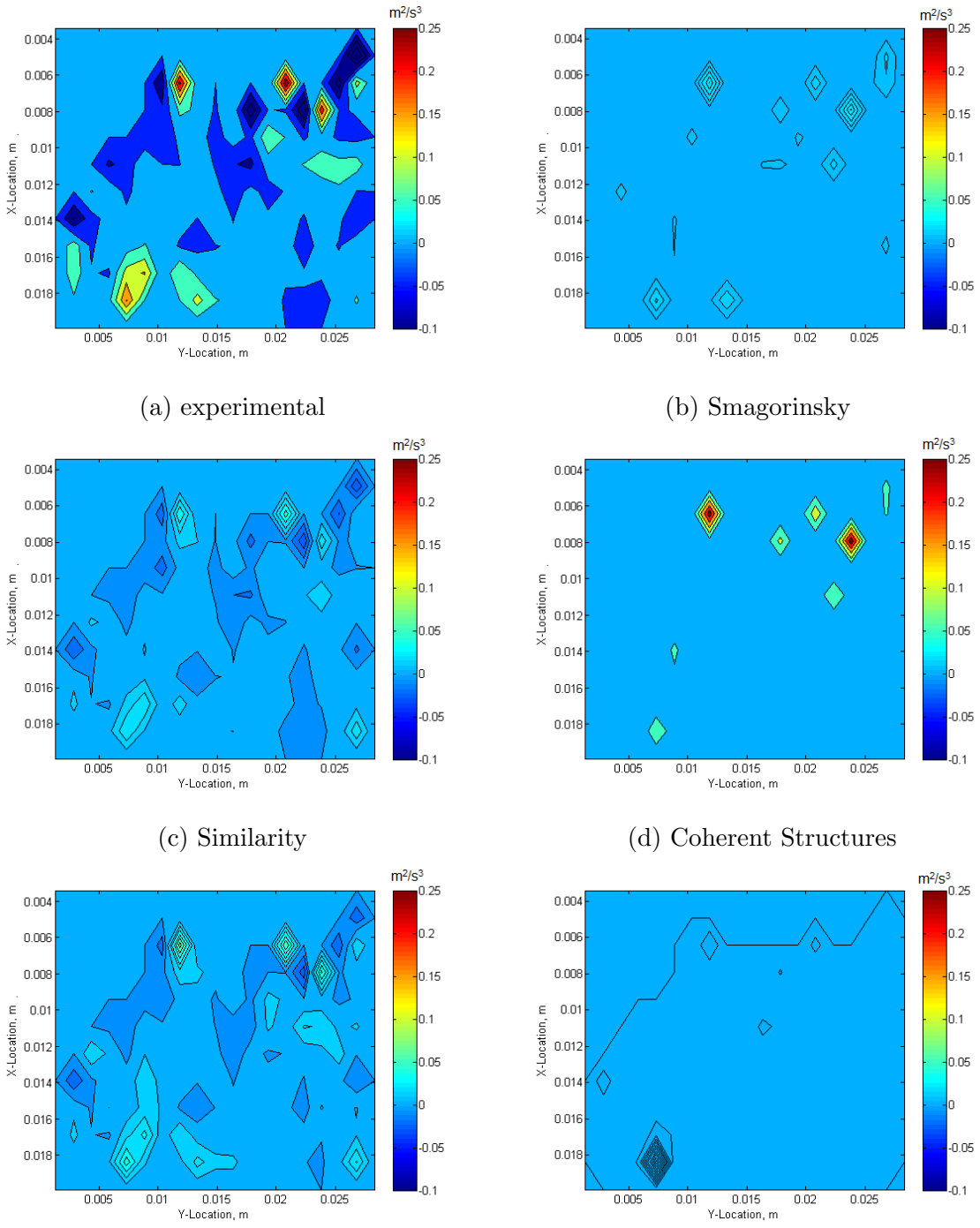
(c) Similarity

(d) Coherent Structures

(e) Mixed

(f) Dynamic

Figure 5.7: A comparison of contour plots for the 12 or XY SGS stress tensor element, τ_{12}^{Δ}. Flow direction is top-to-bottom.



(a) experimental

(b) Smagorinsky

(c) Similarity

(d) Coherent Structures

(e) Mixed

(f) Dynamic

Figure 5.8: A comparison of contour plots for SGS energy dissipation approximated by the transfer of energy to the SGS, $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$. Flow direction is top-to-bottom.

Correlation of SGS Stress Tensor Elements

Final values for the correlation coefficients of the deviatoric SGS stress tensor elements are summarized in Table 5.1. The Smagorinsky Model stress tensor element correlations are in the expected range of 0∼0.2 for all of the correlations presented in this study.[7] The Similarity and Mixed Models had relatively high correlations and were comparable to each other as suggested by Meneveau and Katz (2000).[7]

The Dynamic and Coherent Structures Models, however, did not perform well in any of the correlation diagnostics. It is hypothesized that for the Dynamic model, the coefficient values were consistently negative due to the complex nature of the flow being studied. This led to minimal or negative correlation values, even after the ensemble averaging of coefficients. While the averaging of Dynamic Model coefficient values along a direction of statistical homogeneity has traditionally shown improvement in the consistency of modeled values, the coefficients in this study still exhibited highly variable behavior, including some negative values. In addition to the complex nature of the flow, more data points in the step-wise Y-axis may be required in order to achieve stable Dynamic Smagorinsky coefficient values.

According to Kobayashi *et al.* (2008),[9] the Coherent Structures Model has been described as having the same level of performance as the Dynamic Smagorinsky Model.[9] For this case, the Coherent Structures Model was on the same level of performance from a correlation coefficient standpoint. Kobayashi's comparison is not necessarily extensible to the present results as the Dynamic Model coefficients were ultimately not consistent, still having negative and fluctuating values even after spatial ensemble averaging in the direction of statistical homogeneity.

Table 5.1: Correlation coefficients of modeled deviatoric stress tensor elements. Columns represent $\rho(\tau_{12}^{\Delta}, \tau_{12}^{\Delta,\mathrm{mod}})$ for the relevant model and stress tensor component.

Model Type	τ_{12}^{Δ}	τ_{13}^{Δ}	τ_{23}^{Δ}
Smagorinsky Model	0.0497	0.0118	0.0288
Similarity Model,$\gamma = 1$	0.5553	0.5174	0.5379
Similarity Model,$\gamma = 2$	0.4310	0.3552	0.3768
Mixed Model,$\gamma = 1$	0.5554	0.5170	0.5377
Mixed Model,$\gamma = 2$	0.4310	0.3543	0.3764
Dynamic Model,$\alpha = 1$	-0.0238	-0.0502	-0.0152
Dynamic Model,$\alpha = 2$	-0.0431	0.0045	0.0135
CSM	0.0490	0.0126	0.0444

Correlation of SGS Force

Final correlation coefficient values for SGS force, the divergence of SGS stress, are presented in Table 5.2. As previously described, the Similarity and Mixed Models correlations are relatively high. The Smagorinsky, Dynamic, and Coherent Structures Models were negligibly correlated. As previously mentioned, these models of the eddy viscosity form did not exhibit optimal performance for the same reasons as previously mentioned.

Correlation of SGS Dissipation

Finally, the correlation of the estimation of SGS dissipation, or energy transfer to the SGS, are summarized in Table 5.3. In a mean sense, all models achieved the correct order of magnitude of SGS dissipation except for the Dynamic and Coherent Structures Model. The Dynamic Model showed nearly no energy dissipation while the Coherent Structures Model overestimated dissipation by an order of magnitude.

Table 5.2: Correlation coefficients of modeled SGS Force, $\nabla \cdot \tau^{\Delta}$

Model Type	$\rho(\nabla \cdot \tau^{\Delta}, \nabla \cdot \tau^{\Delta,\mathrm{mod}})$
Smagorinsky Model	-0.0055
Similarity Model,$\gamma = 1$	0.4805
Similarity Model,$\gamma = 2$	0.4384
Mixed Model,$\gamma = 1$	0.4795
Mixed Model,$\gamma = 2$	0.4367
Dynamic Model,$\alpha = 1$	-0.1924
Dynamic Model,$\alpha = 2$	-0.0932
CSM	0.0005

The Similarity Model, while having very high correlation coefficients, did not result in adequate levels of dissipation without the addition of the Smagorinsky Model to form the Mixed Model. The addition of the Smagorinsky Model lowered the correlation coefficient, however, resulted in better mean dissipation levels. The trends for the Smagorinsky, Similarity, and Mixed Models followed many of those summarized by the review literature.[7] The trends in the Dynamic and Coherent Structures Models, however, do not necessarily reflect the performance of these models according to past studies and warrant additional examination and future study.

5.3.4 Summary

From the data acquired in this study, a selection of which is presented above, it is clear that the best-performing model was the Mixed Model. As expected from previous studies, the Mixed Model had the high correlation coefficient values provided by the Similarity Model and the adequate dissipation added from the Smagorinsky Model.[7] From the correlation coefficients of dissipation, mean dissipation, and global dissipation standpoints, the Mixed Model would be the model of choice as deduced

Table 5.3: Correlation coefficients of modeled SGS Dissipation, $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$

Model Type	$\rho(-\tau_{ij}^{\Delta}\widetilde{S}_{ij}, -\tau_{ij}^{\Delta,\mathrm{mod}}\widetilde{S}_{ij})$
Smagorinsky Model	0.1123
Similarity Model,$\gamma = 1$	0.8645
Similarity Model,$\gamma = 2$	0.8088
Mixed Model,$\gamma = 1$	0.7886
Mixed Model,$\gamma = 2$	0.7936
Dynamic Model,$\alpha = 1$	-0.1358
Dynamic Model,$\alpha = 2$	-0.0466
CSM	0.0840

by this study. If the performance of the Dynamic Model could be improved by the addition of more data in the step-wise direction, it is reasonable to expect that the performance of a Dynamic Mixed Model would exceed that of the traditional Mixed Model. To this end, further study should be focused on increasing system resolution and VOI size in pursuit of better model performance and stability.

Chapter 6

CONCLUSIONS AND FUTURE STUDY

> *"It is one of our most exciting discoveries that local discovery leads to a complex of further discoveries. Corollary to this we find that we no sooner get a problem solved than we are overwhelmed with a multiplicity of additional problems in a most beautiful payoff of heretofore unknown, previously unrecognized, and as-yet unsolved problems."*
>
> Buckminster Fuller

Although only a sampling of the data collected in this study has been presented, conclusions may be drawn regarding the success of the implementation of the 3DPTV system, the application of LES SGS models, and testing of these models. Additionally, future studies are recommended based on questions that were raised throughout the study development and analysis.

6.1 Conclusions

A macro-scale 3DPTV system based on the triangulation method of particle location mapping was developed with the spatial resolution required for LES SGS stress model testing for turbulent flows. To this end, the flow facility was modified featuring a test section reconstructed with a clear acrylic floor for back-lit calibration and an additional particle seeding port. The camera system, including CCDs, 6-axis translation stages, lenses, jig assembly, and water-filled prism from previous experimentation were used to develop the new 3DPTV system. To follow the approach taken by Tien *et al.* (2014)[26] on the micro-scale, calibration components including

the calibration rig, water-filled prism platform, and a back-lighting assembly were designed and constructed to facilitate the building of a complete calibration database to characterize the full VOI. The volume of the calibration database was nearly four orders of magnitude greater than the one featured in previous work, yet had comparable uncertainty levels.[25] To allow for finer assessment of data acquired from turbulent flows, a global gray-level thresholding technique used to isolate blob regions of images with unimodal histogram distributions, Rosin's Method,[43] was adopted to great effect. Also, the 3DPTV vision-based feature association method was fine-tuned with improved data distribution among computational voxels and iterative σ definition for characteristic displacement magnitudes within each voxel. Both of these tracking modifications allowed finer resolution of small-scale, localized turbulent motions.

To bridge the gap between traditional unstructured 3DPTV results and LES SGS model testing, the second-order Taylor Series Expansion SVD interpolation developed by Talapatra and Katz (2012)[55] was adopted. A variety of historically significant and recently-proposed SGS stress models was applied and compared to known stress values, and a base-level model performance analysis was conducted. In summary, all study goals defined at the outset were accomplished along with additional contributions that were made over the course of the study. The work described in this paper has laid the foundation for turbulent 3DPTV study and SGS stress model testing for future experimentation. While this study certainly raised many new questions, to be detailed in Section 6.2, it is the author's hope that, with this initial groundwork, new applications of the 3DPTV system will be ultimately be pursued, and a new model may be proposed.

6.2 Future Study

Questions that have arisen throughout the study development and analysis gave way to recommendations for important future studies to be conducted. While some of the future studies may be accomplished on the current data, it is recommended that further data collection is conducted with new and improved configurations and study designs. The full list and description of recommended future studies is broken down by category in the subsections that follow and listed in order of importance. The merits and importance of each future study are also discussed.

6.2.1 Data Processing and Analysis

In an effort to further validate data reduction methods and results achieved in this study, it is recommended to further examine a few steps of the process. Additionally, further studies may be completed using additional processing techniques and SGS stress model types. These updates and additions to data processing and analysis are presented in order of importance in the list that follows:

1. The second-order Taylor Series Expansion SVD interpolation implemented in this study should be fully assessed using the techniques developed and fully described by Talapatra and Katz (2012).[55] The assessment technique consisted of the accumulation of statistical data that characterized the level of divergence present in the interpolated vector field at various grid spacing intervals. The author recommends that this test be performed iteratively in conjunction with the 3-D unstructured median outlier filtering technique to achieve the most accurate, highest-density, divergence-free interpolation with the fewest outliers and smallest grid spacing. While the test may not need to be as extensive as the one performed by Talapatra and Katz, the author believes it is critical to proving

the validity of the data. If further median outlier filtering in conjunction with the divergence assessment fails to produce valid results, the author recommends incorporating the divergence-free filtering techniques mentioned later in this list.

2. Newer models not examined in this study should be applied and tested, including the Stretched-Vortex Model developed by Misra and Pullin (1997).[56] The model relates SGS stress values to SGS vortex structure orientations and kinetic energy using the GS data. The application and testing of this model could provide further validation for the methods and data collected in this study.

3. A processing technique that may be of interest to future researches is the divergence-free solenoidal filtering technique for 3-D velocity measurements developed by Schiavazzi *et al.* (2014).[57] The filtering technique adjusts inaccuracies in flow data by enforcing incompressibility, insuring that the zero divergence condition is upheld locally. This technique may be especially useful in future studies if the velocity fields exhibit divergence when tested using the approach mentioned above.

4. Another aspect of the 3DPTV tracking algorithm that must be analyzed further is the iterative σ definition. The effect of an iterative σ definition on the accuracy of tracking results should be assessed in a controlled experiment, possibly in combination with the test of the overall algorithm performance when applied to turbulent data. In this test, different definitions of σ should be compared such as the mean, median, or maximum of the previous voxel's data or some multiple of these values.

5. Other processing techniques that should be explored include the Proper Orthogonal and Dynamic Mode Decompositions (POD and DMD, respectively)

employed by Sampath and Chakravarthy (2014).[58] As the POD is simply an application of the SVD on a matrix containing columns of spatially-sampled concatenated velocity vector components, each from a distinct field, it is a straightforward application that could shed light on structures present in the velocity field and the relationships between fields at different moments in time. The DMD, however, requires temporally-resolved flow field data. As the data rate of the system in this study was relatively low, the DMD may not be applicable to this case but may be applied to results from a new system with higher temporal resolution.

6. While the 3DPTV tracking algorithm featured in this study has been tested extensively by Lei *et al.* (2012),[50] only a limited test on high-Reynolds number turbulence has been accomplished in this study. It is recommended that an approach similar to the one taken by Lei *et al.* is accomplished using data queried from the JHTDB with the addition of noise. This will help simulate real data that are likely to contain some portion of imperfect or incomplete particle location inputs.

7. The impact that the image processing steps, de-noising and contrast adjustment, have on the accuracy of peak locations should be explored. The question has been raised as to whether or not the Gaussian blur and the histogram stretching processes erroneously shift the sub-pixel peak locations. It should be noted, though, that past work by Ponchaut (2005)[48] has found that a Gaussian blur process improves the accuracy of peak location.

6.2.2 Experimental Hardware

In pursuit of achieving even higher spatial resolution, some experimental changes should be explored. These changes are presented in order of importance in the list that follows:

1. Additional seeding locations may be created in various locations throughout the water tunnel and test section. These additional locations will allow for complete and consistent seeding densities in the VOI regardless of particle dropout that may occur due to higher specific gravity.

2. Smaller particles should be used for seeding to maintain volume ratios well below the two-way coupling limit discussed earlier in this study and to allow for more vectors per image pair.

3. Modification to the system to allow for the reduction of particle image areas to 16px^2 would allow for the capture of approximately 200,000 vectors per image pair. This would allow for an increased Z-depth of 10mm and a step Reynolds number, Re_h, of 8,000. Note: Incorporation of an additional beam-expanding cylindrical lens to increase the Z-depth of the VOI is required once sufficient density is achieved.

4. The largest improvement to system resolution would result from the implementation of new 50.1 MP cameras from Illunis. Such an increase in resolution along with the imaging of smaller particles would result in approximately 10^6 vectors per image pair. This spatial resolution would, in theory, allow for a Z-depth of 20mm and a step Reynolds number of 21,000. While these values are rough estimates and may not scale as predicted, there is certainly room for orders of magnitude of improvement in the system's resolution.

In addition to the primary effects of improved spatial resolution, the capture of wider, more-developed energy spectra would allowing for better LES filter placement within the inertial range for improved analysis of the balance of turbulent kinetic energy. With an increased VOI and resolution would also come improved modeling fidelity, for models such as the Coherent Structures Model and the Dynamic Model, and the ability to test across a wider range of LES filter length scales. While there is certainly merit in the collection of turbulent experimental data in its own right, these resolution increases may even push the 3DPTV system's fidelity beyond the capability of modern DNS, depending on the flow type and relevant experimental conditions.

BIBLIOGRAPHY

[1] B F Armaly, F Durst, J C F Pereira, and B Schönung. Experimental and theoretical investigation of backward-facing step flow. *J. Fluid Mech.*, 1983.

[2] R Grothe. Modification of a DDPIV system for imaging small volumes using triangulation for particle identification. Master's thesis, University of Washington, 2008.

[3] P. G. Spazzini, G. Iuso, M. Onorato, N. Zurlo, and G. M. Di Cicca. Unsteady behavior of back-facing step flow. *Experiments in Fluids*, 30(5):551561, May 2001.

[4] Stephen B. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

[5] T. R. Troutt, B. Scheelke, and T. R. Norman. Organized structures in a reattaching separated flow field. *J. Fluid Mech.*, 143(-1):413, Jun 1984.

[6] F. Scarano, C. Benocci, and M. L. Riethmuller. Pattern recognition analysis of the turbulent flow past a backward facing step. *Physics of Fluids*, 11(12):3808, 1999.

[7] C Meneveau and J Katz. Scale-invariance and turbulence models for large-eddy simulation. *Annu. Rev. Fluid Mech.*, 2000.

[8] Hiromichi Kobayashi. The subgrid-scale models based on coherent structures for rotating homogeneous turbulence and turbulent channel flow. *Physics of Fluids*, 17(4):045104, 2005.

[9] H. Kobayashi, F. Ham, and X. Wu. Application of a local SGS model based on coherent structures to complex geometries. *International Journal of Heat and Fluid Flow*, 29(3):640653, Jun 2008.

[10] Charles Meneveau, Thomas S. Lund, and William H. Cabot. A lagrangian dynamic subgrid-scale model of turbulence. *J. Fluid Mech.*, 319(-1):353, Jul 1996.

[11] Mustafa Barri, George K. El Khoury, Helge I. Andersson, and Bjrnar Pettersen. DNS of backward-facing step flow with fully turbulent inflow. *International Journal for Numerical Methods in Fluids*, page n/an/a, 2009.

[12] Pankaj M. Nadge and R. N. Govardhan. High reynolds number flow over a backward-facing step: structure of the mean separation bubble. *Experiments in Fluids*, 55(1), Jan 2014.

[13] Toshio Kobayashi, Youhei Morinishi, and Keon-Je Oh. Large eddy simulation of backward-facing step flow. *Communications in Applied Numerical Methods*, 8(7):431441, Jul 1992.

[14] E.W. Adams and J.P. Johnston. Effects of the separating shear layer on the reattachment flow structure part 2: Reattachment length and wall shear stress. *Experiments in Fluids*, 6(7), 1988.

[15] J. Smagorinsky. General circulation experiments with the primitive equations. *Monthly Weather Review*, 91(3):99164, Mar 1963.

[16] D.K. Lilly. The representation of small scale turbulence in numerical simulation experiments. *IBM Sci. Comput. Symp. Environ. Sci.*, page 195210, 1967.

[17] A. Leonard. Energy cascade in large-eddy simulations of turbulent fluid flows. *Turbulent Diffusion in Environmental Pollution, Proceedings of a Symposium held at Charlottesville*, page 237248, 1975.

[18] J. Bardina, J. Ferziger, and W. Reynolds. Improved subgrid-scale models for large-eddy simulation. *13th Fluid and PlasmaDynamics Conference*, Jul 1980.

[19] Massimo Germano, Ugo Piomelli, Parviz Moin, and William H. Cabot. A dynamic subgrid-scale eddy viscosity model. *Phys. Fluids A*, 3(7):1760, 1991.

[20] D. K. Lilly. A proposed modification of the germano subgrid-scale closure method. *Phys. Fluids A*, 4(3):633, 1992.

[21] S Liu, C Meneveau, and J Katz. On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet. *J. Fluid Mech.*, 1994.

[22] Shewen Liu, Charles Meneveau, and Joseph Katz. Experimental study of similarity subgrid-scale models of turbulence in the far-field of a jet. *Appl. Sci. Res.*, 54(3):177190, Apr 1995.

[23] Ugo Piomelli, William H. Cabot, Parviz Moin, and Sangsan Lee. Subgrid-scale backscatter in turbulent and transitional flows. *Phys. Fluids A*, 3(7):1766, 1991.

[24] B Tao, J Katz, and C Meneveau. Statistical geometry of subgrid-scale stresses determined from holographic particle image velocimetry measurements. *J. Fluid Mech.*, 2002.

[25] W H Tien. *3-D Particle Tracking Velocimetry: Development and Applications in Small Scale Flows*. PhD thesis, University of Washington, 2013.

[26] W H Tien, D Dabiri, and J Hove. Color-coded three-dimensional micro particle tracking velocimetry and application to micro backward-facing step flows. *Exp. Fluids*, 2014.

[27] F Pereira, M Gharib, D Dabiri, and D Modarress. Defocusing digital particle image velocimetry: a 3-component 3-dimensional DPIV measurement technique. application to bubbly flows. *Exp. Fluids*, 2000.

[28] C.E. Willert and M. Gharib. Digital particle image velocimetry. *Experiments in Fluids*, 10(4), Jan 1991.

[29] N. C. Angarita-Jaimes, M. G. Roca M., C. E. Towers, N. D. Read, and D. P. Towers. Algorithms for the automated analysis of cellular dynamics within living fungal colonies. *Cytometry*, 75A(9):768780, Sep 2009.

[30] G. L. Scott and H. C. Longuet-Higgins. An algorithm for associating the features of two images. *Proceedings of the Royal Society B: Biological Sciences*, 244(1309):2126, Apr 1991.

[31] RL Grothe and D Dabiri. An improved three-dimensional characterization of defocusing digital particle image velocimetry (DDPIV) based on a new imaging volume definition. *Meas. Sci. Technol.*, 2008.

[32] Illunis. *Illunis XMV-11000 USA OEM Q2-04 Product Brief*, 2004.

[33] D Schmitt. Development of a 3-D defocusing liquid crystal thermometry and velocimetry (3DDLCPTV) system. Master's thesis, University of Washington, 2007.

[34] L Kajitani and D Dabiri. A full three-dimensional characterization of defocusing digital particle image velocimetry. *Meas. Sci. Technol.*, 2005.

[35] Schneider-Kreuznach. *Makro-Symmar HM.*

[36] New Wave Research. *Solo PIV Nd:YAG Laser System Operator's Manual*, April 2006.

[37] C Tropea, A Yarin, and J Foss. *Springer Handbook of Experimental Fluid Mechanics*. Springer, 2007.

[38] Christopher E. Brennen. *Fundamentals of Multiphase Flow*. Cambridge University Press, 2005.

[39] C T Crowe, T R Troutt, and J N Chung. Numerical models for two-phase turbulent flows. *Annual Review of Fluid Mechanics*, 28(1):1143, Jan 1996.

[40] S. Elghobashi. On predicting particle-laden turbulent flows. *Appl. Sci. Res.*, 52(4):309329, Jun 1994.

[41] C.T. Crowe. *Multiphase Flow Handbook*. Mechanical and Aerospace Engineering Series. Taylor & Francis, 2005.

[42] K. Takehara and T. Etoh. A study on particle identification in PTV particle mask correlation method. *J Vis*, 1(3):313323, Sep 1999.

[43] Paul L. Rosin. Unimodal thresholding. *Pattern Recognition*, 34(11):20832096, Nov 2001.

[44] P.K Sahoo, S Soltani, and A.K.C Wong. A survey of thresholding techniques. *Computer Vision, Graphics, and Image Processing*, 41(2):233260, Feb 1988.

[45] C.A. Glasbey. An analysis of histogram-based thresholding algorithms. *CVGIP: Graphical Models and Image Processing*, 55(6):532537, Nov 1993.

[46] N Otsu. A threshold selection method from gray-level histograms. *IEEE T. Syst. Man. Cyb.*, 1979.

[47] Pascal Henry, Wei, Eric, Yuanhui, and Jean-Jacques. Volumetric monitoring and modeling of indoor air pollutant dispersion by the use of 3D particle tracking velocimetry. *Chemistry, Emission Control, Radioactive Pollution and Indoor Air Quality*, Jul 2011.

[48] N Ponchaut. *Part I: 3DPTV: Advances and Error Analysis Part II: Extension of Guderleys Solution for Converging Shock Waves*. PhD thesis, California Institute of Technology, 2005.

[49] J Duncan, D Dabiri, J Hove, and M Gharib. Universal outlier detection for particle image velocimetry (PIV) and particle tracking velocimetry (PTV) data. *Measurement Science and Technology*, 21(5):057002, Mar 2010.

[50] Y.-C. Lei, W.-H. Tien, J. Duncan, M. Paul, N. Ponchaut, C. Mouton, D. Dabiri, T. Rsgen, and J. Hove. A vision-based hybrid particle tracking velocimetry (PTV) technique using a modified cascade correlation peak-finding method. *Experiments in Fluids*, 53(5):12511268, Aug 2012.

[51] B. Luo and E.R. Hancock. Iterative procrustes alignment with the em algorithm. *Image and Vision Computing*, 20(5-6):377396, Apr 2002.

[52] J Graham, M Lee, N Malaya, R D Moser, G Eyink, and C Meneveau. *Turbulent Channel Flow Data Set*. Johns Hopkins University, Baltimore, MD 21218.

[53] Yi Li, Eric Perlman, Minping Wan, Yunke Yang, Charles Meneveau, Randal Burns, Shiyi Chen, Alexander Szalay, and Gregory Eyink. A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *TJOT*, 9, 2008.

[54] Eric Perlman, Randal Burns, Yi Li, and Charles Meneveau. Data exploration of turbulence simulations using a database cluster. *Proceedings of the 2007 ACM/IEEE conference on Supercomputing - SC 07*, 2007.

[55] Siddharth Talapatra and Joseph Katz. Three-dimensional velocity measurements in a roughness sublayer using microscopic digital in-line holography and optical index matching. *Measurement Science and Technology*, 24(2):024004, Dec 2012.

[56] Ashish Misra and D. I. Pullin. A vortex-based subgrid stress model for large-eddy simulation. *Physics of Fluids*, 9(8):2443, 1997.

[57] Daniele Schiavazzi, Filippo Coletti, Gianluca Iaccarino, and John K. Eaton. A matching pursuit approach to solenoidal filtering of three-dimensional velocity measurements. *Journal of Computational Physics*, 263:206221, Apr 2014.

[58] Ramgopal Sampath and S. R. Chakravarthy. Proper orthogonal and dynamic mode decompositions of time-resolved piv of confined backward-facing step flow. *Experiments in Fluids*, 55(9), Aug 2014.

[59] Ronald J Adrian and Jerry Westerweel. *Particle image velocimetry*. Number 30. Cambridge University Press, 2011.

[60] D Barnhart, R Adrian, and G Papen. Phase-conjugate holographic system for high-resolution particle-image velocimetry. *Appl. Opt.*, 1994.

[61] E C Graff and M Gharib. Performance prediction of point-based three-dimensional volumetric measurement systems. *Meas. Sci. Technol.*, 19(7):075403, Jun 2008.

[62] J Hong, J Katz, C Meneveau, and M Schultz. Coherent structures and associated subgrid-scale energy transfer in rough-wall turbulent channel flow. *J. Fluid Mech.*, 2012.

[63] H Huang, D Dabiri, and M Gharib. On errors of digital particle image velocimetry. *Meas. Sci. Technol.*, 1997.

[64] L Kajitani and D Dabiri. A full three-dimensional characterization of defocusing digital particle image velocimetry (erratum). *Meas. Sci. Technol.*, 2008.

[65] Nobuhide Kasagi and Akio Matsunaga. Three-dimensional particle-tracking velocimetry measurement of turbulence statistics and energy budget in a backward-facing step flow. *International Journal of Heat and Fluid Flow*, 16(6):477485, Dec 1995.

[66] J. Kostas, J. Soria, and M. Chong. Particle image velocimetry measurements of a backward-facing step flow. *Experiments in Fluids*, 33(6):838853, Dec 2002.

[67] O Labbé, P Sagaut, and E Montreuil. Large-Eddy simulation of heat transfer over a backward-facing step. *Numer. Heat Tr. A-Appl.*, 2002.

[68] M. Menacer, A. Aroussi, and C. Guendouz. Adaptive contrast enhancement method for typical histogram configuration. *Electronics Letters*, 35(15):1235, 1999.

[69] C Meneveau. Statistics of turbulence subgridscale stresses: Necessary conditions and experimental tests. *Phys. Fluids*, 1994.

[70] Charles Meneveau, Thomas S. Lund, and William H. Cabot. A lagrangian dynamic subgrid-scale model of turbulence. *J. Fluid Mech.*, 319(-1):353, Jul 1996.

[71] F Pereira and M Gharib. Defocusing digital particle image velocimetry and the three-dimensional characterization of two-phase flows. *Meas. Sci. Technol.*, 2002.

[72] M. Piirto, P. Saarenrinne, H. Eloranta, and R. Karvinen. Measuring turbulence energy with piv in a backward-facing step flow. *Experiments in Fluids*, 35(3):219236, Sep 2003.

[73] Ugo Piomelli, Parviz Moin, and Joel H. Ferziger. Model consistency in large eddy simulation of turbulent channel flows. *Physics of Fluids*, 31(7):1884, 1988.

[74] A K Prasad and K Jensen. Scheimpflug stereocamera for particle image velocimetry in liquid flows. *Appl. Opt.*, 1995.

[75] A Villegas, M Tawfik, Y Cheng, D C Bissell, S Pothos, and F J Diez. 3D-3C visualization of velocity gradients and coherent structures in free shear turbulent jets by TRSS-PIV and volumetric PIV. In *17th International Symposium on Applications of Laser Techniques to Fluid Mechanics*, Lisbon, Portugal, July 2014.

[76] M Virant and T Dracos. 3D PTV and its application on Lagrangian motion. *Meas. Sci. Technol.*, 1997.

[77] Wenquan Wang, Lixiang Zhang, and Yan Yan. Large eddy simulation of turbulent flow downstream of a backward-facing step. *Procedia Engineering*, 31:1622, Jan 2012.

[78] Hans Wengle, Andr Huppertz, Gnter Brwolff, and Gerd Janke. The manipulated transitional backward-facing step flow: an experimental and direct numerical simulation investigation. *European Journal of Mechanics - B/Fluids*, 20(1):2546, Jan 2001.

[79] J Westerweel. *Digital Particle Image Velocimetry*. PhD thesis, Technische Universiteit Delft, 1993.

[80] J Zhang, B Tao, and J Katz. Turbulent flow measurement in a square duct with hybrid holographic PIV. *Exp. Fluids*, 1997.

[81] V Zimin, H Meng, and F Hussain. Innovative holographic particle velocimeter: a multibeam technique. *Opt. Lett.*, 1993.

Appendix A

Manual for 3DPTV Software Suite

Author: Nicholas Dona, *University of Washington*

Original Programmers: Wei-Hsin Tien, PhD, *National Taiwan University of Science and Technology*

Joey Duncan, *University of Washington*

Updates: Nicholas Dona, *University of Washington*

A.1 Introduction

Hello! Congratulations on taking on the challenge of 3DPTV. This manual was designed to guide the use of the 3DPTV software suite, either step-by-step via GUI or in a fully-automated sense via a single driver program with appropriate settings files. Both methods are included in this guide, however, the GUI guide contains full descriptions of each parameter and process. The automated data reduction (without GUI) guide only contains usage instructions on the automated method itself, not any settings or processes contained within.

Compatibility Note: The full GUI process was designed for use with MATLAB® 2011-2013. The Particle Tracking and Outlier Removal routines, however, may only be used in MATLAB® 2011 due to the Delauney Tesselation feature.

A.2 Getting Started

The software featured in this software manual, including all GUI panels, subfunctions, and related functions are contained in a complete .zip file. First, extract the .zip file to the desired MATLAB® directory. Now, it is critical to add the extracted folder (with subfolders) to the MATLAB® path. Save the resulting path definition. For complete instructions on how to add a folder with subfolders to the MATLAB® path, see MATHWORKS® help files. This software suite expects image inputs in the form of 3 channels of double-precision intensity values ranging from 0 to 255 stored in a MATLAB® variable file with a '.mat' extension. If the setup contains three cameras, be sure to include each of the exposures in a single three-channel image prior to beginning of this process.

A.3 Calibration Process

What's Required

For this process, the GUI Panels 'ColorSepPanelV3.m,' 'PeakFindPanel.m,' and 'CalibrationPanelv1.m,' in Figures A.1 - A.16 below, are used to develop a full RBF calibration for 3DPTV data reduction. The image input expected by the GUI is in the form of double-precision intensity values ranging from 0 to 255. These values should be saved in a MATLAB® variable file with the extension '.mat.' Also, be sure to add the 3DPTV GUI package to the MATLAB® path as described above.

To begin the image processing portion of calibration, in MATLAB® , launch the .m file 'ColorSepPanelV3.m' from the folder:
'GUI_Micro3DPTV_AAStereo/GUI_MicroDDPIV_Full3Dmethod/GUI_panels'
The panel used for background averaging, subtraction, image pre-processing, color separation, and post-processing will be launched, Figure A.1. For ease of use, the ColorSepPanel is broken down into five subpanels corresponding to each of the relevant functions and may be seen closer in Figures A.2 - A.11

Optional: Averaging

During the calibration process, averaging is an optional feature that the user may opt to conduct. For processing of stationary calibration images, simple averaging may be necessary to remove non-stationary particles or objects that could interfere with calibration. The averaging process is identical to the one used in 3DPTV data processing, however, when used in calibration, the resulting averaged "background" image is *not* subtracted from all images in the set. Instead, the resulting averaged image for each plane is the single image that is used for calibration.

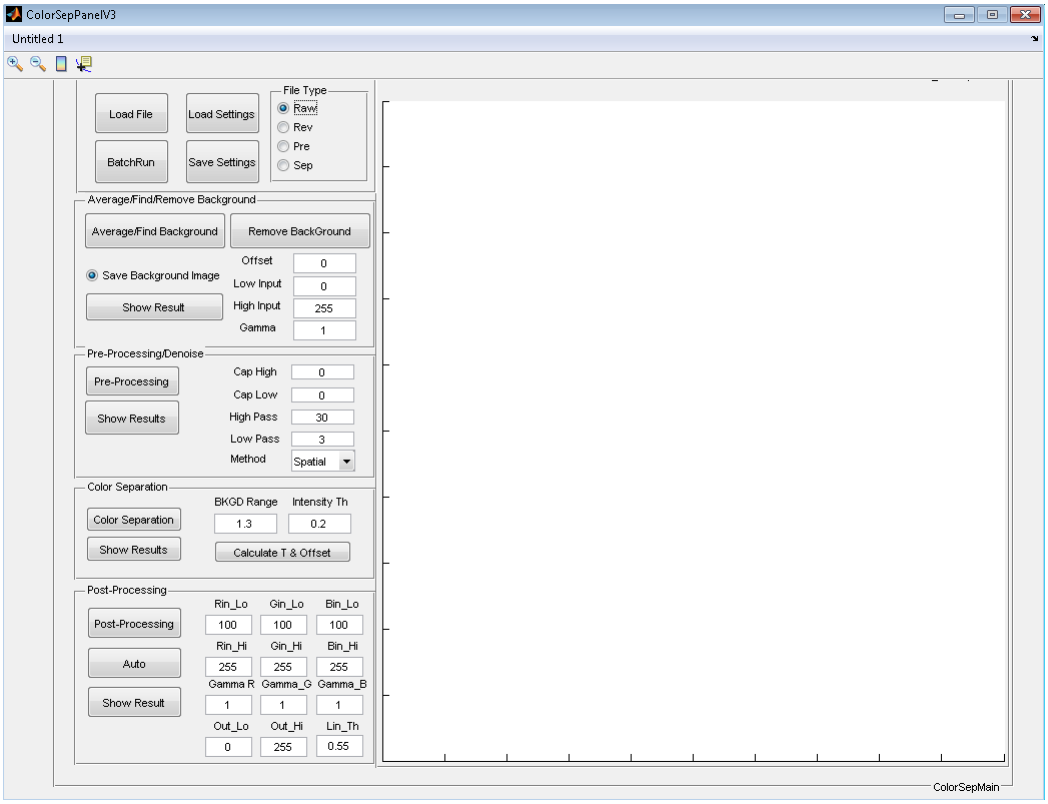


Figure A.1: Full GUI for 'ColorSepPanelV3,' used in background averaging, background subtraction, pre-processing, color separation, and post-processing.

Begin by launching the 'ColorSepPanelV3.m' file in MATLAB® as mentioned above. In the General Subpanel, Figure A.2, insure under "File Type," that the "Raw" option is selected. This option will insure that the file explorer GUI will not sift out the file extensions of your raw calibration images. Now, select the "Load File" button and ctrl+click each file or shift+click to select multiple calibration images taken for the same calibration plane. Once all images corresponding to the desired calibration plane have been selected, press "Open." A "Load File" dialogue box should appear confirming the number of files that have been selected. Insure this number matches the number of images taken for the current plane then acknowledge

the dialogue box by clicking "Okay." Now, the selected images have been stored in the GUI's memory and will be used in the current processing step.

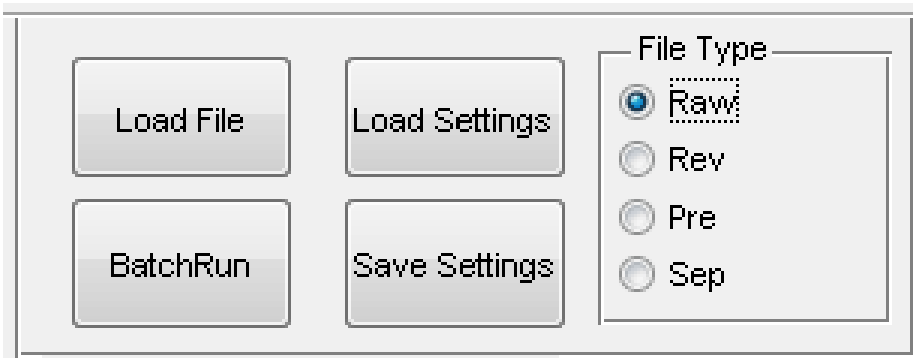


Figure A.2: Color Separation General Subpanel

Now, within the Background Averaging Subpanel, Figure A.3, click the button labeled, "Show Result," if you would like to see the resulting image in the large display area to the right of the GUI. When activated, the button will change to a blue color. Otherwise, leave the button unselected and go directly to the Background Finding step by clicking the button labeled, "Average/Find Background." Upon clicking this button, a "Save" dialogue window will appear. Save the resulting averaged image with a title corresponding to the current calibration plane.

The settings in this subpanel, including "Offset," "Low Input," "High Input," and "Gamma" may be left at default values of 0, 0, 255, and 1, respectively. If, upon further examination, the resulting averaged calibration images need contrast adjustment, these fields may be further-tuned for optimal contrast.

The "Remove BackGround" button is not to be used in this situation. This button is used to remove the resulting averaged image from other images in data reduction.

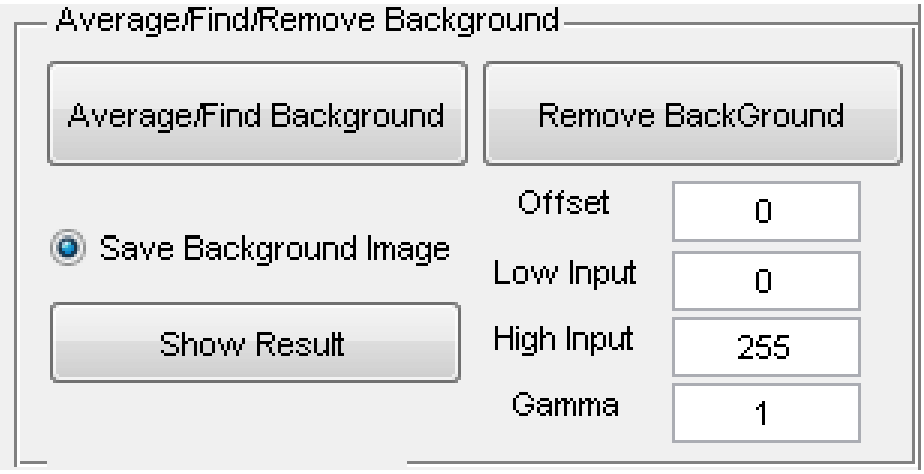


Figure A.3: Color Separation Background Processing Subpanel

This is not necessary in the calibration process. The image Averaging process described above may be repeated for each plane of the calibration until all planes have been successfully processed. Again, this step is optional and is suggested for situations with liquid media where floating particles of dust or dirt may interfere with calibration. The resulting images from this step, if completed correctly, will have the convention, "XXXX_bkgd.mat," where 'XXXX' was the desired name specified earlier in this step.

Pre-Processing

If the optional averaging step described above was accomplished, use the resulting 'XXXX_bkgd.mat' images for this Pre-Processing step. Otherwise, use the individual 'XXXX.mat' images taken during calibration.

To begin the Pre-Processing step, check that the "Raw" toggle button is selected in the File Type portion of the General Subpanel, seen in Figure A.2. This option

will insure that the file explorer GUI will not sift out the file extensions of your raw calibration images. Now, click the "Load File" button to load the File Explorer GUI. Within the GUI, select the desired image corresponding to each plane taken in the calibration and click the "Open" button. A "Load File" dialogue box should appear confirming the number of files that have been selected. Insure this number matches the number of images taken for the current plane then acknowledge the dialogue box by clicking "Okay." Now, the selected images have been stored in the GUI's memory and will be used in the current processing step.

Now, within the Pre-Processing/Denoise Subpanel, Figure A.4, click the button labeled, "Show Results," if you would like to see the resulting image in the large display area to the right of the GUI. When activated, the button will change to a blue color. Otherwise, leave the button unselected and go directly to the Pre-Processing step by clicking the button labeled, "Pre-Processing."

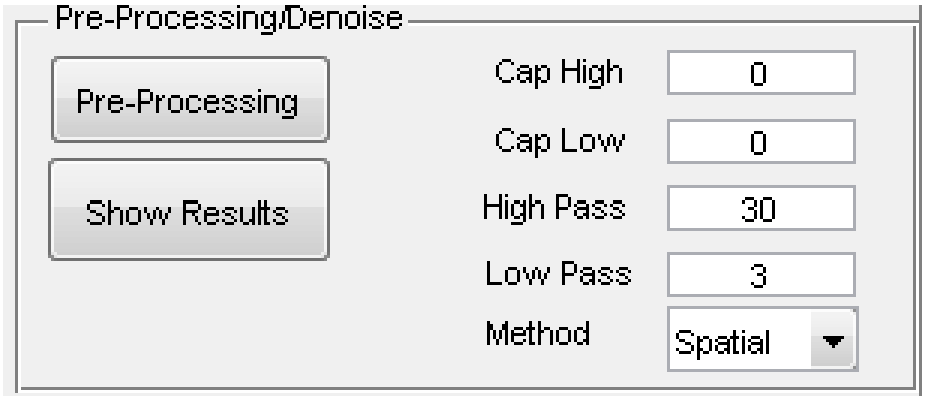


Figure A.4: Color Separation Pre-Processing/Denoise Subpanel

The resulting images from this step, if completed correctly, will have the convention, "XXXX_pre.mat," where 'XXXX' was the full original file name prior to this

step.

The default values for "Cap High," "Cap Low," "High Pass," "Low Pass," and "Method" should be set to 0, 0, 30, 3, and 'Spatial,' respectively in most cases. The "Cap High" and "Cap Low" settings may be used if clipping of high or low intensity values is desired. For example, an input of 5 in each of these fields will result in an image that has the top and bottom 5% of its intensity values clipped. The range is then stretched to fit the original histogram's range of values. High Pass and Low Pass values refer to the spatial periods between which a band pass filtration will occur. Other denoising methods available in this GUI include Freq and FIR. These methods are neither used nor described in this manual.

Optional: Color Separation

When it's Needed:

Color separation is required when a single, color CCD in conjunction with a three-pinhole plate is used for capturing 3DPTV images. The resulting issue will be in the form of a spectral overlapping between the color channels in the images. If identical peaks of varying intensities create any level of aliasing on multiple color channels, then color separation is absolutely necessary. This is due to the fact that the pinhole plate's color filters may not perfectly match the CCD's RGB filters. An example of this cross-talk and post-color separation results may be seen in Figure A.5.

In the backward-facing step flow turbulence study, three separate CCDs were used. Thus, color channels were assigned ex post facto and no color separation was required. Therefore, no explanation of color separation theory may be found in the aforementioned document. For a full description of these methods, see the Tien PhD

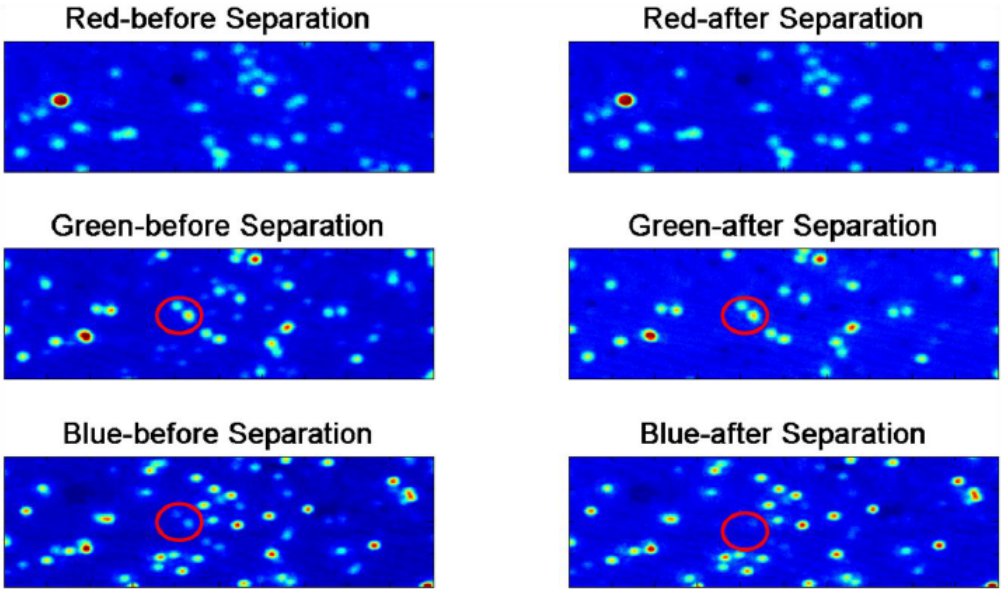


Figure A.5: This is an example of when color separation is needed; note G and B channel interplay.[25]

dissertation.[25]

How it's Done:

Use the resulting 'XXXX_pre.mat' images from the Pre-Processing Step for this step. To begin the Color Separation step, check that the "Pre" toggle button is selected in the File Type portion of the General Subpanel, seen in Figure A.2. This option will insure that the file explorer GUI will display only pre-processed images. Now, click the "Load File" button to load the File Explorer GUI. Within the GUI, select the desired pre-processed images corresponding to the calibration planes and click the "Open" button. A "Load File" dialogue box should appear confirming the number of files that have been selected. Insure this number matches the number of images

taken for the current plane then acknowledge the dialogue box by clicking "Okay." Now, the selected images have been stored in the GUI's memory and will be used in the current processing step.

Now, within the Color Separation Subpanel, Figure A.6, click the button labeled, "Show Results," if you would like to see the resulting color space rotation tensor, **T**, intensity offsets, and the resulting image and channel intensities in a new window. When activated, the button will change to a blue color. Otherwise, leave the button unselected. Calculate the **T** and Offset values by first pressing the button labeled, "Calculate T & Offset." Upon selecting this button, it will turn light blue. Then, with this button selected, press the button labeled "Color Separation." Pressing the Color Separation button with the Calculate option selected will calculate the **T** and offset values for all selected images and store them in the GUI memory. Wait for the **T** and offsets to appear in the MATLAB® Command Window. This will not apply the transform yet.

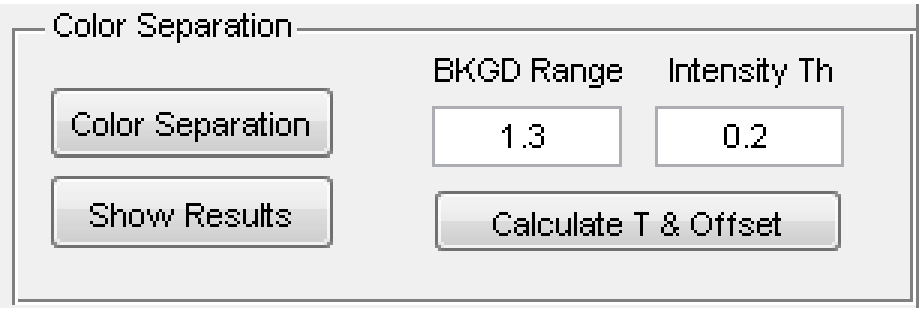


Figure A.6: Color Separation Subpanel

Now that the rotation tensor and offsets are calculated, click the "Calculate T & Offset" button once more to deselect it. This will return it to the normal gray color. Now, without the Calculate option selected, click the button labeled "Color

Separation" once more. This will now apply the stored transform and offsets and save the resulting images with the added end extension '-sep.'

The default values for "BKGD Range" and "Intensity Threshold" are 1.3 and 0.2, respectively. These values typically do not need adjusting, however, if the results for Color Separation do not appear satisfactory, they may be adjusted. "BKGD Range" relates to the background saturation range. The Intensity Threshold relates to the threshold above which pixels in other image channels will be considered for cross correlation.

Optional: Line Removal

When It's Needed:

While the typical calibration plate is a transparent glass plate of regularly-spaced etched dots, sometimes more complex targets are needed. In the case of the 3DPTV water tunnel turbulence study, a grid and dot pattern was used to precisely align the cameras to remove distortions and rotations. With this added grid pattern, lines are now present in the calibration plane images. These lines may be seen as 1-D horizontal and vertical Gaussian distributions surrounding each dot. With the addition of these Gaussian lines, the peak finding algorithm would be flooded with lines of false positive particle matches. Thus, a computer vision technique was developed to remove these lines prior to further processing.

If the calibration images contain a similar or identical glass grid dot plate, this line removal technique is necessary.

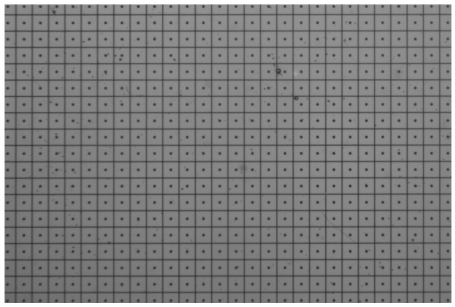
How it's Done:

In the folder labeled 'FunctionsNotInGUI,' find the m-script labeled 'LineRemovalv6.m.' Open the m-file and become familiar with the code. Each segment of code should be adequately labeled with a description of the process and some basic theory. The Line Removal function expects '-pre' inputs from the above pre-processing step, though the function may be easily altered to expect '-sep' inputs in the event that color separation was required. It expects three input images for each calibration plane, one from each CCD channel labeled with the string patterns, 'TL,' 'TR,' or 'BC' somewhere in each file name. The assumption is that these images are grayscale with a light background and dark grid dot pattern. The program will take the image complement, adjust the image histogram in an adaptive manner, execute the computer vision routine to remove the lines, complete tophat filtering of the image, and save a single, 3-component RGB image for each calibration plane with the extension '-line.mat.' Figure A.7 shows a before and after example of this Line Removal Process.

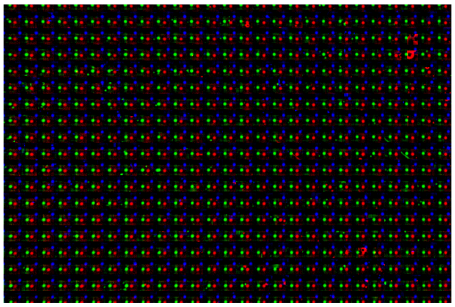
Now, find the two cells of code that have labels beginning with "USER INPUT:," and examine their contents. These are the only two cells that require any sort of user input.

The first cell, Figure A.8 relates to the line thickness of the vertical and horizontal lines present in the image. Open a sample calibration plane image and measure the thickness of the horizontal and vertical lines. Input the thickness values for the x-line and y-line thickness parameters in the first cell.

The second cell for user input, Figure A.9 is an empirical adjustment for each image channel, red, green, and blue, or from each individual CCD. This empirical adjustment determines how many lines of resolution from each camera input should



(a) Sample region of target plate in a raw calibration image from one of three cameras



(b) Same image as Figure 4.1a after post-processing, line-removal, and 3-camera combination

Figure A.7: Side-by-side comparison of raw and post-processed sample calibration images

```
31      %% UESR INPUT: Set line thickness erase parameters
32
33      %These settings adjust for line thickness on grid target plate. After the
34      %SVD Identification and erase procedure, the program goes back and refines
35      %the black lines of resolution by erasing additional xerase lines to the
36      %left/right of vertical lines and additional yerase lines to the top/bottom
37      %of horizontal lines.
38
39 -    settings.xerase=10; %Pixels to erase to the left and right of vertical lines
40 -    settings.yerase=10; %Pixels to erase to the top and bottom of horizontal lines
41
```

Figure A.8: First User Input for Line Removal Program: vertical and horizontal line thickness

be set to zero in the vertical and horizontal orientation. To get a baseline datum for these values, comment out the for loop on line 68 and comment in the for loop on line 69, both shown in Figure A.10. This will insure the program only executes the process on a single complete image. When the process is finished, examine the

first image. If there are still vertical or horizontal grid lines remaining in any given channel of the image, increase the line removal values corresponding to that channel and orientation. If lines of dots begin to disappear, however, decrease the value of lines removed corresponding to that channel and orientation. Continue the process until optimal values have been chosen for the first image.

```
102        %% USER INPUT:Set Number of PC1 Resolution Lines to Erase in Each Channel
103        %The following values must be adjusted for resolution of images being
104        %processed. Example: Values of approximately 425 vertical lines and 270
105        %horizontal lines were used for 4008x2672 images. These values are
106        %defined for each channel of red, green, and blue portion of images.
107        %Recommend processing 1 image to assess results. If not enough lines
108        %erased, increase number of terms in that color and direction. If dots
109        %begin to disappear, decrease number of terms in that color and
110        %direction.
111 -      if strfind(imagelist(imi,:),'TR')   %If Top Right image (green),
112            %Sets number of vertical lines of resolution to be deleted in Top
113            %Right image channel
114 -          xterms=425;
115            %Sets number of Horizontal lines of resolution to be deleted in Top
116            %Right image channel
117 -          yterms=270;
118 -      elseif strfind(imagelist(imi,:),'TL')  %Same for Top Left (red)
119 -          xterms=375;
120 -          yterms=250;
121 -      elseif strfind(imagelist(imi,:),'BC')  %Same for Bottom Center (blue)
122 -          xterms=375;
123 -          yterms=250;
124 -      else    % Returns error if naming convention doesn't match above logic
125 -          error('LineRemoval:NameConv',['Check Naming Convention for',...
126                '3-Camera System']) %%check name convention of file list
127 -      end
128
129
130
131
132
133
134
```

Figure A.9: Second User Input for Line Removal Program: number of vertical and horizontal lines of resolution to be removed for each channel

```
68 -  for imi=1:length(imagelist)     %Use this line to process all images.
69    % for imi=1:3        %Use this line to assess results of 1 image.
```

Figure A.10: Full Analysis or Debug For Loop Selection

Once the above user inputs have been defined, use the original for loop on line 68 and comment out the for loop on line 69. Now when the program runs, it will run for all images that contain a '-pre' in the file name.

Post-Processing

Use the resulting 'XXXX_pre.mat,' 'XXXXsep.mat,' or 'XXXX_-line.mat' images from the previous step for this Post-Processing step. To begin the Post-Processing step, in the File Type portion of the General Subpanel, seen in Figure A.2, check that the appropriate toggle button is selected. For pre-processed images, select the "Pre" toggle button. For color separated images, select the "Sep toggle button. For line-removed images, select the "Raw" toggle button, as the GUI will not show these images due to the "-line" extension. Now, click the "Load File" button to load the File Explorer GUI. Within the GUI, select the desired images corresponding to the calibration planes and click the "Open" button. A "Load File" dialogue box should appear confirming the number of files that have been selected. Insure this number matches the number of images taken for the current plane then acknowledge the dialogue box by clicking "Okay." Now, the selected images have been stored in the GUI's memory and will be used in the current processing step.

Now, within the Post-Processing Subpanel, Figure A.11, click the button labeled, "Show Results," if you would like to see the resulting image in the large display area to the right of the GUI. When activated, the button will change to a blue color. Otherwise, leave the button unselected. This post-processing step serves to adjust

the contrast in each channel of the image independently. For typical use, the "Auto" setting is sufficient. Activate the "Auto" setting by clicking on the "Auto" button to highlight it blue. Selecting this setting instructs the GUI to ignore all user-defined fields on the right side of the subpanel, except for the 'Lin_Th' field. Instead, the GUI will automatically select the appropriate values for each channel of each image based upon histogram values and the 'Lin_Th' setting. Finally, click on the "Post-Processing" button to begin the Post-Processing routine for all of the selected images. The resulting contrast-adjusted images will be saved with an added "-1.mat" file extension.

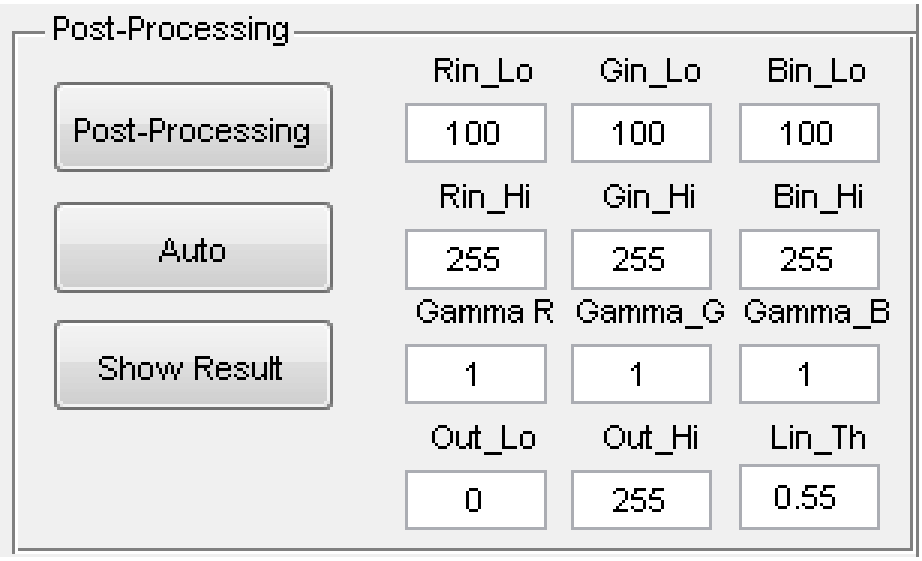


Figure A.11: Color Separation Post-Processing Subpanel

If, after examining the results, the automatic setting does not provide an adequate contrast adjustment to the images, user-defined settings may be used by turning off the "Auto" button and defining a value in each field. These values are analogous to

the values one would define for the MATLAB® function "imadjust," which is essentially a stretching function for intensity values of the image. The input values may be defined separately for each color channel for the images, but will be stretched to the same output range for all channels. The Gamma values describe how each image histogram will be skewed by the stretching, while a value of '1' indicates no change. Finally, 'Lin_Th' is the image histogram threshold used in determining the rest of the settings automatically. This is the only field used during Post-Processing with the 'Auto' option selected.

Peak Finding

Now that the images have been fully processed, it's time to determine the dot locations with subpixel accuracy within the calibration planes in image coordinates (pixel values in the x and y directions). To begin the peak finding process, run the file for the GUI panel called 'PeakFindPanel.m.' This will launch the GUI Panel in Figure A.12

First and foremost, defining parameters accurately in the Settings Subpanel, Figure A.13, is critical to the performance of the Peak Finding by CCM Algorithm. While this guide provides a brief overview of the settings, it is important to fully understand the theory behind the process and relevant settings by referring to the Tien PhD Dissertation[25] as well as the Dona Masters Thesis.

Peak Finding Settings

Each paragraph within this section corresponds to and explains a field within the CCM_Parameters Subpanel, Figure A.13.

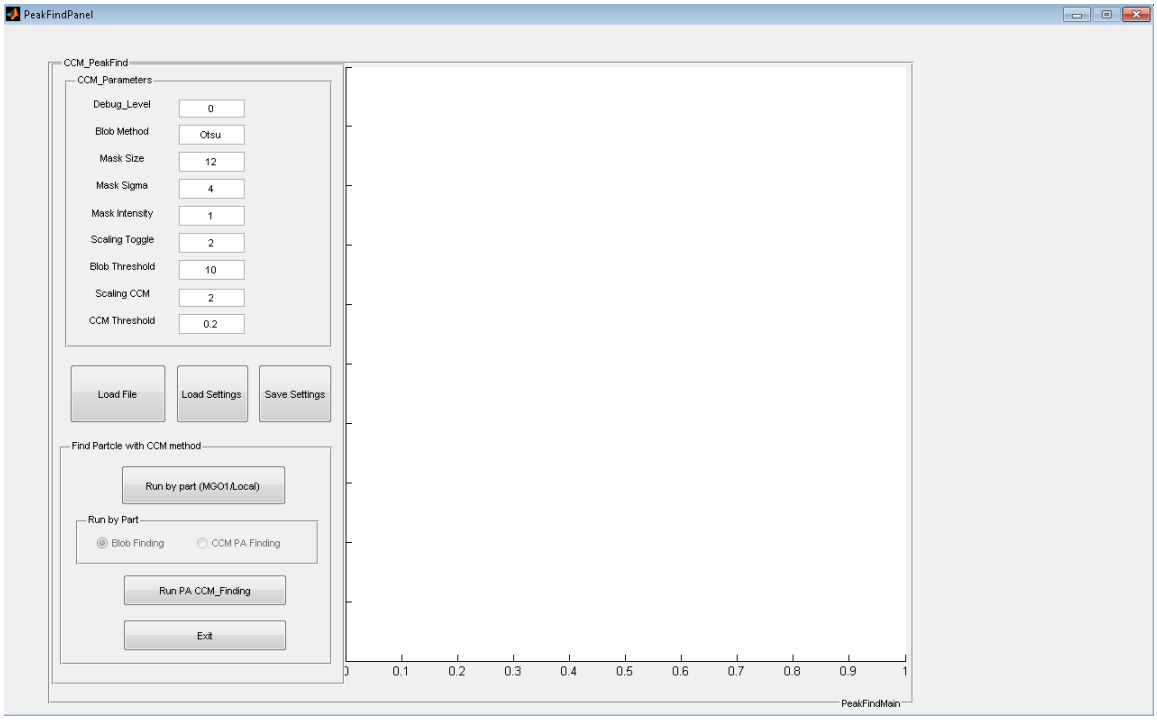


Figure A.12: Full GUI for 'PeakFindPanel,' used to find calibration plate dot locations in image coordinates.

Note: The parameter values featured in Figures A.12 and A.13 are not default values. These values correspond to the values used throughout most of the Backward-Facing Step Flow study in the Dona Thesis.

Debug_Level The debug level varies the verbosity of the output from the peak finding step. Number of plots displayed vastly increases with debug level and number of particles in the image used for debugging. Use caution when employing a Debug Level of 2 as the number of MATLAB® plots can easily exceed 100. Types of plots shown include the initial Gaussian mask used in the cross-correlation calculation as

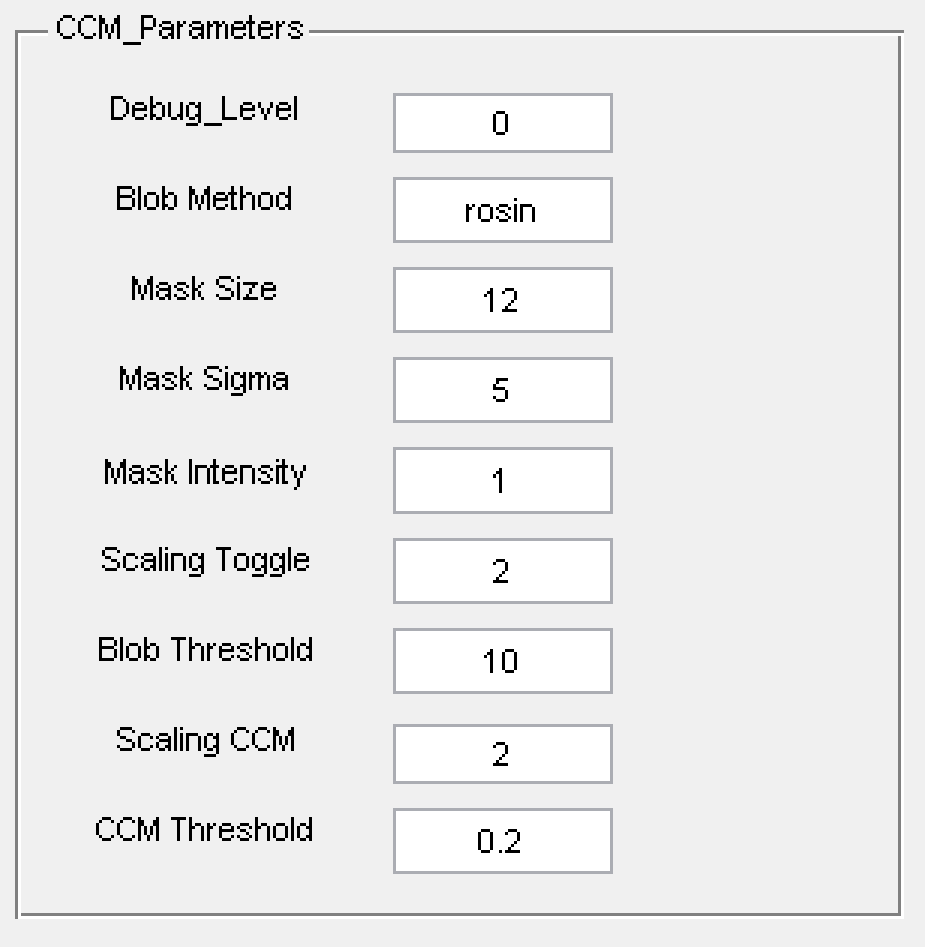


Figure A.13: Settings Subpanel for the Peak Finding GUI Panel

well as each successive cascading iteration of decreasing radius for least squares fitting. It is recommended that a very small portion of a full data image is used with a Debug Level of 2.

Blob Method Blob Method is the method by which the Peak Finding process separates potential particle or particle cluster images from the background noise

of the image. Each option is a different thresholding technique to select the background/foreground cutoff point. Prior to selecting which Blob Thresholding Method to use, it is critical to check the image intensity histogram distribution. There are three options contained within this GUI, 'otsu,' 'rosin,' and 'gray'. Otsu is the typical method used for this process, however Otsu's method assumes a bimodal histogram in the particle image and selects a grayscale intensity value that maximizes the interclass variance.[46] Image histograms do not always have bimodal distributions, however. Therefore, a unimodal global thresholding method called Rosin Thresholding was also added to the GUI.[43] Finally the Gray option is a hybrid method utilizing Otsu's method as well as the MATLAB® morphological command 'imclose' to isolate blob images. When selecting the Blob Method, be sure to type the lowercase word 'otsu,' 'rosin,' or 'gray' verbatim.

Mask Size The Mask Size refers to the initial overall size of the Gaussian Mask used in the Cascade Cross-Correlation Method. For instance, a Mask Size of 12 corresponds to an overall 12x12 size for the 2-dimensional Gaussian image of a model particle. It does not, however, necessarily correspond to the size of the particle contained within the mask. It simply represents the overall size of the mask image.

Mask Sigma Mask Sigma refers to the radius of the mask particle (in pixel values). This setting should correspond to the average particle radius seen in the calibration plane or particle image being processed. The GUI then generates a mask that is smaller than the average particle radius by a factor of 3.67, as per the guidance of Adrian and Westerweel.[59]

Mask Intensity The value for Mask Intensity is a setting used to guide the algorithm in formulating masks with the correct intensity values. The input value for this field ranges from 0 to 1. Within the Peak Finding by CCM algorithm, a unique mask is generated on a per-blob basis. First, each pixel that composes a blob is normalized by the maximum intensity found within the blob, resulting in a blob of intensity values ranging from 0 to 1. Then, a Gaussian mask is developed with a maximum intensity equivalent to the Mask Intensity input defined by the user. Due to this normalization technique, this parameter is typically approximately equal to 1.

Scaling Toggle This toggles the level of scaling accomplished in the peak finding process.

Blob Threshold The Blob Threshold Value is the minimum intensity value that a pixel should have in order to be considered a part of a blob. Visually inspect a representative particle image and determine a conservative minimum value above which a pixel is likely to be more than background noise. The algorithm will consider this user input, however, it will determine an exact threshold automatically.

Scaling CCM This alters the level of scale variation in the modified CCM process.

CCM Threshold This is the cross-correlation threshold above which the Cascade Cross-Correlation Method will identify a potential match. The method will still optimize the number of peaks and fit for each peak, but will only consider candidates

above this minimum coefficient.

Running the Peak Finding Algorithm

Once all of the above settings have been determined, the Peak Finding by Cascade Cross-Correlation Method may be executed. First, click on the button at the center of the GUI Panel, Figure A.12, labeled "Load File." Within the file explorer, select all desired '-1.mat' files. Note that the file explorer GUI launched in this step will NOT display only the '-1.mat' files. Once the desired files have been selected, press the "Open" button. As in the previous GUI, a confirmation dialogue window will appear confirming the number of images selected. Insure this number matches the number of files desired and acknowledge the notification.

While testing or fine-tuning Peak Finding settings, it is recommended that the settings combination used in each trial is saved for future reference. Saving the settings contained in the CCM_Parameters fields is simple. Click on the button labeled "Save Settings" and choose a unique file name for the settings file. The file will be saved 'XXXX_pksettings.mat,' where "XXXX" is the user-specified name. This settings file is a '.mat' file containing a structured array easily opened and manipulated without the GUI, via the MATLAB® Command Window.

To load a previously-specified and saved settings file, simply click on the button labeled "Load Settings." The File Explorer GUI will automatically display only settings files with the extension '_pksettings.mat.' Loading a settings file will result in a change of the user-defined fields in the CCM_Parameters Subpanel, Figure A.13, to reflect those contained in the file. Verify that the settings are correct.

Once the desired settings and image files have been selected, it's time to execute the Peak Finding algorithm using the Execution Subpanel, Figure A.14. The most

common method of execution is to run the entire Peak Finding Method, a combination of initial Blob Finding followed by Cascade Cross-Correlation analysis of each blob, in a continuous, automated series. This is accomplished by simply pressing the button labeled "Run PA CCM_Finding."

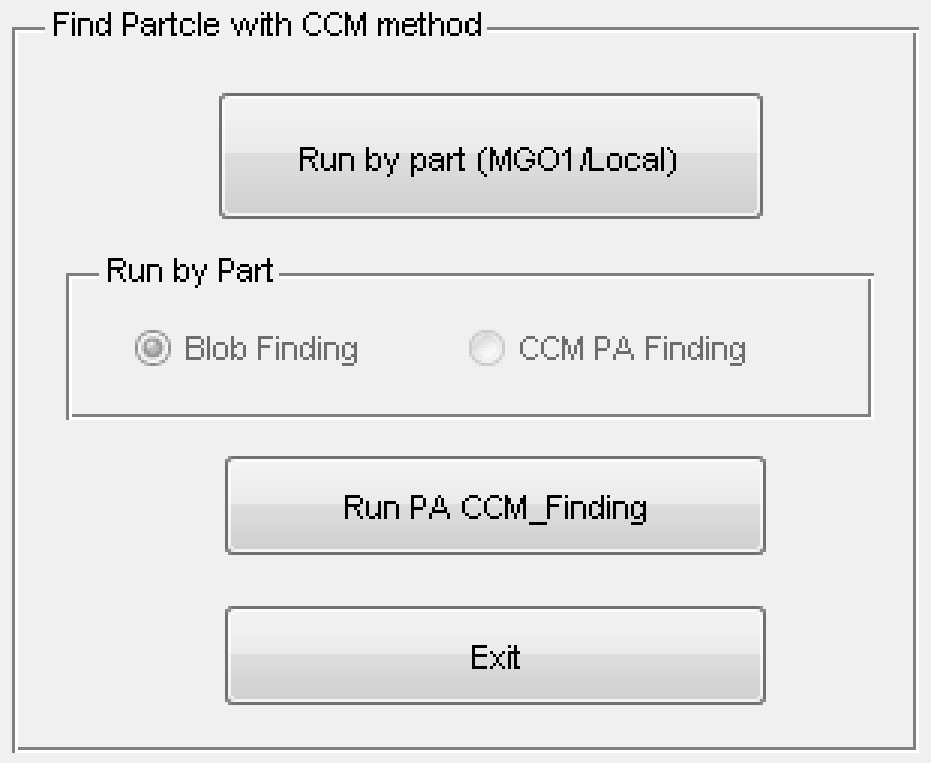


Figure A.14: Execution Subpanel for the Peak Finding GUI Panel

The alternative to the continuous method is to run each process contained in this step separately. This may be advantageous if the user would like to debug the process using the result of Blobs found directly. To accomplish this, first press the button labeled "Run by part (MGO1/Local)." Pressing this button will enter the GUI Panel into the by-part mode. In order to exit this mode, the GUI must be exited

and re-launched. Entering the by-part mode allows the user to first execute the Blob Finding process by selecting the "Blob Finding" toggle button then clicking the "Run PA CCM_Finding" button.

The Blob Finding step, when run in by-part mode, will output three files labeled 'XXXX_-1_blob1.mat,' 'XXXX_-1_blob2.mat,' and 'XXXX_-1_blob3.mat.' These files contain a structure of every blob found in the entire image. Each file corresponds to a single color channel of the original image. Each blob within the overall structure of each channel contains a binary image of the blob as well as image coordinates describing the location of the top left corner of the blob image within the overall image.

Next, the CCM PA Finding process may also be run independently. In order for this process to run, the three blob files corresponding to each image file, described above, must be present in the same folder as the selected image file, 'XXXX-1.mat.' In order to run the CCM PA Finding, specify the desired image files, as described above, if you have not already done so. Then, enter the run by-part mode if the GUI is not already in this mode. Select the CCM PA Finding toggle button then click the "Run PA CCM_Finding" button.

The final results of the Peak Finding step, whether run continuously or by part, are stored in a file with the convention 'XXXX-1_PADATA.mat.' These files contain a PADATA variable structure containing 3 independent structures corresponding to each of the image channels. Each of these channel structures contains a list of particle locations in that channel in x and y image coordinates.

These particle locations may be overlaid on the original image using the MATLAB® script, 'plotPADATAv1.m' contained in the software suite's .zip folder. In order to use this script, place a copy in the same directory as the desired images and their corresponding '_PADATA.mat' files. Then, simply run the 'plotPADATAv1.m' file.

Sample results of this plotting routine are shown in Figure A.15.

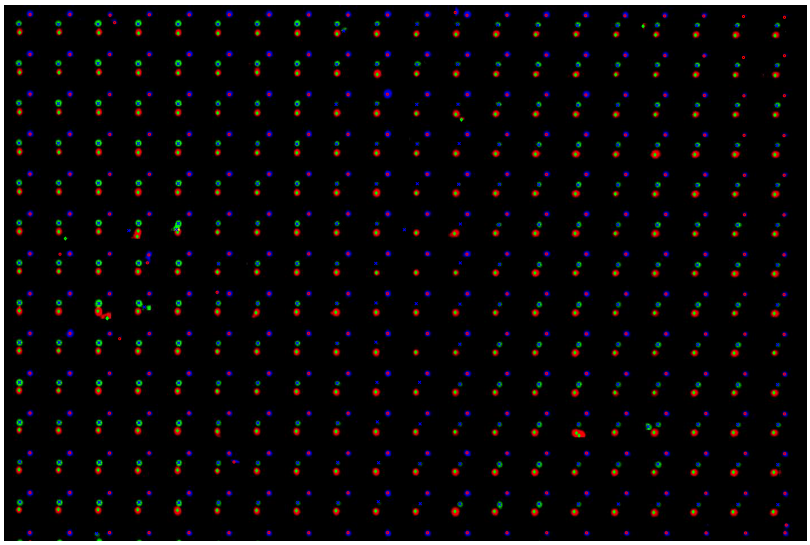


Figure A.15: Results of the Peak Finding by CCM Algorithm overlaid on a small portion of the original image using 'plotPADATAv1.m.' For visibility even in cases of saturation, red peak locations are plotted with green markers, green peaks with blue markers, and blue peaks with red markers.

Building RBF Calibration

Initial Setup

The Calibration step develops the mapping of xy (image) locations of calibration plate dot patterns from each of the three channels to XYZ (physical) locations. The first half of the calibration step involves registering and aligning the mapping points in each plane. The second half of the process involves matching the mapping points and

building the RBF from the volumetric grid data.

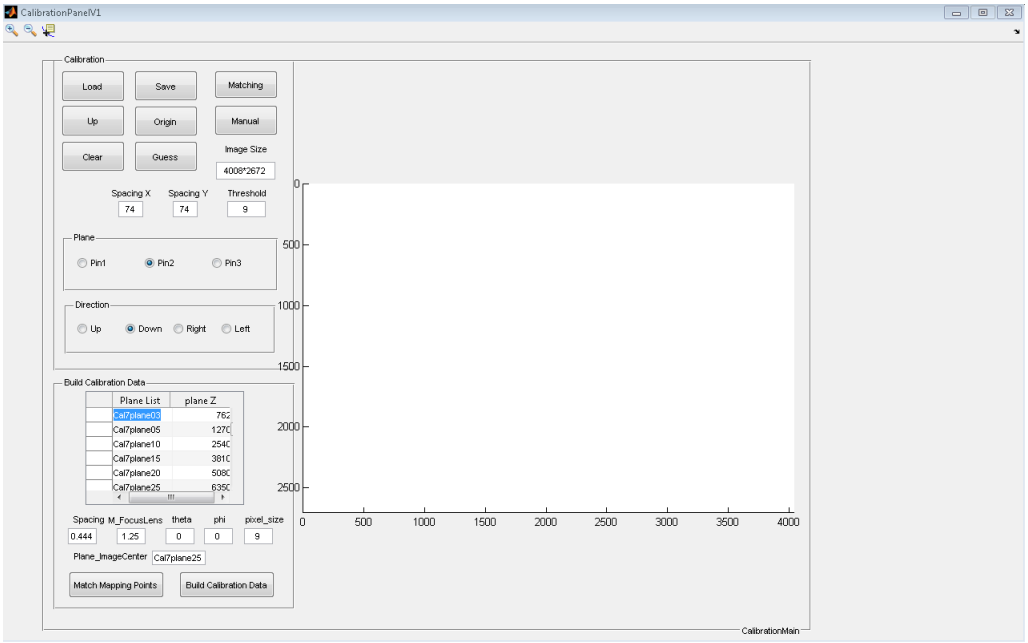


Figure A.16: Full GUI for 'CalibrationPanel,' used to calculate the mapping to transform image coordinates to physical coordinates and build an RBF interpolation.

Begin by launching the 'CalibrationPanelV1.m' file in MATLAB® . This will display the Calibration Panel, Figure A.16. To start, the user input fields in the Buttons Subpanel, Figure A.17, must be populated and the "Plane" and "Direction" buttons must be properly selected.

First, within the Subpanel in Figure A.17, input the resolution of the calibration images being used in the format '$h * v$' and press enter. It is important to note that the plot range of the viewing field to the right side of the GUI will not change limits until a '_PADATA.mat' file of the appropriate resolution is loaded.

Next, populate the fields labeled "Spacing X" and "Spacing Y" with the appro-

priate x-direction and y-direction dot spacing in pixel coordinate system with pixel units. It is important to note that this measure will change as the target plate translates through the experimental VOI. Because of this effect, the user should choose an appropriate value based on the current plane alignment between the GUI projection and actual peaks found. This may be adjusted by trial and error. These settings allow the GUI to overlay a standard, theoretical grid based on pixel measurements over each plane input by the user. The final field in this subpanel, "Threshold," is the pixel threshold distance within which the GUI will automatically identify a dot location based upon the user-defined intervals and the location of the other dots in the plane. Again, this should be adjusted by way of trial and error on a plane-by-plane basis in order to achieve the most accurate "Guess" by the GUI algorithm.

Now, the "Plane" and "Direction" buttons must be selected for the appropriate image channel and flow direction. In the "Plane" Subpanel, selecting "Pin1" will only show the peaks found within the first image channel (red). Selecting "Pin2" would show green peaks with a loaded image, and "Pin3" would show blue.

The "Direction" Subpanel indicates the flow direction of the experiment relative to the calibration plane dots showing in the viewing window to the right of the panel. This establishes the coordinate axes system for the calibration and should not be changed once established.

Finally, the settings shown in Figure A.30 are not relevant until the "Match Mapping Points" step and the "Build Calibration Data" step. They may be left at the default values during the peak-to-grid matching process.

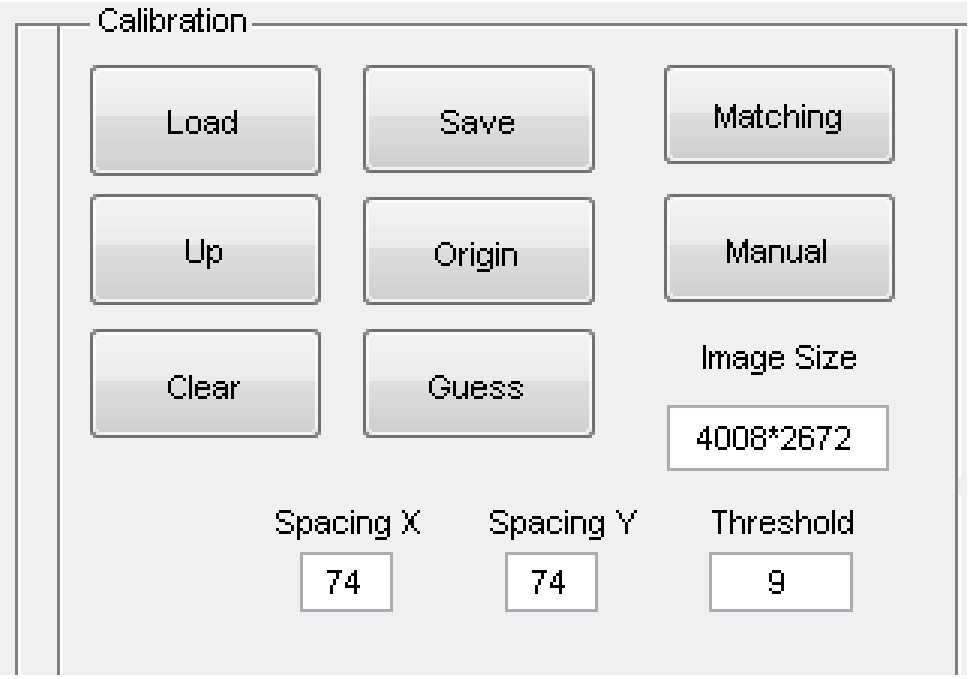


Figure A.17: Buttons Subpanel for the Calibration Panel with resolution, dot spacing, and threshold options

Peak-to-Grid Matching

This is the longest and most user-involved portion of the entire calibration process. The process described in this section must be repeated for each image channel, pins 1-3, of each image plane in the calibration range.

First, change the MATLAB® path to the directory containing the '_PADATA.mat' files corresponding to the calibration plane images. Insure that each of the planes is labeled systematically with a simple title such as "plane000_PADATA.mat" or "plane100_PADATA.mat." Do not include additional underscores in the file names. This will insure correct organization of the planes' coordinate data into the calibration's Excel® file; the prefix to "_PADATA.mat" will become the names of the sheets.

Click on the “Load” button in the Buttons Subpanel, Figure A.17. This will launch a file explorer window showing the calibration plane ‘_PADATA.mat’ files. Select and open the file for the first of the calibration planes to be matched to the grid. The order in which the planes and color channels are processed is inconsequential; they all must be mapped in order to build the calibration. The peaks shown in the viewing window will correspond to the pin selected in the Plane Subpanel. They will be plotted with the appropriate channel color as shown for the green plane, “Pin2,” in Figure A.18.

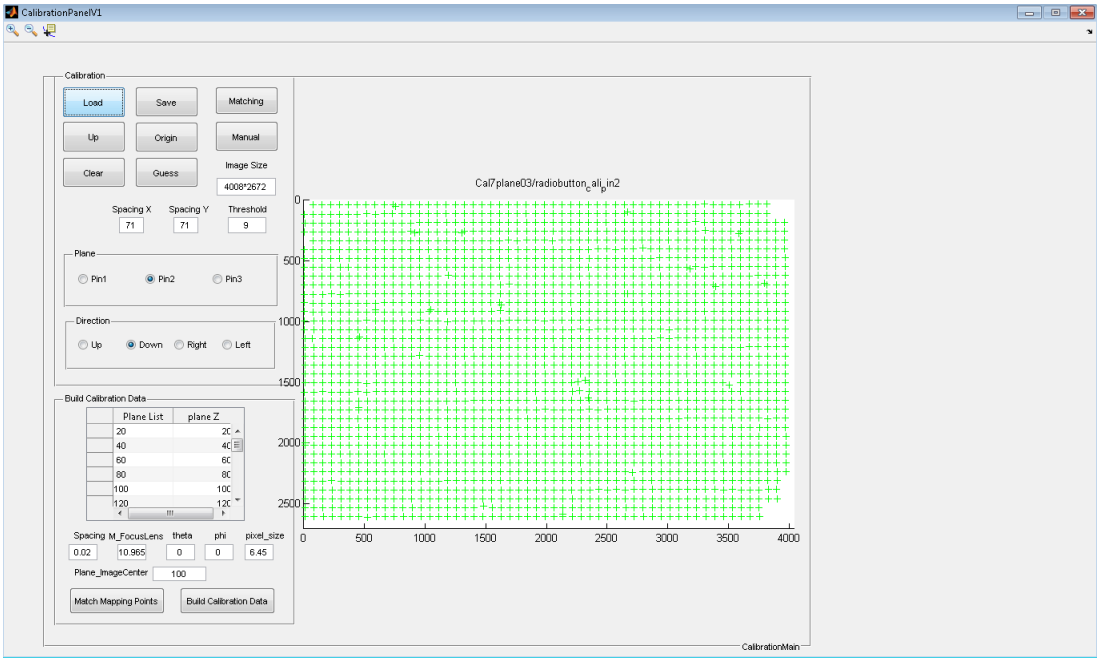


Figure A.18: Green peaks plotted with “Pin2” selected.

An origin must be chosen to anchor the peak-to-grid matching process. This origin must correspond to a specific point in physical space that will remain unchanged and identifiable across all planes and color channels. The easiest way to initially select this point is to define it relative to a missing dot on the calibration plate that manifests

itself in all colors and planes in the calibration. For example, find a missing point on the calibration grid that is missing in all color channels and planes and choose the dot located three spaces to the right and five spaces upwards. It is important to define this dot in physical coordinates, as the image coordinates will dilate and shift across the Z-depth of the calibration volume. Use this dot as the origin in all planes and channels.

Once selected, the origin must be chosen for the GUI in each plane and color channel. To do so, first click on the "Origin" button. This will change the color of the plotted peaks to black. This is normal and expected. Click within the viewing panel until the mouse pointer turns into a cross-hair selector. Use the crosshair selector to highlight the desired origin as shown in Figure A.19. Once selected, press the "Return" or "Enter" button on the keyboard. This will register the origin into the GUI's memory.

Next the selection process must be repeated for the dot directly above the origin. To define the "Up" dot, first click the "Up" button. Again, click within the viewing panel until the mouse pointer turns into a cross-hair selector. Use the crosshair selector to highlight the dot directly upward of the origin as shown in Figure A.20. Once selected, press the "Return" or "Enter" button on the keyboard. This will help register the correct orientation of the plane axes into the GUI's memory.

Once the origin and up-direction are registered with the GUI, press the "Guess" button. The "Guess" process is where the GUI overlays a theoretical grid emanating from the origin based on the spacing entered earlier. The process also involves automatically providing a "Guess" association of which of the plotted peaks correspond to the theoretical grid. The GUI uses the "Threshold" value input earlier to make

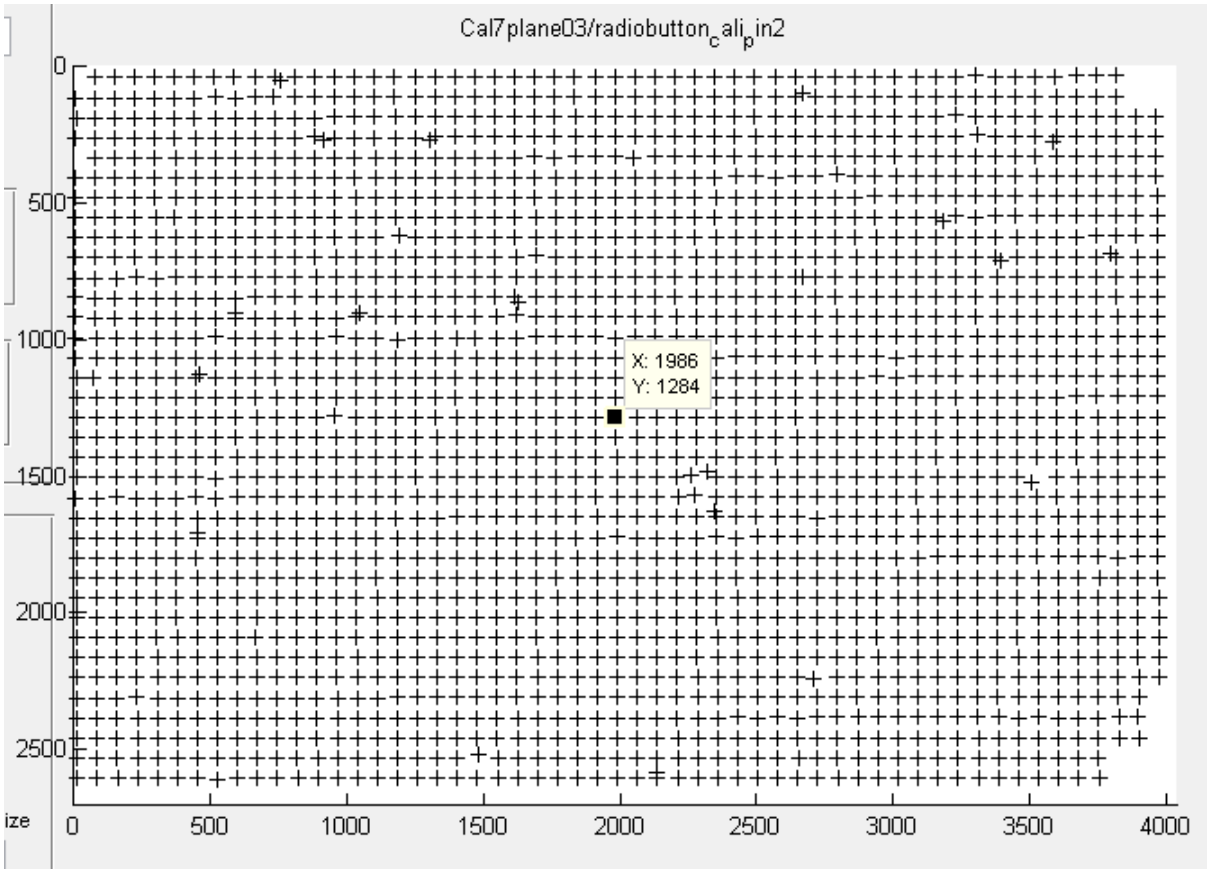


Figure A.19: Calibration window with origin selected

this decision. If a peak is within the Threshold value, it is considered associated with a grid point. If it is not within the threshold value, it is considered an outlier point. There are markings that correspond to each of these categories. Sample results of the "Guess" Process are shown in Figures A.21 and A.22.

These results include a new figure, Figure A.21, specifically showing suspected outliers found during the "Guess" process. These outliers are highlighted with a magenta diamond and a green ray trace from the origin. Additionally, the peaks plotted in the work space of the Calibration Panel will look different. First, there will be

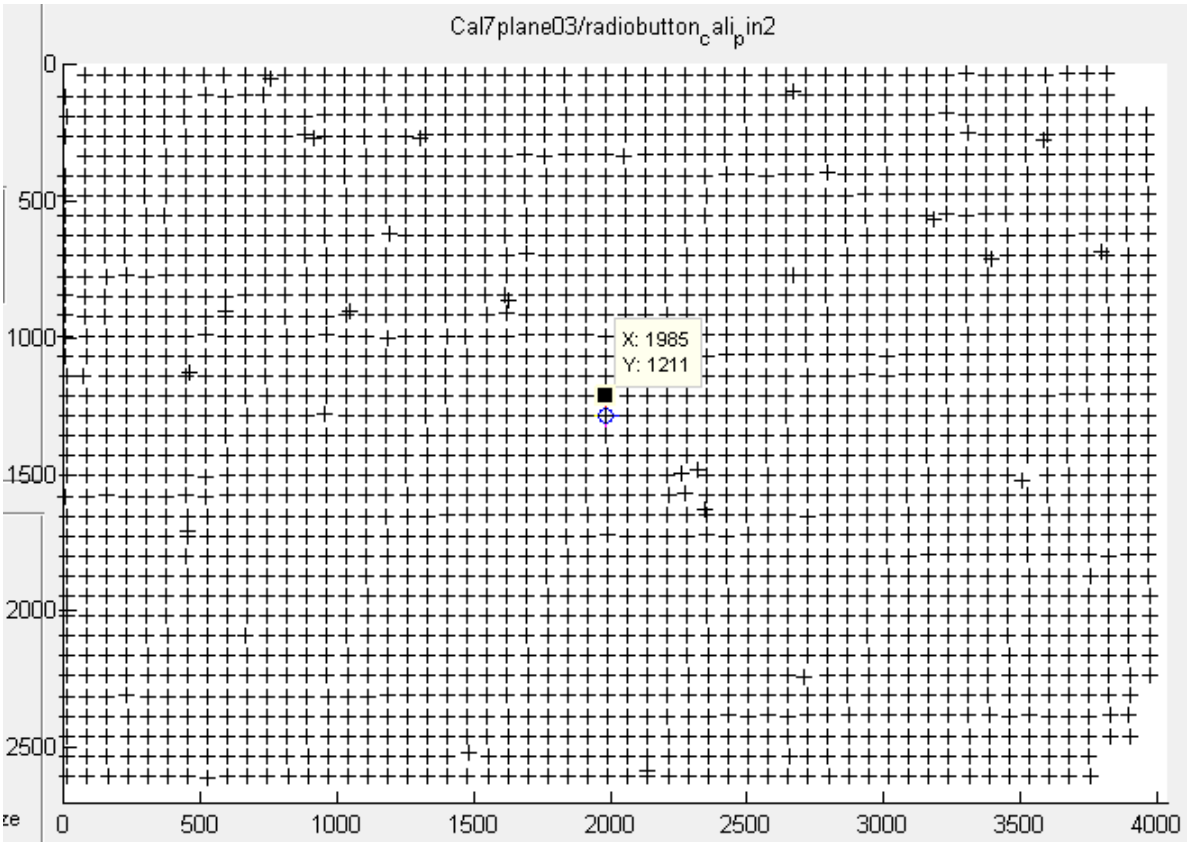


Figure A.20: Calibration window with "Up" selected

a theoretical grid overlaid consisting of blue x's plotted using the user-defined grid spacing in the x and y directions. The peaks from the original '_PADATA.mat' file will still be plotted with black +'s. Finally, peaks that the GUI associates with the theoretical grid, via the Threshold setting, will be outlined with a red diamond. These plotting points are defined in the legend in Figure A.23.

Take a minute to carefully inspect the results of the "Guess" process in both the new figure and the Calibration Panel working space. It should become clear that the "Guess" process is not a perfect one. It should be apparent that many peaks

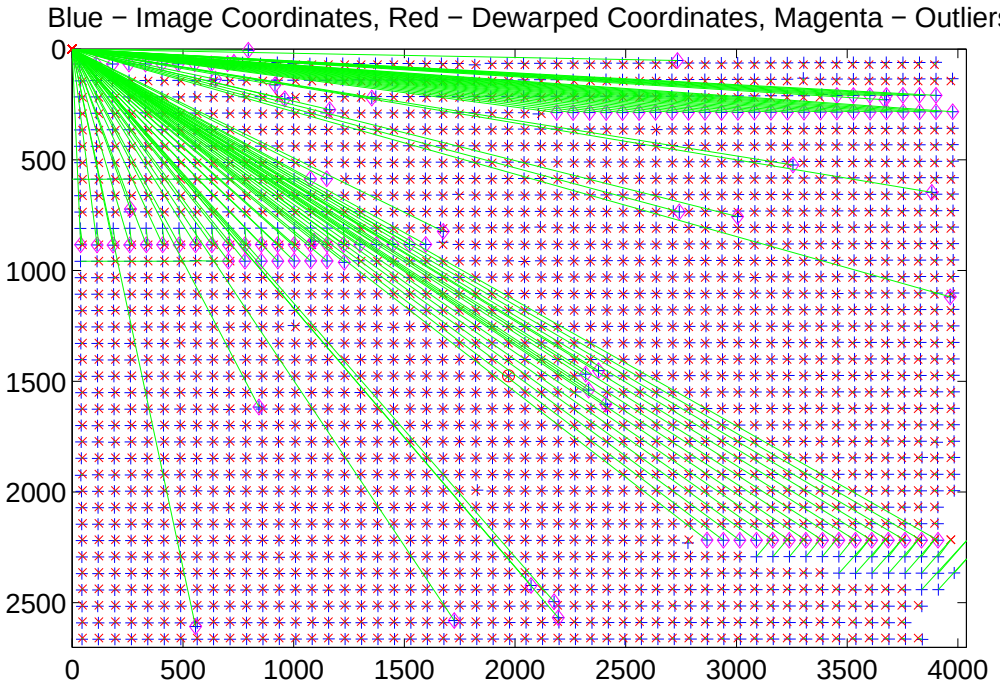


Figure A.21: Sample outlier plot results from the calibration "Guess" process.

that appear to be perfectly valid elements (proper spacing and alignment) of the theoretical grid do not have red diamonds. It should also be clear that there are red diamonds around black +'s that should not be considered part of the grid (slightly out of alignment or improper spacing). This is due to the fact that the GUI works off of the principle of relative thresholding when guessing for peak association. This means that in the element-by-element analysis, if the next black + is really an outlier, but is shifted by less than the Threshold value, it will be associated with the grid. But, if the next black + following the falsely-matched outlier is a good data point, it may

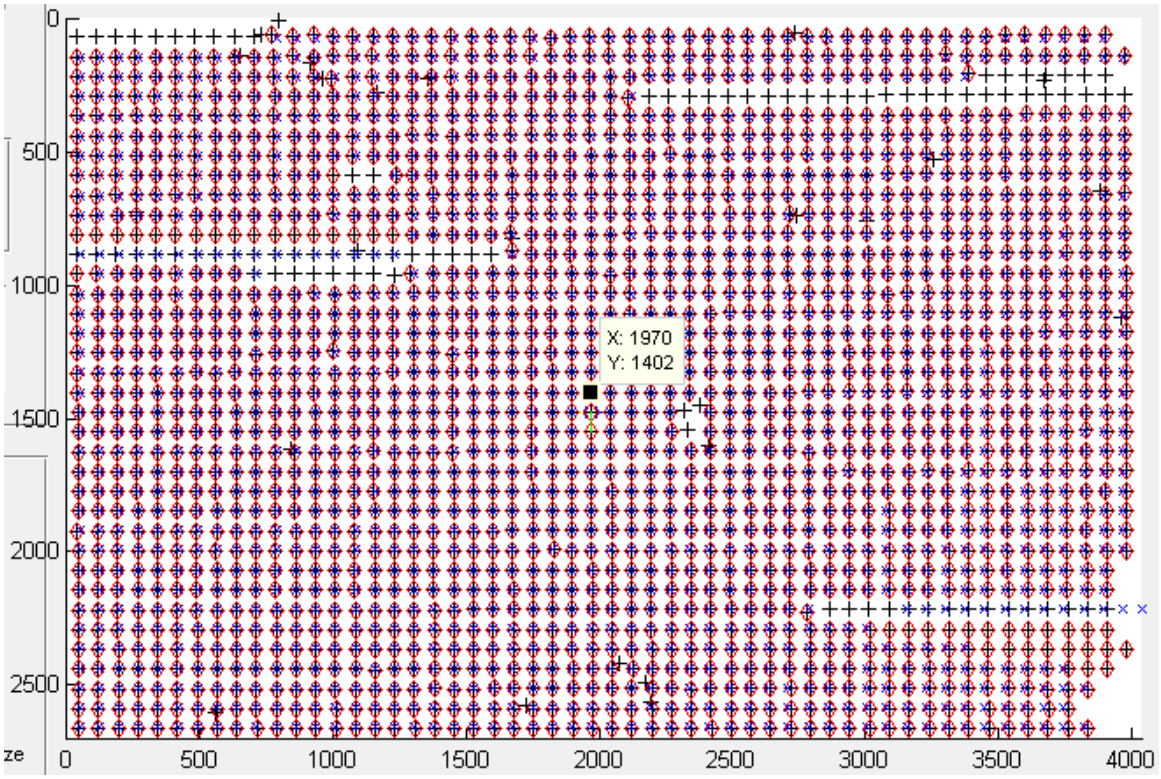


Figure A.22: Sample work space panel results from the calibration "Guess" process.

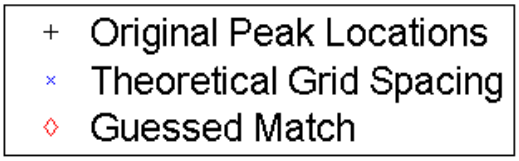


Figure A.23: Calibration Panel "Guess" results legend for workspace results.

not be within the Threshold spacing from the previous, bad point. These outliers "derail" the element-by-element guessing process. Due to this fact, false positives must be removed, and true negatives must be re-instated for re-matching. This is the nature of the following "Manual" process.

Begin the manual adjustment process by clicking on the button labeled "Manual." This changes the Calibration Panel working space by adding some additional visual aids to the plot, as seen in Figure A.24.

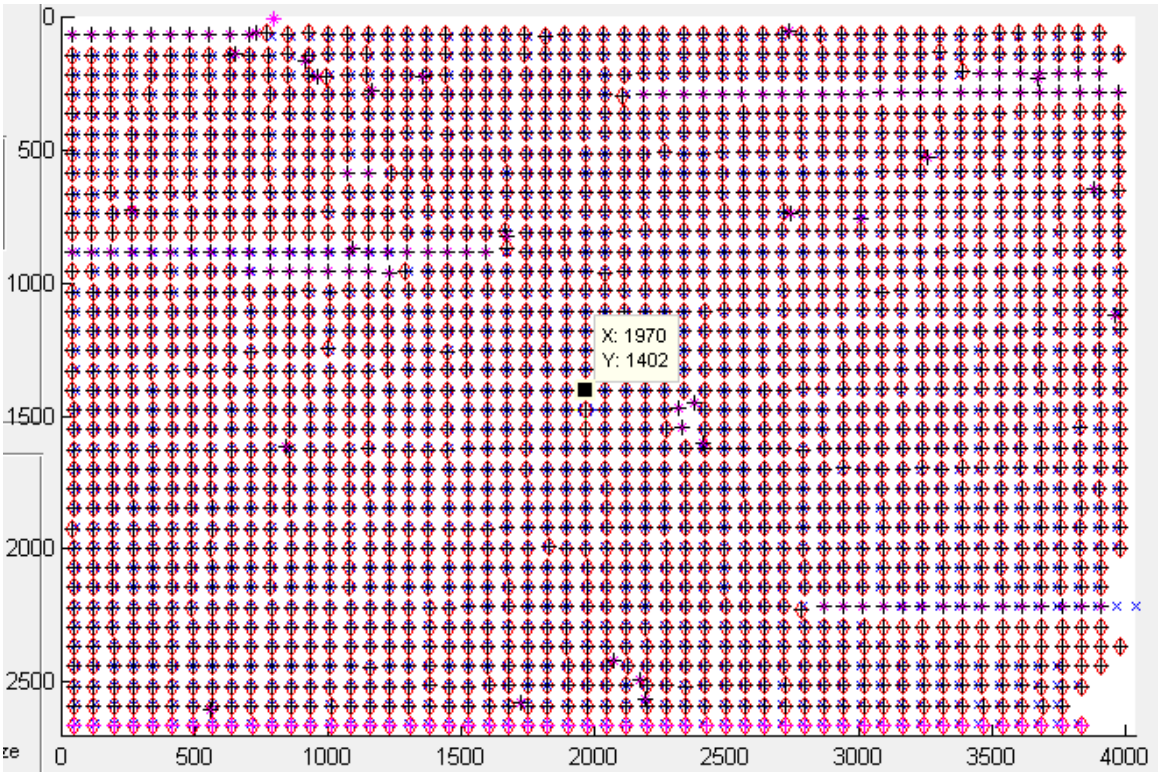


Figure A.24: Work space panel after entering "Manual" mode.

Upon entering manual mode, suspected outliers will now have an additional magenta x marking them. Also, rows of edge peaks that may be out of alignment are now plotted with magenta +'s and highlighted with magenta diamonds instead of red ones. These peaks may or may not be accurate and should be checked by the user. In order to add peaks back into the matching process or to remove a falsely matched peak, select the desired peaks to toggle the status. Selecting a suspected outlier will

overlay a yellow x on the black +. This indicates the peak has been added back for consideration in the next round of guessing. Selecting a matched peak to be removed will result in a small purple dot added to the center of the diamond. This indicates the peak will be removed as an outlier and will not be considered in the next "Guess" iteration.

Selecting peaks can be tricky. All selections must be made from top left to bottom right. First, click the mouse pointer slightly off-frame then drag the pointer into the work space. This will change the pointer to a crosshair. Click and drag again with the crosshair so that the mouse returns to a standard pointer once more. Once these two steps are accomplished, the mouse is now temporarily in select mode for the next click and drag cycle. Click and drag top left to bottom right over any peaks that you would like to either add back into the matching process or remove as an outlier to change their status. Repeat this process to enter select mode for each desired section until the plane is appropriately fixed. Results showing the previously mentioned "de-railing" issue before and after manual correction are shown in Figures A.25 and A.26, respectively.

To zoom in on the affected peaks in order to get a better view, click the zoom magnifying glass in the top left corner of the GUI to enter and exit zoom mode. Repeat the manual selection and removal process until the entire plane has been fixed. Once this has been accomplished, click on the "Matching" button to execute the changes you have highlighted over the course of the "Manual" step. If in the rare instance that there were no manual changes to be made, simply go directly to the "Matching" step to remove any correctly identified outliers. Results of the matching step in the GUI works pace are shown in Figure A.27.

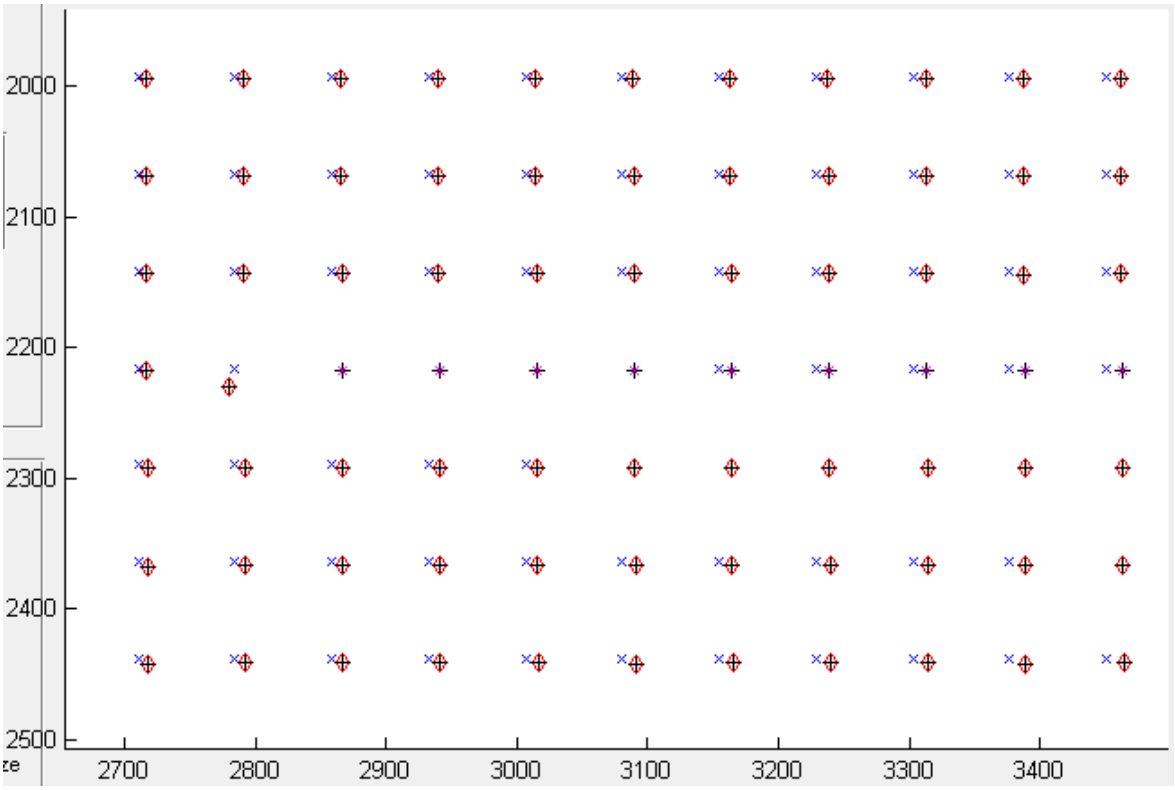


Figure A.25: "Derailed" peaks before correction. This highlights where the "Guess" process got off track and could no longer associate true grid peaks correctly.

Once the matching process has completed, a notification of "Outlier removed!" will appear in the MATLAB® Command Window. Once this appears, click the "Guess" button again to observe the new status of the calibration plane peak-to-grid matches in the new Figure that is launched. A correctly-matched plane should be well-aligned and consistently-spaced with limited holes due to outliers. An example of a fully fixed plane may be seen in Figure A.28.

The "Manual" association process is not a perfect one. Sometimes this process must be repeated several times until outliers are properly removed or added back in. Additionally, sometimes the GUI removes falsely identified outliers before the user

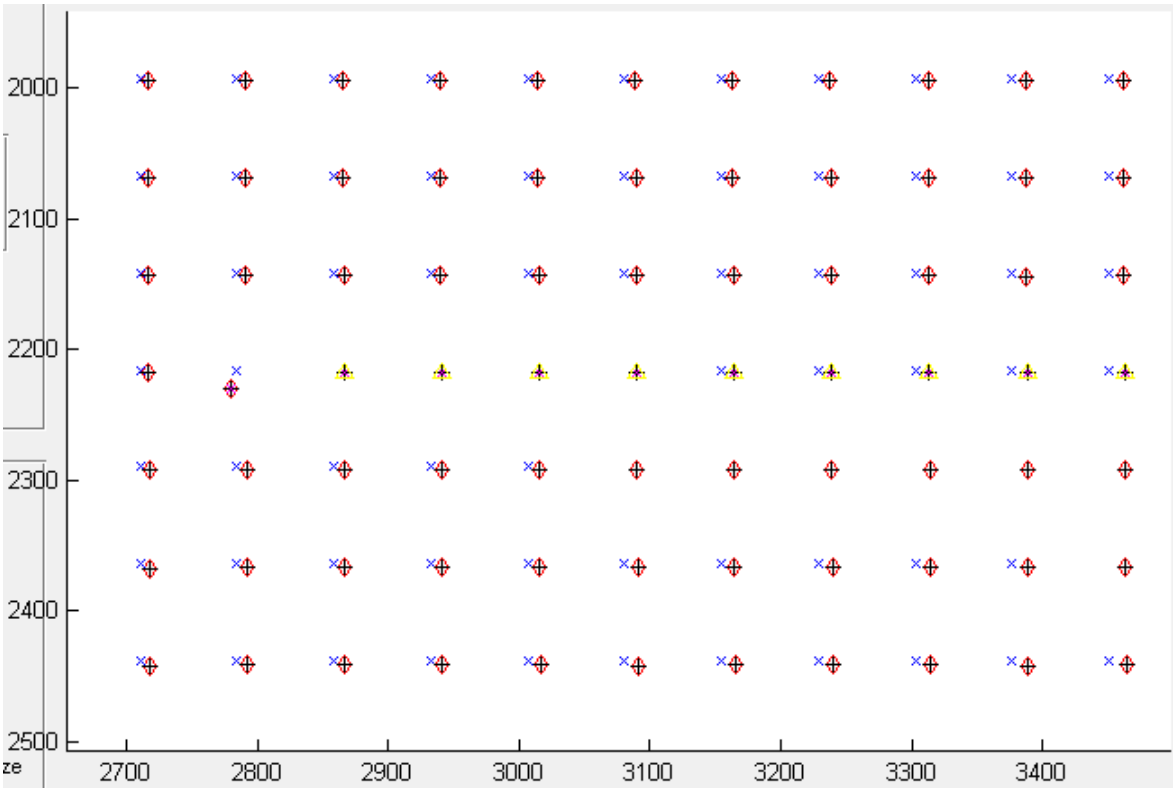


Figure A.26: Peaks that have been added back for the next round of guessing (yellow x's). The removed outlier that caused the derailing was also removed (magenta dot).

gets a chance to add them back in. If this is the case, all peaks must be cleared and reloaded, and the process must be started from the beginning for the current channel of the current plane.

Once a plane that resembles the one shown in Figure A.28 is accomplished, click on the "Save" button to save the calibration plane grid and origin coordinates. If it is the first plane that has been completed, select a name for the .xls file to be saved as. For all subsequent planes, select to save and confirm overwriting this .xls file. The data will simply be appended to the appropriate column range for each new color channel. For each new calibration plane, a new sheet will be added, named as the

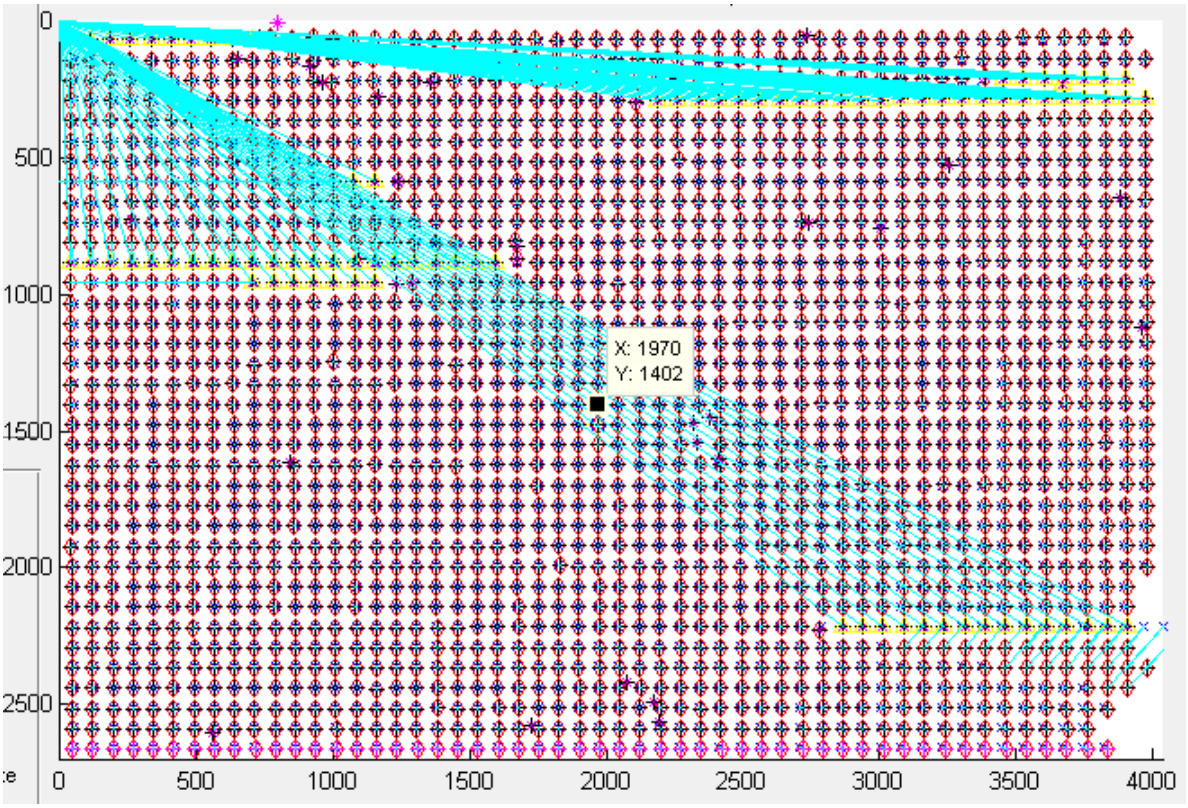


Figure A.27: Example manually fixed and matched calibration plane. Blue rays emanating from the origin indicate outliers that have been manually added back in for the next guess attempt.

text preceding the '_PADATA.mat' file extension.

Once the peak-to-grid matching has been accomplished for all planes and all color channels, it is important to analyze the validity of the matching. This may be accomplished by plotting the origin locations of each channel and plane. The origin locations should match the geometry of the system's epipolar lines in 3-Dimensions, as shown in Figure A.29. If there are any planes or channels that do not fall on the trend lines, they should be re-assessed with the origin more carefully assigned. Varying levels of verification may be accomplished using the MATLAB® codes 'showo-

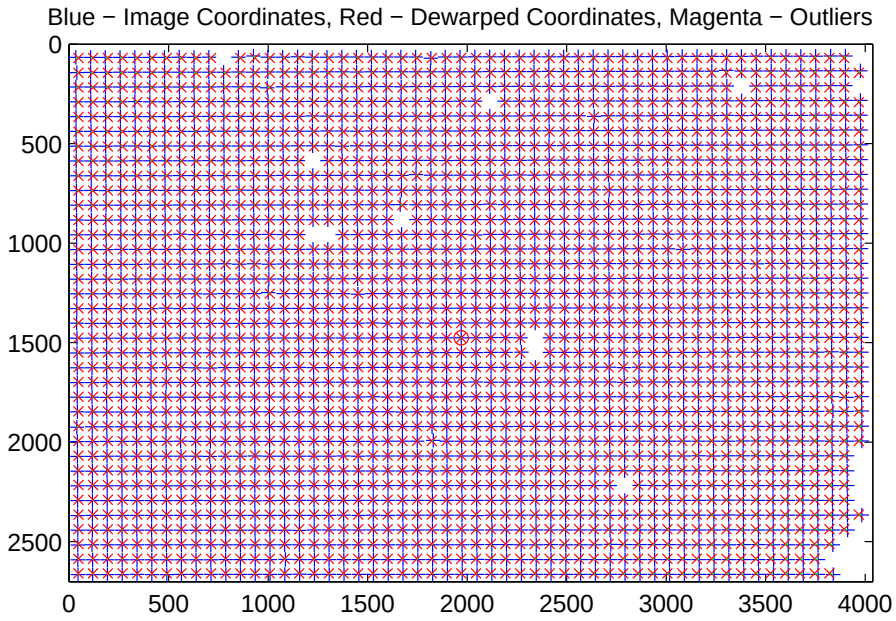


Figure A.28: Final state of a fully fixed and re-matched calibration plane.

rigins.m,' 'showplanes.m,' and 'showvolume.m.' These three scripts will extract that peak-to-grid saved data from the .xls file and plot the origin of each plane in pixel coordinates, all peaks of each color channel of each plane in separate figures, and a complete 3-D plot of all peaks in all planes, respectively.

A.3.1 Matching Mapping Points and Building RBF

Once all planes have been matched, saved, and verified, the "Build Calibration Data" subpanel, Figure A.30, must be populated. First, input the name of each sheet of the .xls file corresponding to a calibration plane under the column labeled "Plane List."

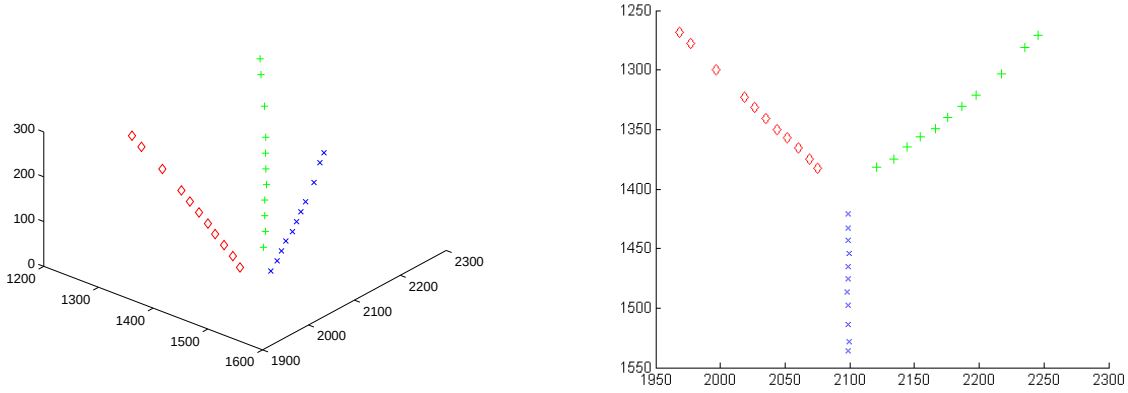


Figure A.29: A sample 3-Dimensional plot of each origin of each color channel and plane in xyZ coordinates. This is also a mapping of the system's epipolar lines.

Then, input the Z-location (in μm) of the plane under the column labeled "plane Z." Next, in the field labeled "Spacing" input the physical value for dot spacing (in mm). Next, input the system magnification in the field labeled "M_FocusLens." If there are any rotation values associated with the system input them in the "theta" and "phi" fields. Then, input the pixel dimensions in the field labeled "pixel_size" (expected value is in μm). Finally, choose a center plane for the calibration and input the sheet name corresponding to that plane in the field labeled "Plane_ImageCenter."

Once all the fields have been populated, click on "Match Mapping Points." Select the .xls file containing the peak-to-grid data. This will match the mapping points across all sheets in the .xls file in preparation for the building of the RBF.

Once the mapping points have been matched, click the "Build Calibration Data" button. This will launch a prompt asking to select a "3D Mapping Method." Click the "RBF" button to select the Radial Basis Function option. This will launch two more prompts, Figure A.31, for parameter inputs, one after the other. All default options, shown in Figure A.31, are adequate for the RBF build process, except for

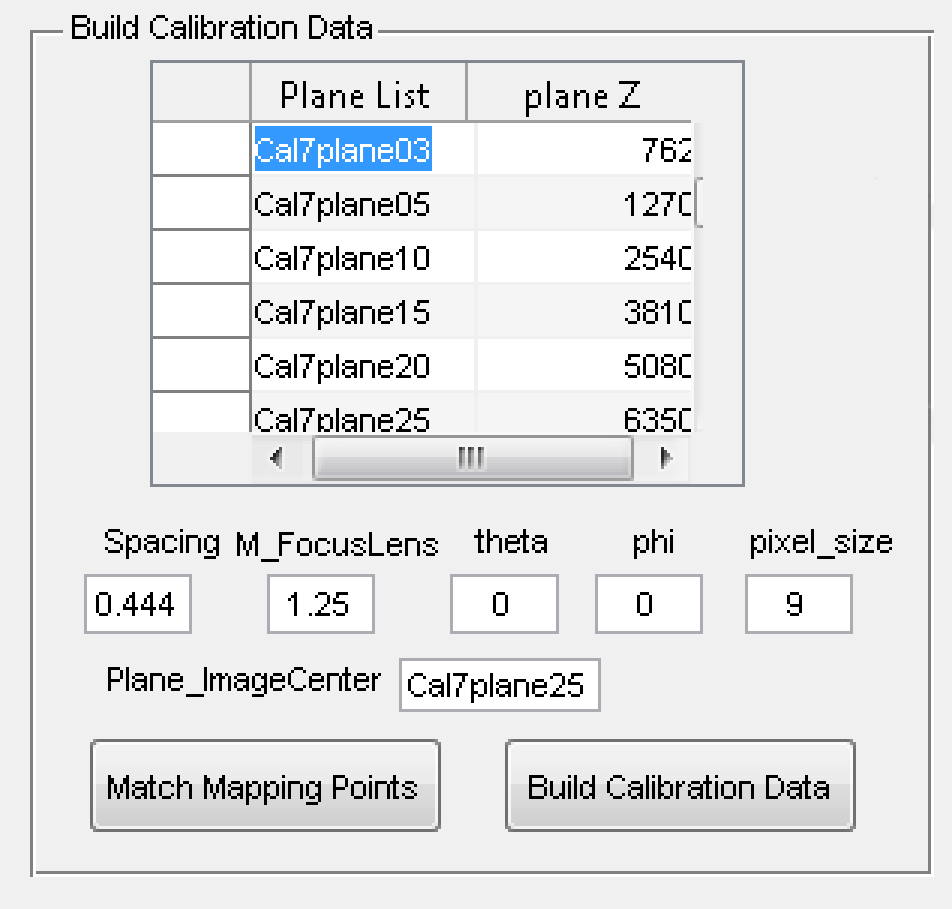


Figure A.30: Calibration Planes and Execution Subpanel for the Calibration Panel with physical dot spacing, system magnification, rotations, pixel size, and center plane

the "Window Size" field. Update this field to reflect that maximum dimension of the input images. For example, images with the dimension $4008 * 2672$ would require a window size of 4008. Select "OK" for both prompts to build the RBF. The output files, "WRBFMapping_xy2XYZ_p0.mat" and "WRBFMapping_Img2Img_p0.mat" are the final results of the calibration process. These contain the RBF interpolation and are the only two files that will be used in the Data Reduction process in the following sections of this software manual.

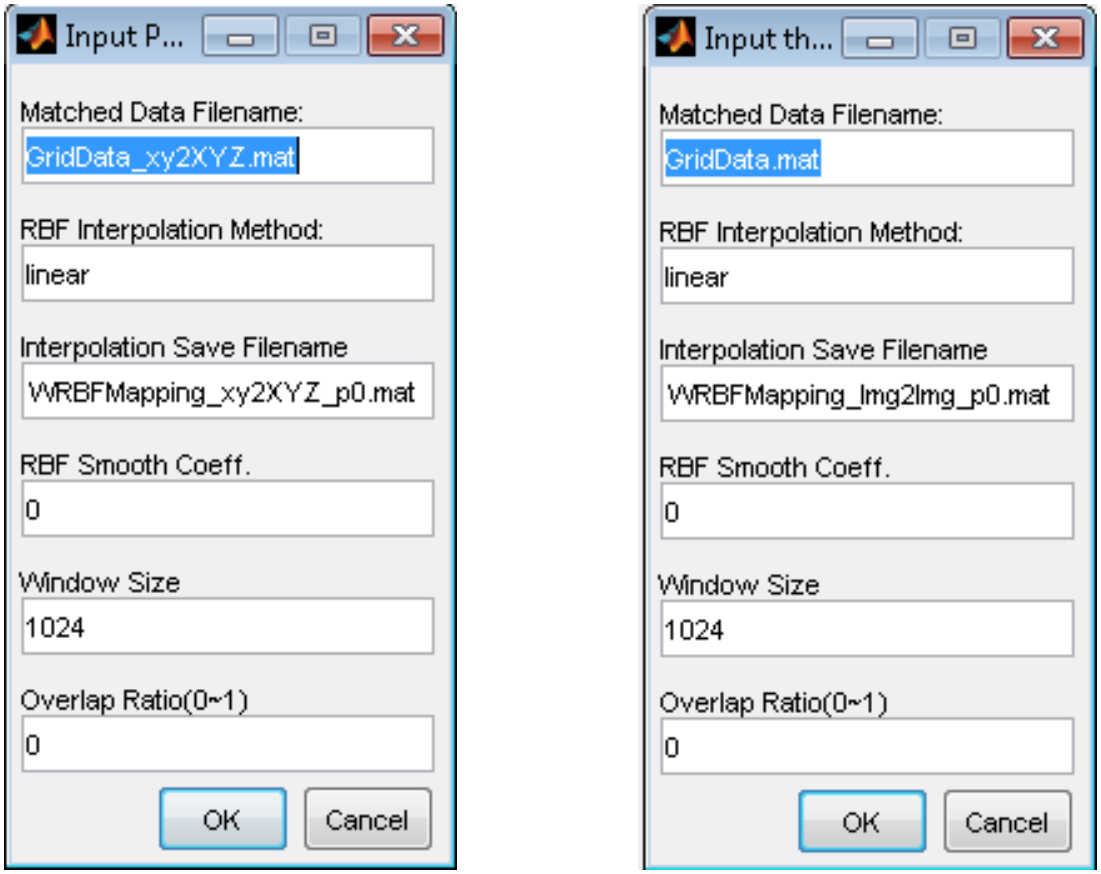


Figure A.31: RBF build option input prompts.

A.4 Data Reduction

What's Required

For the data reduction process, a valid calibration developed using the process described above is required. A valid calibration consists of the two mapping files that are the products of the RBF Build step. In addition to these two files, the GUIs ColorSepPanelV3, PeakFindPanel, and PALocatePanelV1 are required for PTV data reduction.

In addition to the software requirements listed above, PTV images with an ade-

quate single-to-noise ratio, SNR, and even illumination are required. Please note that the settings used in the calibration process are not necessarily valid for use in the data reduction process. They will need to be re-defined and fine-tuned in order to be appropriate for the nature of the data being processed.

Background Subtraction

The background subtraction process must be accomplished on 3DPTV data images prior to further processing. Unlike the optional image averaging step in the calibration process in which the goal was to eliminate moving objects from a static image, the goal of the background subtraction step in the data reduction process is to remove static objects and background noise that is present in all of the data taken in a time series.

The first half of the Background Subtraction process is identical to the Averaging step described in Section A.3. Begin by launching the 'ColorSepPanelV3.m' file in MATLAB® as mentioned above. In the General Subpanel, Figure A.2, insure under "File Type," that the "Raw" option is selected. This option will insure that the file explorer GUI will not sift out the file extensions of your raw data images. Now, select the "Load File" button and ctrl+click each file or shift+click to select all images taken for the entire experiment. Once all images corresponding to the current realization of the experiment have been selected, press "Open." A "Load File" dialogue box should appear confirming the number of files that have been selected. Insure this number matches the number of images taken for the current experiment then acknowledge the dialogue box by clicking "Okay." Now, the selected images have been stored in the GUI's memory and will be used in the current processing step.

Now, within the Background Averaging Subpanel, Figure A.3, click the button

labeled, “Show Result,” if you would like to see the resulting image in the large display area to the right of the GUI. When activated, the button will change to a blue color. Otherwise, leave the button unselected and go directly to the Background Finding step by clicking the button labeled, “Average/Find Background.” Upon clicking this button, a “Save” dialogue window will appear. Save the resulting averaged image with a title corresponding to the current experiment.

When the averaged background image has been successfully calculated and saved, a notification will appear in the MATLAB® Command Window. Next, click on the “Remove BackGround” button. This button is used to remove the resulting averaged image from other images in data reduction. Clicking on this button will launch a File Explorer GUI prompting the user to select a background file to remove from all selected images. The images selected earlier in this step will still be stored in the GUI’s memory. Select the image saved earlier in the step, titled “XXXX_bkgd.mat,” where ‘XXXX’ was the desired name, and click “Open.” Wait for the notification of completion to appear in the MATLAB® Command Window. Verify that the background region has been successfully removed from all images by viewing the background image and resulting images containing the ‘_rev.mat’ extensions.

The settings in this subpanel, including “Offset,” “Low Input,” “High Input,” and “Gamma” may be left at default values of 0, 0, 255, and 1, respectively. If, upon further examination, the resulting images need contrast adjustment, these fields may be further-tuned for optimal contrast.

Pre-Processing

The Pre-Processing/Denoise step is identical to the one described in the Calibration Process Section of this software manual. Use the Pre-Processing/Denoise Subpanel,

Figure A.4, of the ColorSepPanelV3 GUI Panel, Figure A.1. For detailed instructions, refer to Section A.3.

Optional: Color Separation

The Color Separation step is identical to the one described in the Calibration Process Section of this software manual. The Color Separation Procedure is, again, optional and is only necessary under certain experimental conditions. To accomplish color separation, use the Color Separation Subpanel, Figure A.6, of the ColorSepPanelV3 GUI Panel, Figure A.1. For a detailed explanation of when and how to use the color separation technique, refer to Section A.3.

Post-Processing

The Post-Processing step is identical to the one described in the Calibration Process Section of this software manual. To accomplish Post-Processing, use the Post-Processing Subpanel, Figure A.11, of the ColorSepPanelV3 GUI Panel, Figure A.1. For a detailed explanation of how to post-process PTV images, refer to Section A.3.

Peak Finding

Peak finding is accomplished in the same fashion as it was during the calibration procedure in Section A.3. Use the PeakFindPanel, Figure A.12. The settings file defined for the calibration process, however, will not be valid for the data reduction process. New settings must be defined based on the nature of the images according to Section A.3.

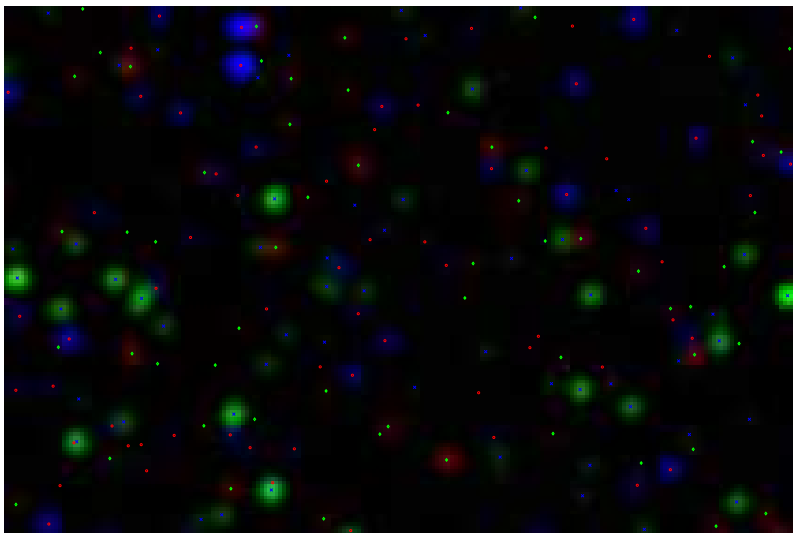


Figure A.32: Results of the Peak Finding by CCM Algorithm overlaid on a small portion of the original image using 'plotPADATAv1.m.' For visibility even in cases of saturation, red peak locations are plotted with green markers, green peaks with blue markers, and blue peaks with red markers. Note: Some artifacting is seen in the above particle image that is not present its original MATLAB® form.

The results may be plotted in the same fashion as described in Section A.3 using 'plotPADATAv1.m.' Figure A.32 is an example of a particle image with subpixel coordinate peak locations overlaid.

Triplet Matching

The Triplet Matching process maps peak locations from all three image channels from the image coordinate system to the physical coordinate system. It accomplishes this

using the RBF interpolation results developed in the calibration process. It is recommended that the user first fine-tunes the Triplet Matching Settings using a relatively lighter particle density, ∼1,000 particles, as the computational cost increase incurred by an increase in particles is non-linear. To begin the process, launch the GUI Panel 'PALocatePanelV1,' Figure A.33. First, the settings for this step are described in Section A.4. Then, instructions on running the process are in Section A.4

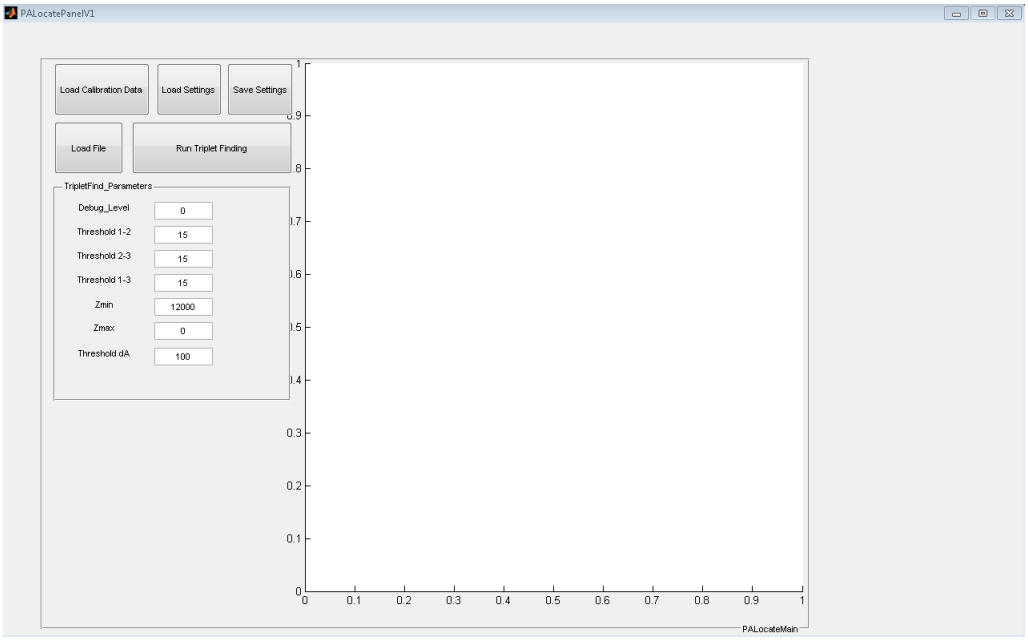


Figure A.33: Full GUI for 'PALocatePanel,' used to map particle image coordinates to physical coordinates via the RBF interpolation built in the calibration process.

Triplet Matching Settings

Each paragraph within this section corresponds to and explains a field within the Triplet Matching Settings Subpanel, Figure A.34.

Note: The parameter values featured in Figures A.33 and A.34 are not default values. These values correspond to the values used throughout most of the Backward-Facing Step Flow study in the Dona Thesis.

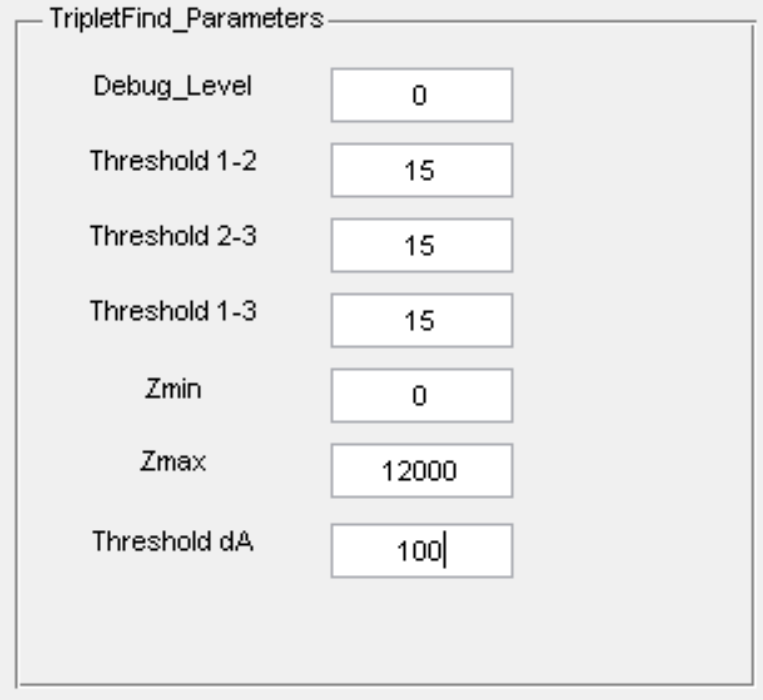


Figure A.34: Settings Subpanel for the Physical Location mapping (Triplet Matching) GUI Panel. Note: These are not the default settings. These are the settings used throughout the Backward-Facing Step Turbulence Study from the Dona thesis.

Debug_Level The debug level varies the verbosity of the output from the Triplet Matching step. Number of plots displayed increases with debug level. A debug level of 2 shows the algorithm cycling through each peak of the peak finding results from

the previous step. It then draws epipolar lines and searches along the length of the line, showing potential triplet matches. It is recommended that a relatively sparse particle image is used with a debug level of 2. It is important to remember while debugging that the first peaks within a significant margin of the image may not be successfully matched, as one or more components of the triplets may be out of frame.

Threshold A-B Threshold 1-2, Threshold 2-3, and Threshold 1-3 all refer to a similar setting. The numbers, A-B, of the setting refer to the epipolar line drawn from peak A to peak B of the triplet being matched. Each threshold number corresponds to the pixel distance from the line within which a particle will be considered as a potential match for a corner of the triplet. For example, setting the Threshold 1-2 value to 15 means the Triplet Matching algorithm considers any potential channel 2 peak within 15 pixels of the epipolar line drawn from peak 1. A visual example of this thresholding is displayed in Figure A.35 where the peaks in the 2-channel of the image are the green x's, the cyan line is the epipolar line from channel 1-2, and the candidates within Threshold 1-2 pixels of the epipolar line are highlighted as red circles. The magnitude of this threshold setting depends on the individual application and should be fine-tuned according to results. In addition to the potentially decreased accuracy caused by selecting too large of a threshold value, computation time significantly increases as the algorithm must consider more peaks as potential triplet matches. Typically, all three of these linear thresholds may be set to the same value.

Zmin The lower limit of the Z-range through which the Epipolar Line Search method will search. As triangle side length decreases as the particle approaches the

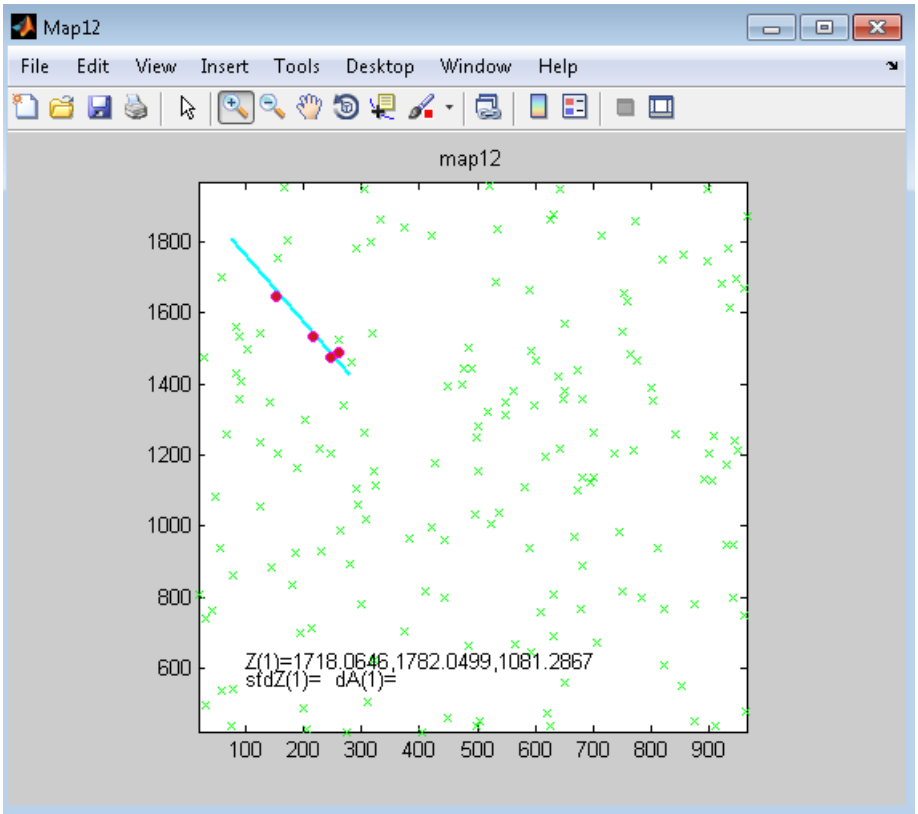


Figure A.35: A visual representation of the Threshold 1-2 process. This figure was developed as an output of debug level 2.

focal plane, the algorithm uses the Zmin value to determine the minimum distance along the epipolar line away from the origin peak at which the algorithm may begin to consider a peak a candidate for the triplet. This value should roughly correspond to the lowest Z-value observed in the experiment (ideally corresponds to the lowest Z-plane included in the calibration).

Zmax The upper limit of the Z-range through which the Epipolar Line Search method will search. As triangle side length increases as the particle departs from the

focal plane, the algorithm uses the Zmax value to determine the maximum distance along the epipolar line away from the origin peak at which the algorithm may still consider a peak a candidate for the triplet. This value should roughly correspond to the highest Z-value observed in the experiment (ideally corresponds to the highest Z-plane included in the calibration).

Threshold dA Essentially, the RBF Interpolated mapping files used in Triplet Matching contain large databases of triangle geometry that a given particle would make as it translates through the Z-range of the volume at any given X-Y location. The dA value quantifies the difference between the pixel area contained within the candidate triplet and the pixel area contained within the RBF-reference triplet. The magnitude of this threshold setting depends on the individual application and should be fine-tuned according to results. In addition to the potentially decreased accuracy caused by selecting too large of a threshold value, computation time also increases as the algorithm must consider more potential triplet matches.

Running the Triplet Matching Algorithm

Once the settings have been defined, running the Triplet Matching Algorithm is simple. First, click on the "Load Calibration Data" button within the Execution Subpanel, Figure A.36. Doing so will launch a File Explorer GUI that displays only files with the extension '_Img2Img.mat,' the first result from the RBF Build step in the calibration process. Proceed to the directory in which the calibration was accomplished, select the appropriate file (only one should appear with this extension), and click the "Open" button. The Image to Image mapping will be stored in the GUI

memory for use during Triplet Matching. Immediately following this step, a new File Explorer GUI will automatically appear. This time, it will only display files with the extension '_xy2XYZ.mat,' the second result from the RBF Build step in the calibration process. Proceed to the directory in which the calibration was accomplished, again, select the appropriate file (should be the only file with this extension), and click the "Open" button. The xy (image) to XYZ (physical) Mapping will be stored in the GUI memory for use during Triplet Matching. No confirmation will appear in the MATLAB® Command Window. The process will simply not execute, however, if this has not been done successfully.

Figure A.36: Execution Subpanel for the Physical Location mapping (Triplet Matching) GUI Panel

While testing or fine-tuning Triplet Matching settings, it is recommended that the settings combination used in each trial is saved for future reference. Saving the settings contained in the Triplet Matching Settings fields is simple. Click on the button labeled "Save Settings" and choose a unique file name for the settings file. The file will be saved 'XXXX_PAsettings.mat,' where "XXXX" is the user-specified name. This settings file is a '.mat' file containing a structured array easily opened and manipulated without the GUI, via the MATLAB® Command Window.

To load a previously-specified and saved settings file, simply click on the button labeled "Load Settings." The File Explorer GUI will automatically display only settings files with the extension '_PAsettings.mat.' Loading a settings file will result in a change of the user-defined fields in the Triplet Matching Settings Subpanel, Figure A.34, to reflect those contained in the file. Verify that the settings are correct.

Next, select the "Load File" button and proceed to the directory where the 3DPTV image peak locations are stored. This will launch a File Explorer GUI that will only display files with the extension '_PADATA.mat.' Select all desired '_PADATA' files and press the "Open" button. Insure the number in the confirmation window matches the number of peak location files selected and acknowledge by pressing "OK." Now that the calibration data, settings data, and peak location data have been loaded into the GUI, the Triplet Finding my be executed. Do so by simply clicking on the "Run Triplet Finding" button.

The results of the Triplet Finding algorithm will be saved in a file with the extension '_PAFound.mat' in place of the '_PADATA.mat' extension. These files contain a structure including the physical coordinates of each matched particle as well as the dA, Z standard deviation, and reference location within the '_PADATA.mat' file from which each peak was taken ('TL,' 'TR,' and 'B' fields). These results my be plotted and viewed in 3-D by extracting the "PAfoundi.XYZg" field from each entry of the structured array.

A.5 Batch Data Processing

For a single script to run the entire software package from background subtraction to triplet matching, use the code "DataReducev2.m." The m file is located in the folder "Data Shell" along with the other relevant codes called by the driver. The

user may manually define any of the image processing options, but the only settings that are absolutely necessary are the peak finding and triplet finding settings files ending in "pksettings.mat" and "PAsettings.mat," respectively. These files must be fully-defined and located in the same folder as the data reduction script. Finally, the user must simply define which frames should be reduced by entering the full number (ex: '010';'011') in the frames cell string. Then, simply identify which processes the script should run by entering a 1 or 0 as the value for each step. The background subtraction step must only be accomplished once for each data set. After this occurs, the script will pull the already background-subtracted frames when the background subtraction variable is set to 0.

A.6 Particle Tracking

The particle tracking process uses two scripts. The first code, "PTV3D_beta4_rootv2.m," calls the second function, "PTV_VisionAssociation_V3d.m," and tracks each frame pair. It is important to insure that each particle location file contained within the folder is located next to its corresponding location file so that tracking occurs between correct data sets. It is important to check that this is occurring correctly or tracking will not be real. The user must define settings in the "PTV3D_beta4_rootv2.m" file including camera system settings, VOI dimensions, maximum possible displacements, and voxel side lengths. The outlier settings are only used if outlier detection during the PTV process is enabled. Otherwise, the outlier detection process is simply accomplished on the completed vector field file. The location of the sigma PTV-guiding definition is at the bottom of the "PTV_VisionAssociation_V3d.m" file. The results for each field in a structure will be in the variable stack after the operation is complete. It will simply be the variable "ans." The user must rename and save the variable in a form that the outlier detection software will read and process.

A.7 Outlier Detection

The median outlier filtering function is coded in the script "MScript_CombinePTVField3sv1.m." This searches for a specific file with a specific variable name that is the output of the particle tracking step. A typical value for Outlier_tol is approximately 0.1. A typically value for Outlier_thr is approximately 3-5, but should be experimented to achieve the most accurate results with no outliers remaining. This function combines all fields contained in the previous answer structure to a single field that has been outlier-filtered. It then saves the resulting field automatically. It also plots the filtered and original vector fields in a color-coded manner to see the outliers that were removed. Accomplish this step in an iterative fashion to achieve the best results.

A.8 SVD Interpolation

Use the "SVDinterptest.m" file to assess the completeness of the data set for SVD interpolation purposes. each ellipsoidal neighborhood must have at least 10 unstructured vectors to be valid. If the data set is dense enough, then use "VectorstoGridv3v1.m" to automatically process the filtered vector fields to a structured vector field with the user-defined units and grid spacing. This routine calls the actual interpolation technique, "SVDinterp3.m." The SVD interpolation can also output the velocity gradient and second-order moments, but must be altered to do so. This process can be computationally intensive.

A.9 Turbulence Analysis

Some model application has been accomplished and can be found in the file "TurbulenceAnalysistestv3.m." This file reads in the structured array of vector fields that the SVD Interpolation built and applies filtering operations and models to the data.

Appendix B

ADDITIONAL PLOTS

B.1 Correlation Coefficients

B.1.1 $\Delta = 10\delta$

Table B.1: Correlation coefficients of modeled deviatoric stress tensor elements. Columns represent $\rho(\tau_{12}^{\Delta}, \tau_{12}^{\Delta,\mathrm{mod}})$ for the relevant model and stress tensor component.

Model Type	τ_{12}^{Δ}	τ_{13}^{Δ}	τ_{23}^{Δ}
Smagorinsky Model	-0.0085	-0.0239	-0.1250
Similarity Model,$\gamma = 1$	0.5682	0.4957	0.4851
Mixed Model	0.5617	0.4774	0.4599
Dynamic Model	-0.0948	-0.0081	-0.0996
CSM	-0.0372	-0.0158	-0.0336

Table B.2: Correlation coefficients of modeled SGS Force, $\nabla \cdot \tau^{\Delta}$

Model Type	$\rho(\nabla \cdot \tau^{\Delta}, \nabla \cdot \tau^{\Delta,\mathrm{mod}})$
Smagorinsky Model	0.0523
Similarity Model,$\gamma = 1$	0.1033
Mixed Model,$\gamma = 1$	0.1062
Dynamic Model,$\alpha = 1$	0.0198
CSM	0.0239

Table B.3: Correlation coefficients of modeled SGS Dissipation, $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$

Model Type	$\rho(-\tau_{ij}^{\Delta}\widetilde{S}_{ij}, -\tau_{ij}^{\Delta,\mathrm{mod}}\widetilde{S}_{ij})$
Smagorinsky Model	0.0614
Similarity Model,$\gamma = 1$	0.8174
Mixed Model,$\gamma = 1$	0.4460
Dynamic Model,$\alpha = 1$	-0.0144
CSM	0.1618

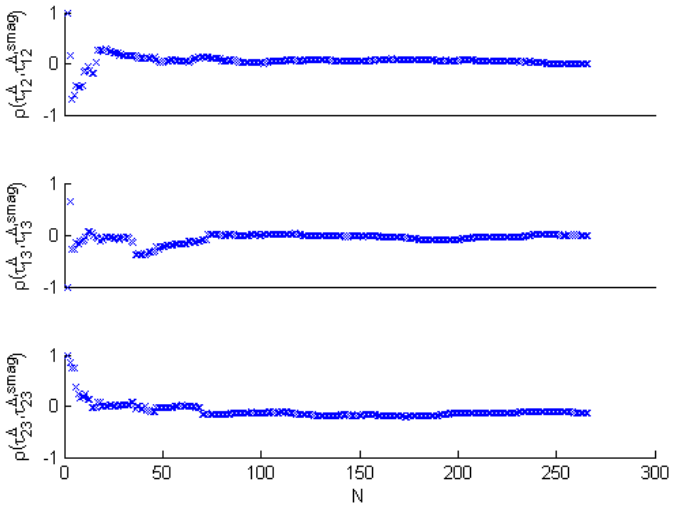


Figure B.1: Smagorinsky Stress Tensor Element Correlation Coefficients

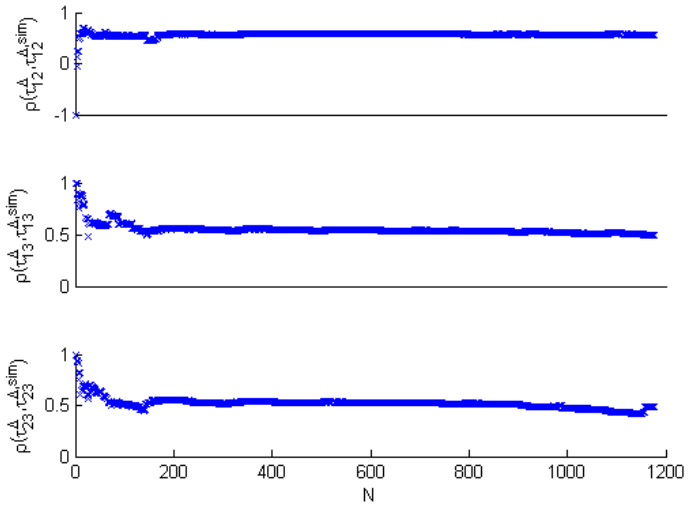


Figure B.2: Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$

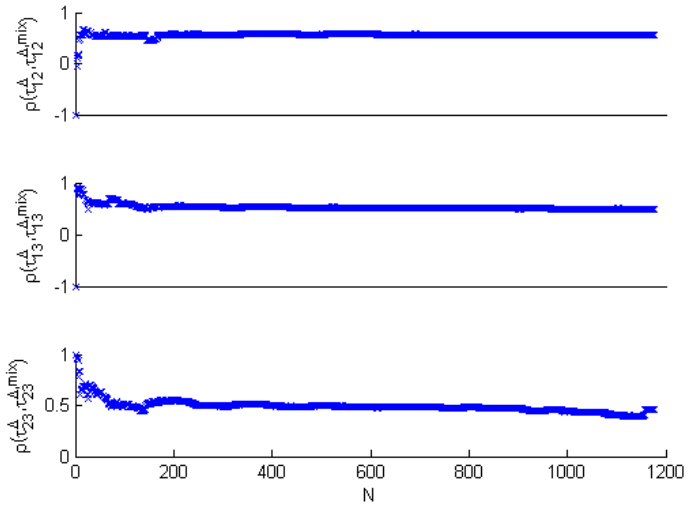


Figure B.3: Mixed Stress Tensor Element Correlation Coefficients

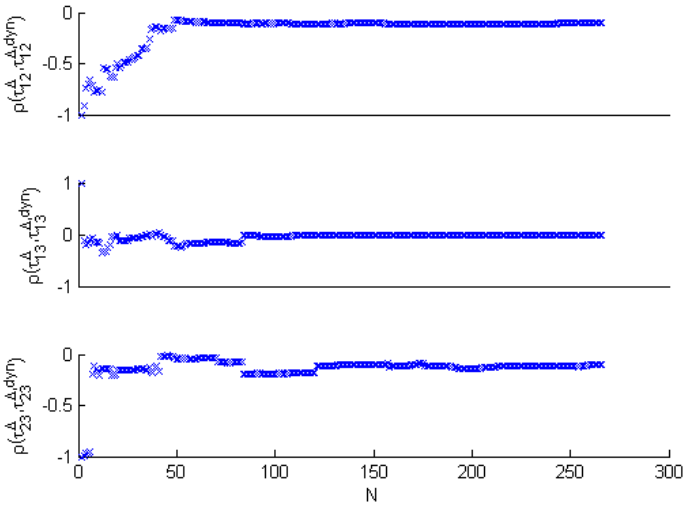


Figure B.4: Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$

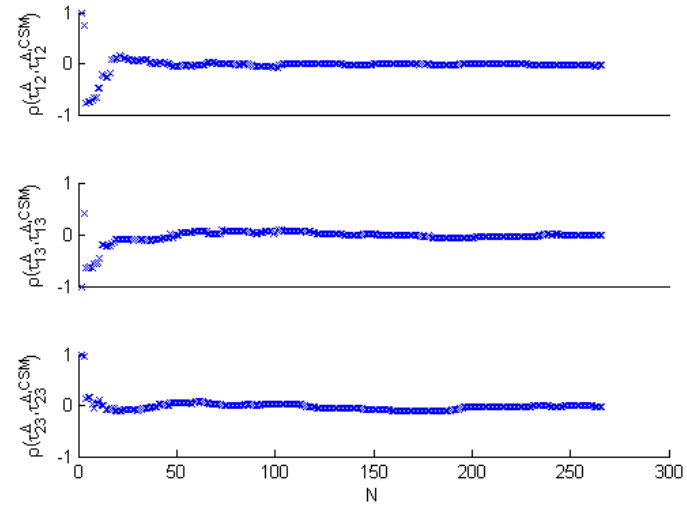


Figure B.5: CSM Stress Tensor Element Correlation Coefficients

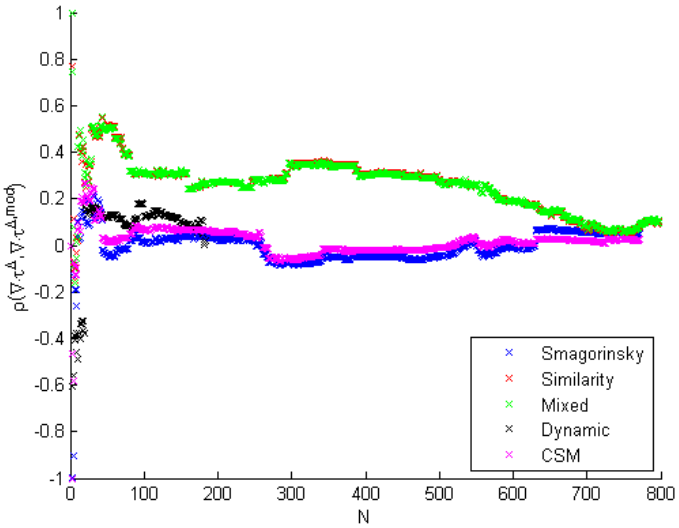


Figure B.6: SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$

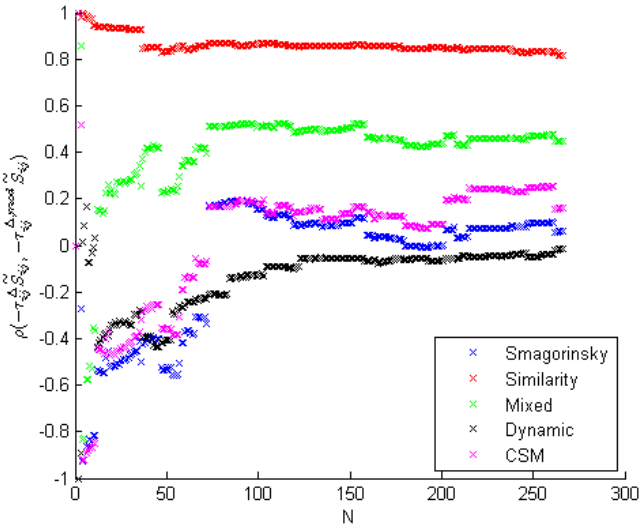


Figure B.7: SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$

B.1.2 $\Delta = 5\delta$

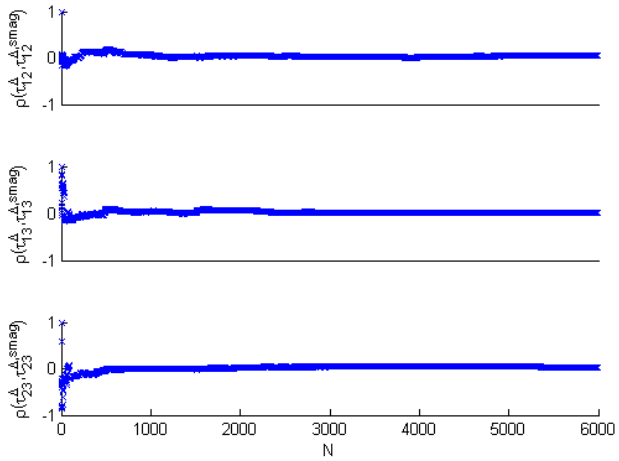


Figure B.8: Smagorinsky Stress Tensor Element Correlation Coefficients

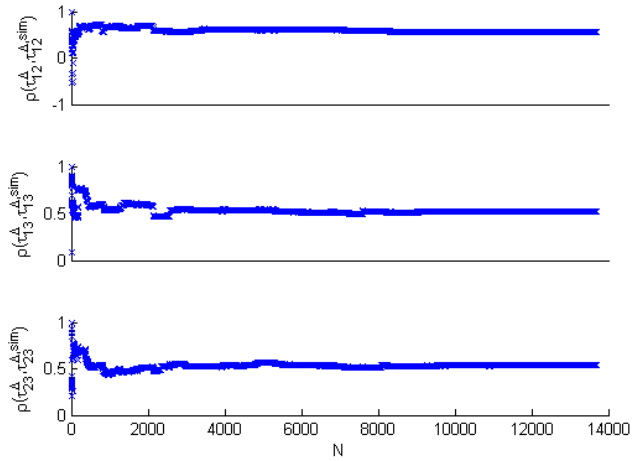


Figure B.9: Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$

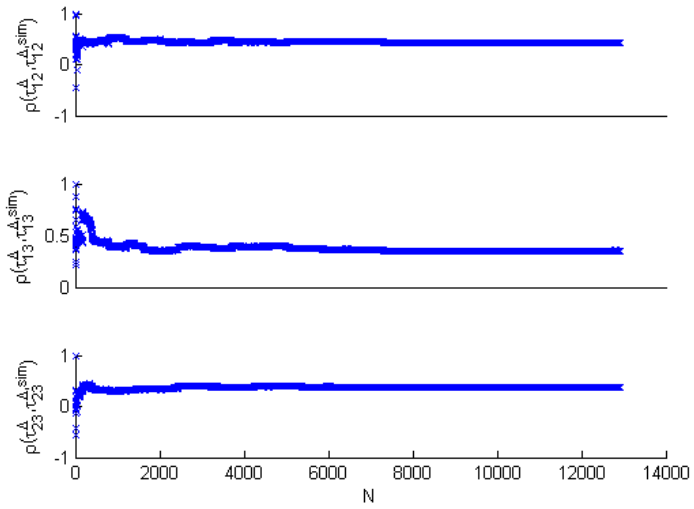


Figure B.10: Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 2$

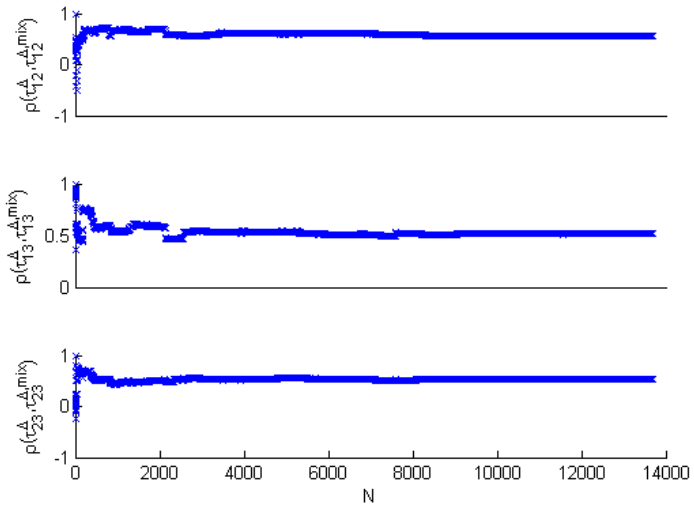


Figure B.11: Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 1$

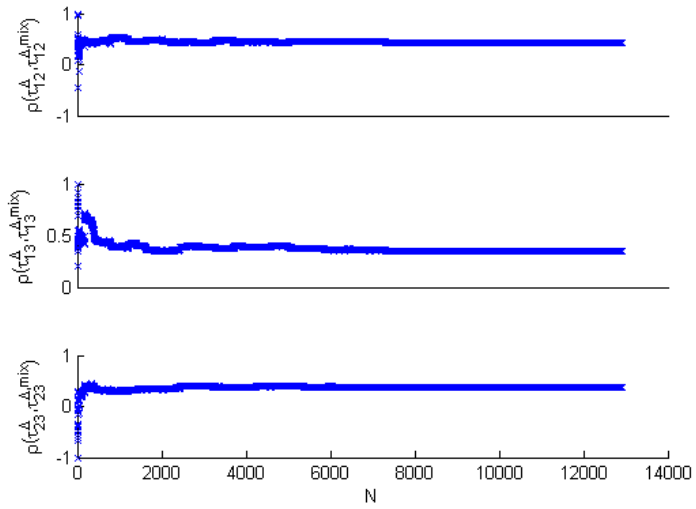


Figure B.12: Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 2$

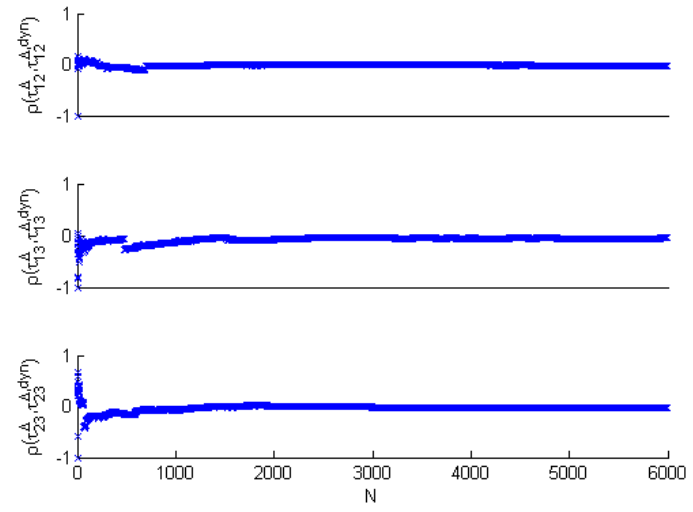


Figure B.13: Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$

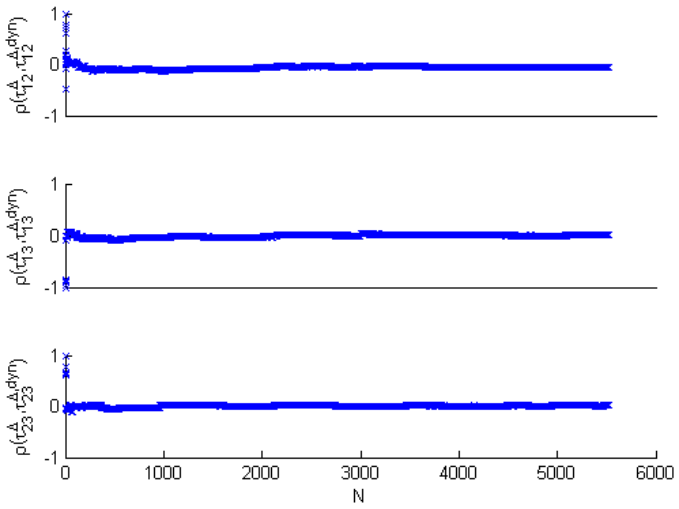


Figure B.14: Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 2$

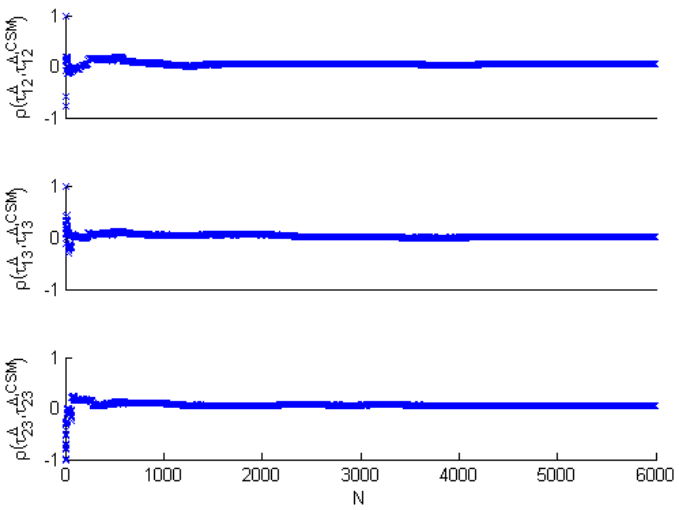


Figure B.15: CSM Stress Tensor Element Correlation Coefficients

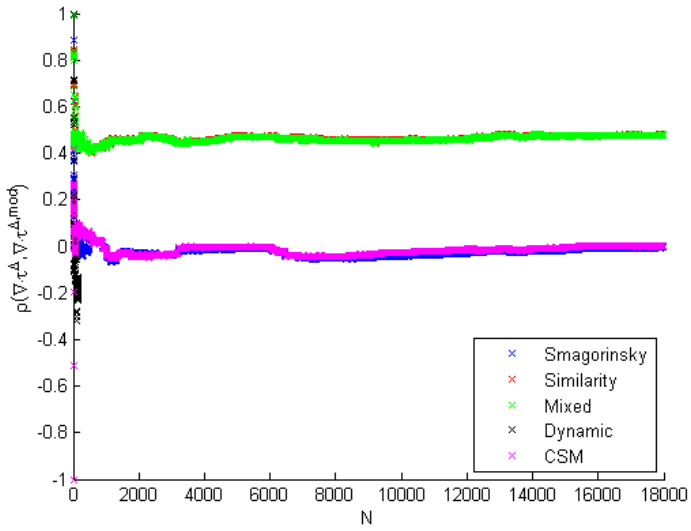


Figure B.16: SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$

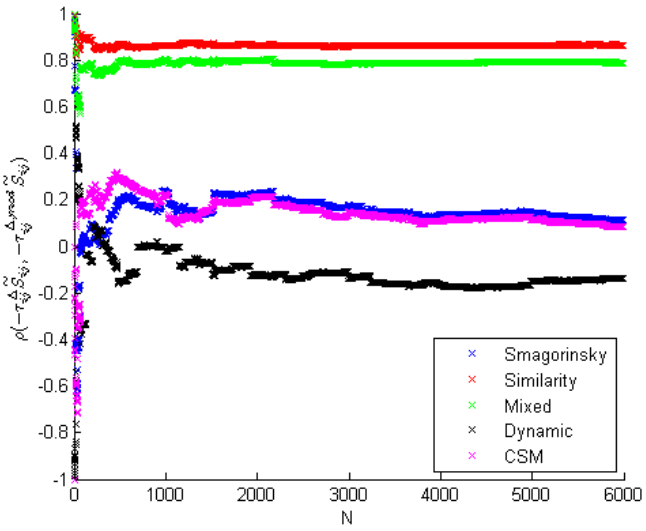


Figure B.17: SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$

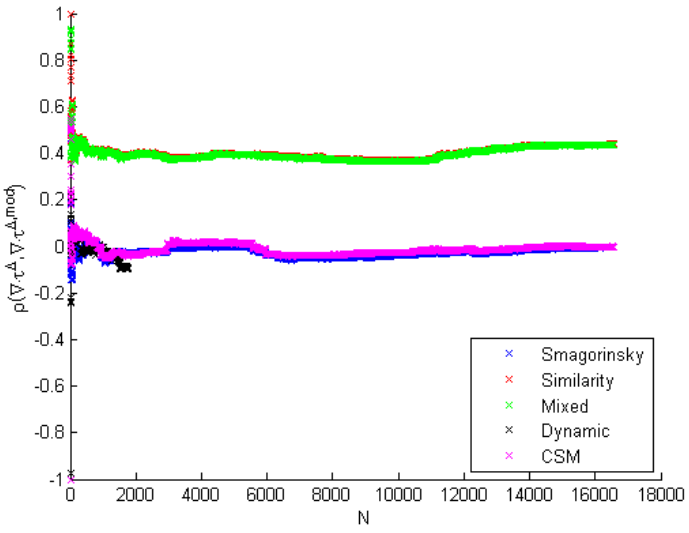


Figure B.18: SGS Force Correlation for all models,$\gamma = 2$,$\alpha = 2$

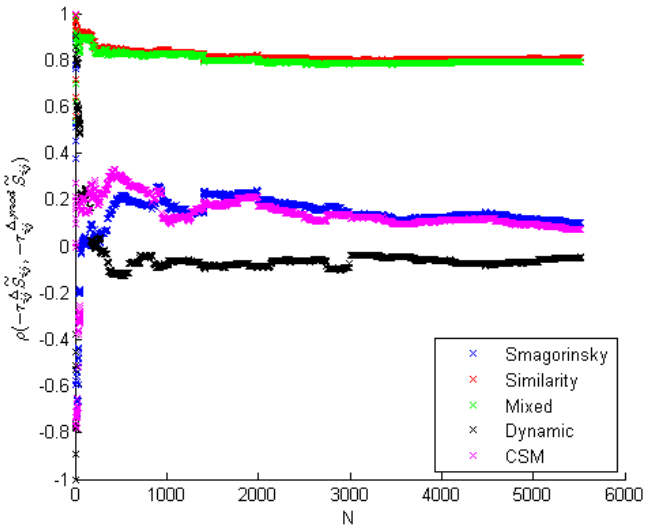


Figure B.19: SGS Dissipation Correlation for all models,$\gamma = 2$,$\alpha = 2$

B.1.3 $\Delta = 2\delta$

Table B.4: Correlation coefficients of modeled deviatoric stress tensor elements. Columns represent $\rho(\tau_{12}^{\Delta}, \tau_{12}^{\Delta,\mathrm{mod}})$ for the relevant model and stress tensor component.

Model Type	τ_{12}^{Δ}	τ_{13}^{Δ}	τ_{23}^{Δ}
Smagorinsky Model	0.0034	0.-0.0091	-0.0074
Similarity Model,$\gamma = 1$	0.3861	0.3733	0.3736
Similarity Model,$\gamma = 2$	0.4310	0.3552	0.3768
Mixed Model,$\gamma = 1$	0.5554	0.5170	0.5377
Mixed Model,$\gamma = 2$	0.4172	0.3865	0.3901
Dynamic Model,$\alpha = 1$	-0.0060	0.0163	0.0163
Dynamic Model,$\alpha = 2$	-0.0088	0.0056	-0.0013
CSM	0.0215	-0.0111	-0.0023

Table B.5: Correlation coefficients of modeled SGS Force, $\nabla \cdot \tau^{\Delta}$

Model Type	$\rho(\nabla \cdot \tau^{\Delta}, \nabla \cdot \tau^{\Delta,\mathrm{mod}})$
Smagorinsky Model	0.0063
Similarity Model,$\gamma = 1$	0.3937
Similarity Model,$\gamma = 2$	0.4725
Mixed Model,$\gamma = 1$	0.3907
Mixed Model,$\gamma = 2$	0.4699
Dynamic Model,$\alpha = 1$	-0.0084
Dynamic Model,$\alpha = 2$	-0.0203
CSM	-0.0009

Table B.6: Correlation coefficients of modeled SGS Dissipation, $-\tau_{ij}^{\Delta}\widetilde{S}_{ij}$

Model Type	$\rho(-\tau_{ij}^{\Delta}\widetilde{S}_{ij}, -\tau_{ij}^{\Delta,\mathrm{mod}}\widetilde{S}_{ij})$
Smagorinsky Model	0.0189
Similarity Model,$\gamma = 1$	0.4678
Similarity Model,$\gamma = 2$	0.6207
Mixed Model,$\gamma = 1$	0.4434
Mixed Model,$\gamma = 2$	0.6031
Dynamic Model,$\alpha = 1$	-0.0084
Dynamic Model,$\alpha = 2$	-0.0435
CSM	0.0130

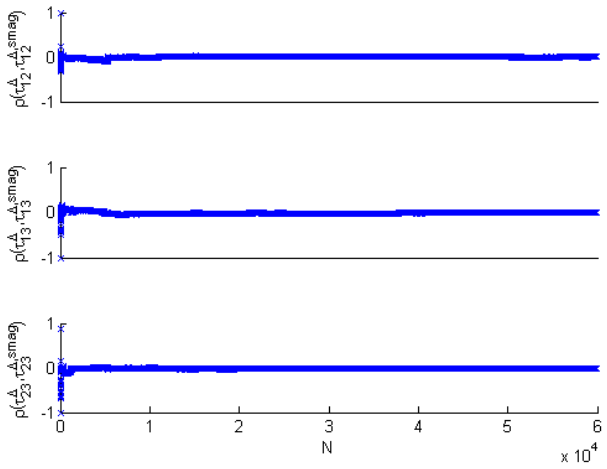


Figure B.20: Smagorinsky Stress Tensor Element Correlation Coefficients

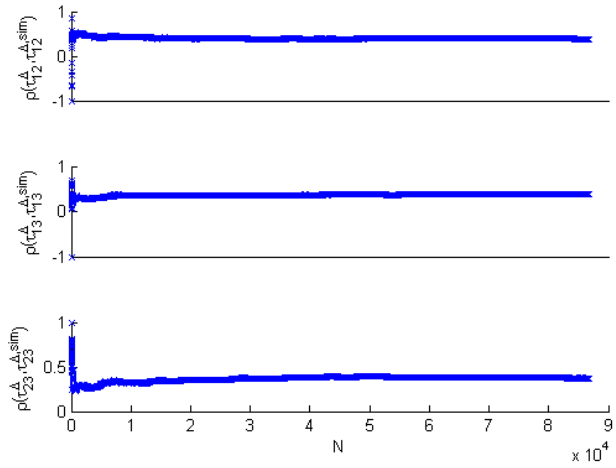


Figure B.21: Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 1$

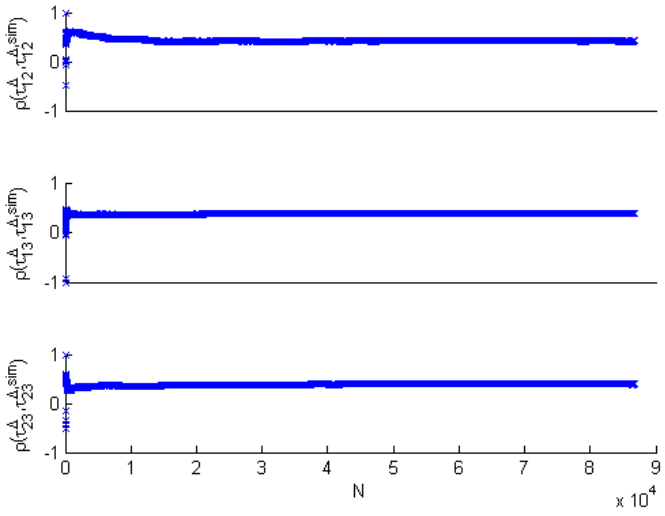


Figure B.22: Similarity Stress Tensor Element Correlation Coefficients,$\gamma = 2$

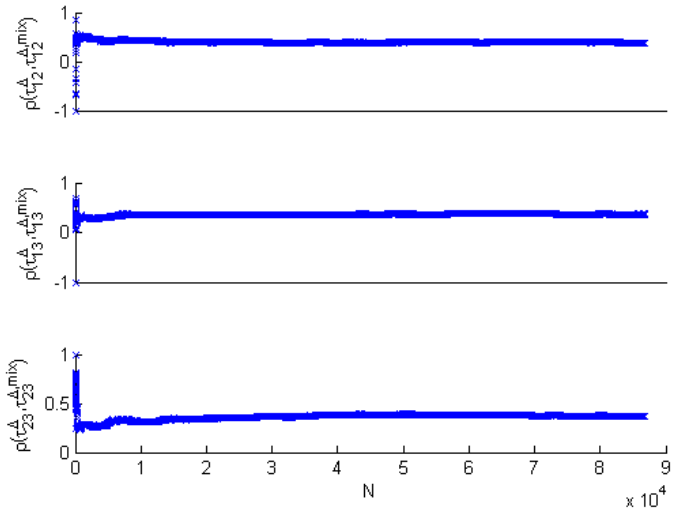


Figure B.23: Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 1$

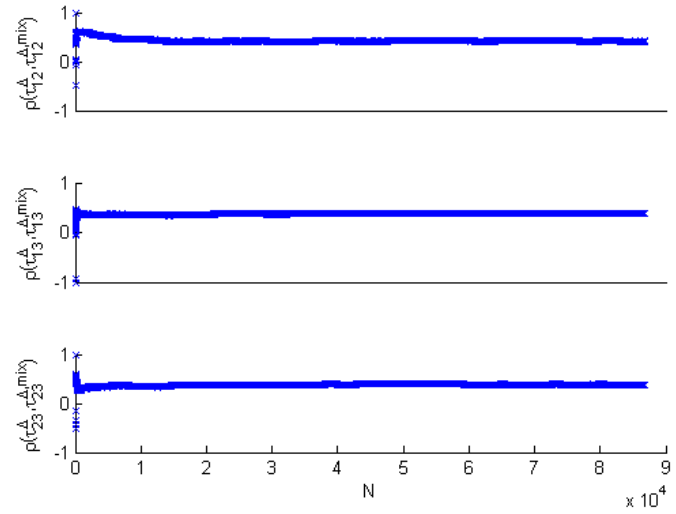


Figure B.24: Mixed Stress Tensor Element Correlation Coefficients,$\gamma = 2$

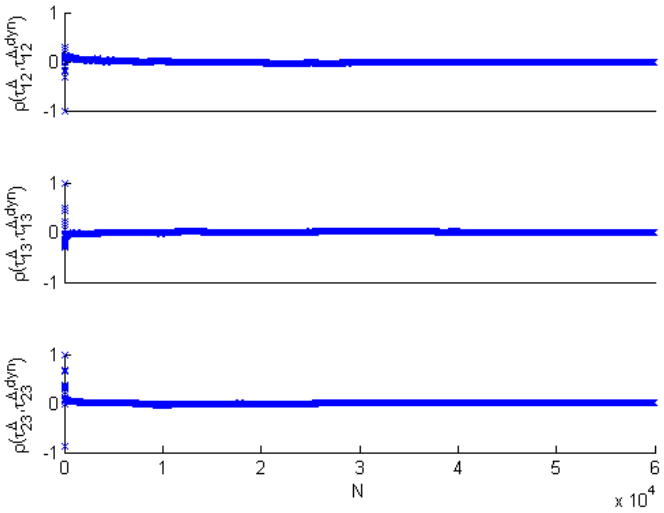


Figure B.25: Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 1$

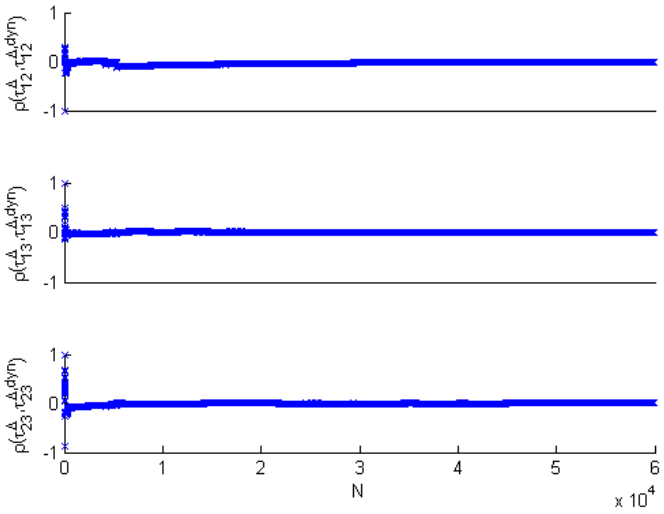


Figure B.26: Dynamic Stress Tensor Element Correlation Coefficients,$\alpha = 2$

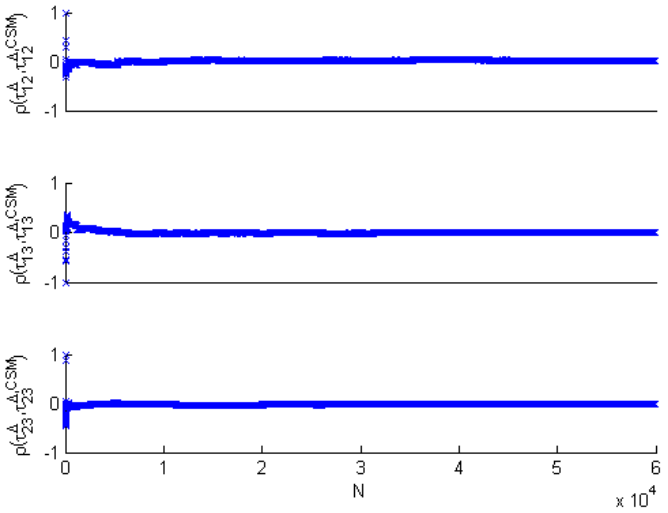


Figure B.27: CSM Stress Tensor Element Correlation Coefficients

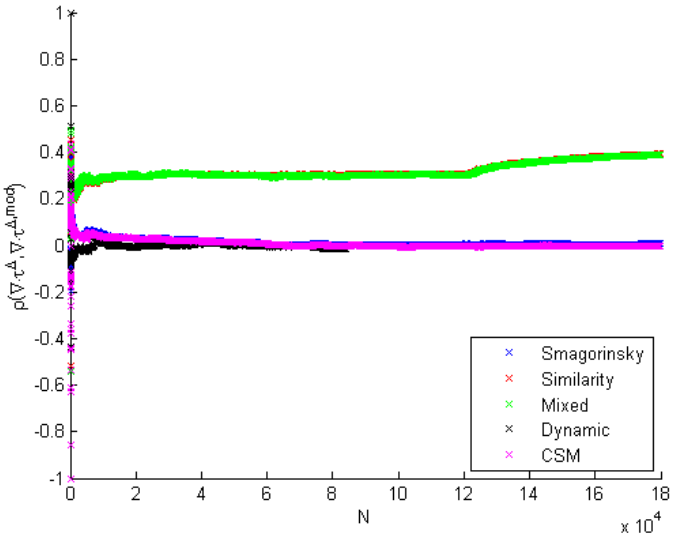


Figure B.28: SGS Force Correlation for all models,$\gamma = 1$,$\alpha = 1$

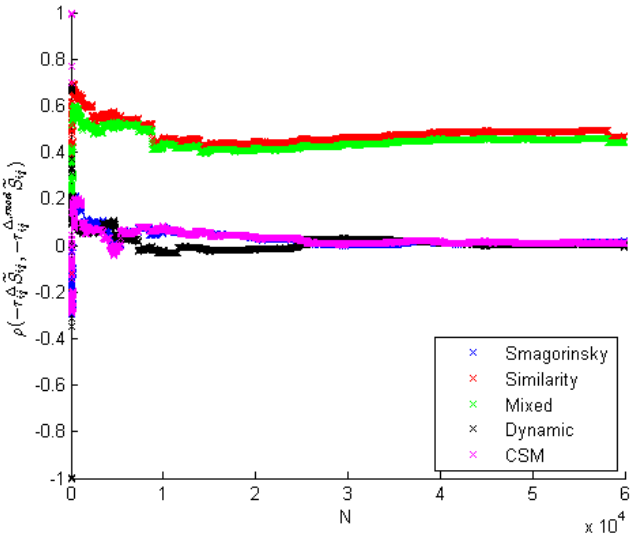


Figure B.29: SGS Dissipation Correlation for all models,$\gamma = 1$,$\alpha = 1$

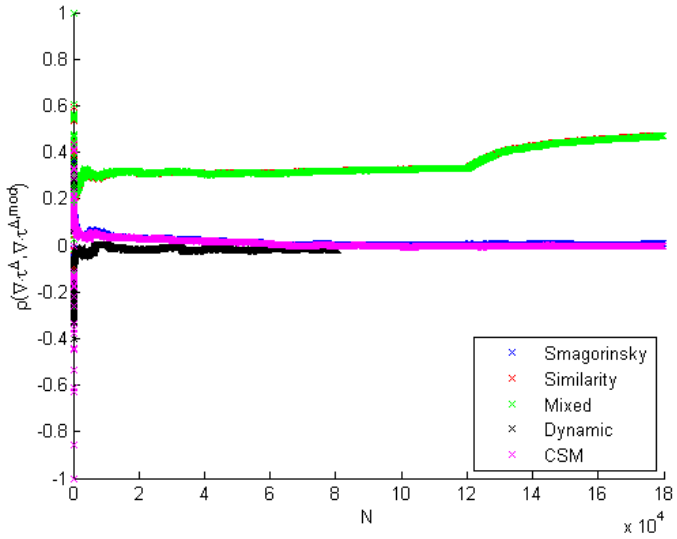


Figure B.30: SGS Force Correlation for all models,$\gamma = 2$,$\alpha = 2$

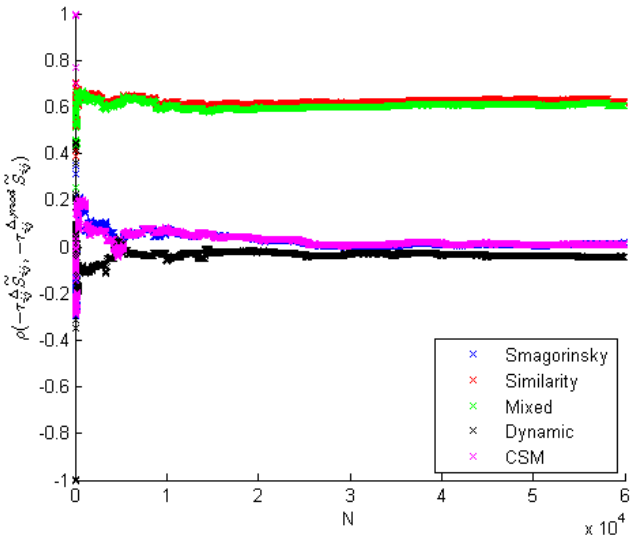


Figure B.31: SGS Dissipation Correlation for all models,$\gamma = 2$,$\alpha = 2$