Particle Streak Anemometry: A New Method for Proximal Flow Sensing from Aircraft

by

T. W. Nichols

B.S., Massachusetts Institute of Technology, 2012

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Aerospace Engineering Sciences

2017

This thesis entitled:
Particle Streak Anemometry: A New Method for Proximal Flow Sensing from Aircraft
written by T. W. Nichols
has been approved for the Department of Aerospace Engineering Sciences

Brian Argrow

Dale Lawrence

Eric Frew

Gijs de Boer

John Farnsworth

Date

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.

Nichols, T. W. (Ph.D.,Aerospace Engineering Sciences)

Particle Streak Anemometry: A New Method for Proximal Flow Sensing from Aircraft

Thesis directed by Prof. Brian Argrow

Accurate sensing of relative air flow direction from fixed-wing small unmanned aircraft (sUAS) is challenging with existing multi-hole pitot-static and vane systems. Sub-degree direction accuracy is generally not available on such systems and disturbances to the local flow field, induced by the airframe, introduce an additional error source.

An optical imaging approach to make a relative air velocity measurement with high-directional accuracy is presented. Optical methods offer the capability to make a proximal measurement in undisturbed air outside of the local flow field without the need to place sensors on vulnerable probes extended ahead of the aircraft. Current imaging flow analysis techniques for laboratory use rely on relatively thin imaged volumes and sophisticated hardware and intensity thresholding in low-background conditions. A new method is derived and assessed using a particle streak imaging technique that can be implemented with low-cost commercial cameras and illumination systems, and can function in imaged volumes of arbitrary depth with complex background signal.

The new technique, referred to as particle streak anemometry (PSA) (to differentiate from particle streak velocimetry which makes a field measurement rather than a single bulk flow measurement) utilizes a modified Canny Edge detection algorithm with a connected component analysis and principle component analysis to detect streak ends in complex imaging conditions. A linear solution for the air velocity direction is then implemented with a random sample consensus (RANSAC) solution approach. A single DOF non-linear, non-convex optimization problem is then solved for the air speed through an iterative approach. The technique was tested through simulation and wind tunnel tests yielding angular accuracies under 0.2 degrees, superior to the performance of existing commercial systems. Air speed

error standard deviations varied from 1.6 to 2.2 m/s depending on the techniques of implementation. While air speed sensing is secondary to accurate flow direction measurement, the air speed results were in line with commercial pitot static systems at low speeds.

Dedication

This thesis is dedicated to my parents

Acknowledgements

I am extremely grateful to David Pope for the work this dissertation grew out of and supporting me at the beginning stages of this project, Prof. Lijun Chen for assistance in setting up the optimization problem and to my advisor Prof. Brian Argrow for his guidance.

Wind tunnel testing for this project was made possible by the efforts of Steve Semmer And Laura Tudor at the National Center for Atmospheric Research (NCAR) in Boulder.

I am extremely grateful to my fellow RECUV students, Roger Laurence III, Nicholas Campbell, Emily Ranquist, William Silva, Andrew Mills, Katiee Glasheen, Ramya Kanlapuli Rajasekaran, Scott Sheahan and Sangwoo Moon for their assistance and feedback that helped to shape this work. I am also deeply grateful for the support of my family and friends throughout this process.

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

1.1 Introduction

Small unmanned aircraft systems (sUAS) are increasingly used for high-accuracy, in situ wind measurements in the atmospheric boundary layer (ABL) at relatively low cost [19] for path planning [52], [8] or meteorology[56],[25],[27],[4]. With the advantage of three-dimensional mobility, sUAS increasingly complement fixed and mobile mesonets, radar and other land-based systems used for atmospheric sensing in the lower troposphere.

Key to accurate flow measurements from these systems is an accurate measurement of the aircraft-relative flow direction [38]. Currently this measurement is most often made by pitot-static multi-hole probe systems, flow-angle vanes and, sonic anemometers. However these approaches are generally not well suited to sUAS applications. Typically sUAS have poorly understood local flow fields which interfere with the accuracy of probes and vanes that generally do not extend far enough from the aircraft to be relatively free of local effects. Furthermore, the versions of these systems suited to sUAS size weight and power (SWAP) and price constraints generally do not offer sub-degree direction sensing accuracy [13]. These issues are especially impactful when high accuracy vertical and lateral flow components are desired. In these cases, flow direction accuracy is of secondary importance to flow speed accuracy. A need thus exists for flow sensing techniques with high directional accuracy that are suitable for sUAS applications.

This work supports the development of a new class of sensors for sUAS using an

optical particle streak imaging approach. Optical fluid flow measurement methods offer the capability of non-intrusive flow measurements at greater standoff ranges than possible with probes. Additionally the hardware can be safely protected within the vehicle, providing greater survivability. This work mathematically develops a method using particle streaks to determine the relative wind velocity from a fixed-wing sUAS and assesses its performance using simulations validated by wind tunnel testing. The developed technique is primarily geared to flow direction accuracy but also offers an air speed sensing capability.

In general, two classes of optical fluid flow sensing methods exist. The first are Doppler methods. Laser Doppler velocimetery/anemometry (LDV/LDA) and light detection and ranging (LiDAR). These methods rely on the frequency shift of light reflected by fine particulate to generate accurate velocity measurements along a single axis. By measuring over multiple independent axes, a three-dimensional velocity measurement can be made.

The second are particle imaging methods. Traditionally, these methods have been used in indoor environments and wind tunnel tests with particles seeded into the flow to provide imagery. Differing methods are then used depending on the seeding density. Dense particle environments are well suited to imaging clouds of particulate in successive images and correlating a large number of separate interrogation windows across image sequences to derive fine scale 2D velocities. This approach is known as particle image velocimetry (PIV). In sparsely seeded flows individual particles may be tracked over a sequence of images in an approach known as particle tracking velocimetry (PTV). A final approach is to track individual particles by using the streaks they leave in a single long-exposure image. This is known as particle streak velocimetry (PSV).

All of these imaging methods are generally used for 2D tracking though stereo extensions exist to allow for three dimensional velocity measurement. The main objective in these methods is to measure velocity fields which vary over the imaged domain.

Laser Doppler systems exist for making proximal air data measurements from aircraft, however they are currently too large for use in sUAS [54]. As such, this work focuses on an

imaging approach.

The problem of interest in this work is somewhat different than the typical measurements made by optical imaging methods. Rather than determine two-dimensional air velocity at fine resolution across an imaged plane, this work aims to determine a single mean three dimensional velocity within an imaged volume of significant depth. This provides a single measurement of the relative wind, analogous to measurements made by multi-hole probes, air speed sensors paired with vanes, and sonic anemometers in similar applications.

To make this measurement a particle streak approach is used. Streaks are imaged in a deep volume (as opposed to the thin sheets typically used in 2D particle velocimetry methods) allowing streaks to result from motion in all three spatial dimensions. Streaks are identified and, based on their position in the image, their end points are mapped to three-dimensional unit vectors which are used to construct a solution matrix allowing for a linear solution. To make such a measurement it is necessary that the mean velocity of particles within the image dominates turbulent variations in velocity between particles within the imaged volume. The validity of this assumption for small, fixed-wing sUAS is shown in section 2.2

Because artificial flow seeding is not available in the outdoor environment, the aim of this method is to be used with ambient particulate suspended in the atmosphere. A short review of the lower troposphere particulate environment in which sUAS typically operate is given in chapter 2. This review shows that sufficiently large particles may be suspended in sufficient numbers for particle imaging to be a possibility depending on environment and illumination intensity. Imaging capabilities are not investigated in detail, but a limited study of particle densities and scattering characteristics and discussion of the potential for ambient particulate imaging is given in the same chapter.

The proposed method for using the detected particles to make the desired flow measurement is shown in chapter 3. This chapter describes the techniques for identifying particle streaks in images taken in environments with noise and background; using the identified

streaks to estimate flow direction; and estimating the air speed through knowledge of the imaged volume dimensions. Using a simulation that includes realistic camera parameters, background features, noise and turbulence, the proposed method is tested in chapter 4. The results of this simulation are supported by wind tunnel test results shown in chapter 5.

1.2 Prior sUAS Wind Sensing Work

There have been a significant number of studies using sUAS for wind measurement. The methods employed vary widely in cost and quality. A few are summarized below that are broadly representative of the methods currently in use, ranging from relatively high cost solutions using precise multi-hole probes and high-accuracy carrier-phase GPS units, to low cost systems relying simply on measurements of aircraft ground speed and track.

In 2007 a group from the Technical University of Braunschweig and Mavionics GmbH, tested a sUAS wind measurement system using a multi-hole probe with attitude data provided by a GPS-INS system using a carrier phase GPS receiver to compute inertial frame accelerations [56]. Measured winds were compared with ground based SODAR and tower measurements and showed generally good agreement although vertical velocity measurements showed large standard deviations.

A similar mission in 2009, conducted by a group from Technische Universitat Carolo-Wilhelmina Braunschweig, Eberhard Karls Universitat Tubingen, and the German Meteorological Service, using the same hardware showed general agreement with measurements from SODAR, tower, and wind profiler [33]. Wind measurement agreement was not as close as in the prior testing however some of these differences may be explained by the physical separation of the sensor systems and the averaging effects of the ground-based systems.

During the VORTEX2 tornado research mission in 2009 and 2010 a group from the University of Colorado Boulder operated sUAS to collect meteorological data [20]. Wind measurement was not a primary objective of this mission, however wind estimates from the Piccolo autopilot system were recorded. These estimates were made without a complete

air data system and were based on a kinematic model of the aircraft. While these were not validated against another measurement source, they were compared in a simulation environment where results indicated that the autopilot's aircraft-dynamics based estimate provided generally poor performance [18].

The Small Unmanned Meteorological Observer (SUMO) has been used in a number of atmospheric sampling missions [36][47][49]. In order to maintain low cost, it often does not carry an air data system instead making only horizontal wind measurements using a technique that involves determining the wind based on increases and decreases in the aircraft's ground speed on either side of a circular orbit [48]. The approach keeps costs low but reduces data rate significantly and effectively smooths measurements while providing no vertical wind data.

These operations show the real-world viability of sUAS wind measurement but also illustrate the need for low cost relative wind measurement systems that can allow for full three-dimensional wind measurement without the expense and vulnerability of current systems.

1.3 Optical Fluid Flow Sensing Overview

The proposed method is an optical fluid flow sensing approach. Optical fluid flow sensing uses light frequencies in the electromagnetic spectrum that can be observed by cameras (as opposed to radar measurements) to determine relative wind velocities. A number of such approaches have been developed and a brief review is presented here.

1.3.1 Doppler Approaches

Laser Doppler systems rely on a laser light source to illuminate airborne particles and record the Doppler shift of the returned light. The Doppler shift of light reflected by particles in the flow allows for an air speed measurement to be taken along the axis of the laser illumination. By taking three measurements at different angles a full 3D flow velocity

can be determined. Laser doppler measurements are used both in laboratory settings and in outdoor sensing [22],[58]. They do not require actually imaging particles, allowing them to be used to measure winds from air with only very small diameter aerosols.

Laser Doppler Velocimetry and Laser Doppler Anemometry are two names given to a class of sensing techniques generally used for high accuracy laboratory measurements. These use a split laser bear to create an interference or fringe pattern of standing waves in intensity, perpendicular to the measured flow component [16]. Particles passing through this pattern scatter light whose intensity varies with a frequency proportional to the velocity at which they cross the pattern. Figure 1.1 shows a conceptual overview of how the technique works.

Doppler LiDAR uses the Doppler frequency shift of coherent light scattered by ambient particulate to make a one dimensional flow speed measurement [23]. By taking measurements along multiple independent axes, Doppler LiDAR provides a three-dimensional flow sensing capability. Doppler LiDAR systems offer high accuracy and the benefit of being able to take measurements well upwind of the sensor's location but are expensive and somewhat large compared to other wind sensors [29]. Additionally, laser operations from aircraft raise significant regulatory difficulties. Doppler LiDAR systems have been used on sUAS for point cloud sensing but not for wind sensing [57].

One existing airborne Doppler LiDAR system is the Wind Sceptor by Optical Air Data Systems [54]. This system is mainly marketed at full-size helicopter applications. It offers airspeed accuracy of roughly 0.5 m/s and flow angle accuracy of 0.25°. This level of accuracy is superior to that offered by small multi-hole probes but the system is too large for practical use on sUAS.

1.3.2 Imaging Approaches

This class of methods use images of airborne particles to make flow measurements. The proposed method uses this type of measurement because the hardware required can be relatively simple (a camera and illumination source) making them potentially better suited

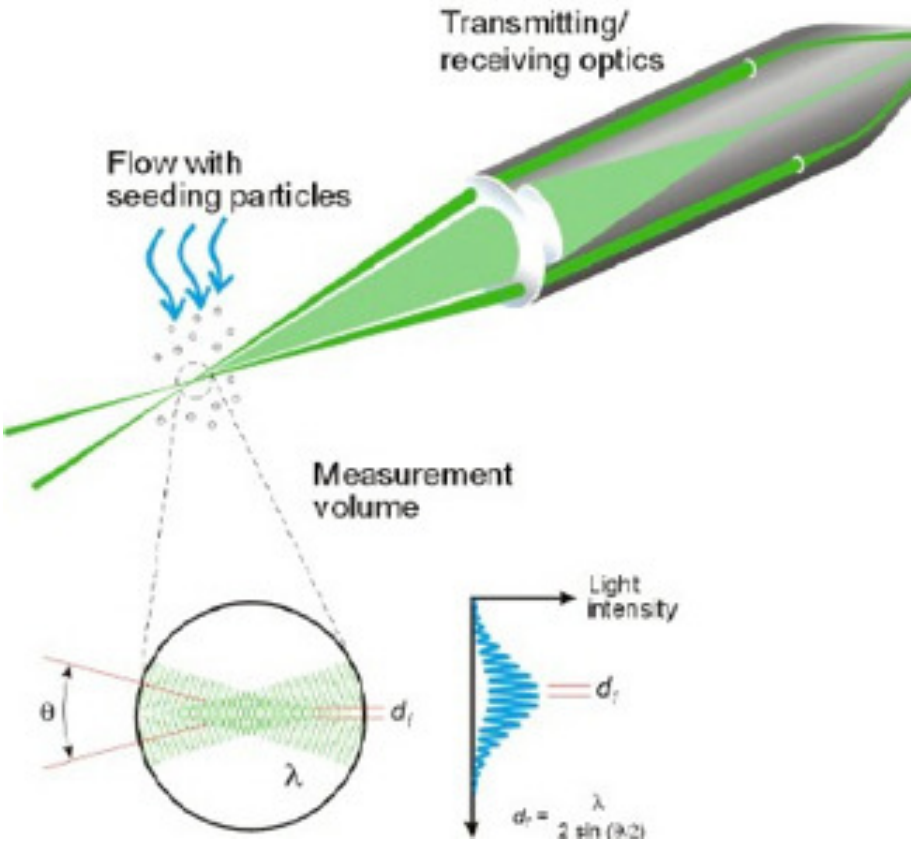


Figure 1.1: Example of an LDA system from [16].

to SWAP and price constraints on sUAS.

1.3.2.1 PIV

Particle image velocimetry (PIV) is a method for determining a flow field by imaging particles added to a fluid flow. PIV is generally conducted in a closed laboratory environment. Typically a laser source is used to illuminate the particles captured by high-speed, high-resolution cameras. Image sequences are then captured in this fashion. The movement of the particles within the plane of the images can then be assessed by correlating various sections of the images in the sequence, interrogation windows, to produce vector fields [2]. A single vector is generated in each interrogation window. With high-resolution cameras, large numbers of interrogation windows can be used allowing for high resolution 2-D fields. Laboratory PIV requires high-end imaging systems and significant computational power. An image of a flow field determined via PIV is shown in figure 1.2.

PIV allows for the sensing of specific structures within the flow in great detail. While generally two-dimensional, three-dimensional extensions are possible through the use of multiple cameras and stereo processing [46]. Multi-camera systems are poorly suited to sUAS applications. The required baseline is difficult to achieve if longer standoff ranges are desired in the measurement and integration becomes far more difficult due to the need to precisely align the cameras. Additionally, one, or possibly both, cameras would need to be placed on the wings of a fixed-wing vehicle. Because the wings flex significantly in flight, maintaining accurate camera alignment would be impossible. Furthermore, sufficient baseline to provide range exceeding the rotor wash on multi-rotor craft may be difficult or even impossible to achieve

One PIV-like approach has been proposed for sUAS by Pope et al. [45]. This approach used a single interrogation window and a small, low-cost camera. Unlike in traditional PIV, two illuminated planes were used. A two image sequence was taken with the first image taken as particles passed through the first illuminated plane and the second image taken as

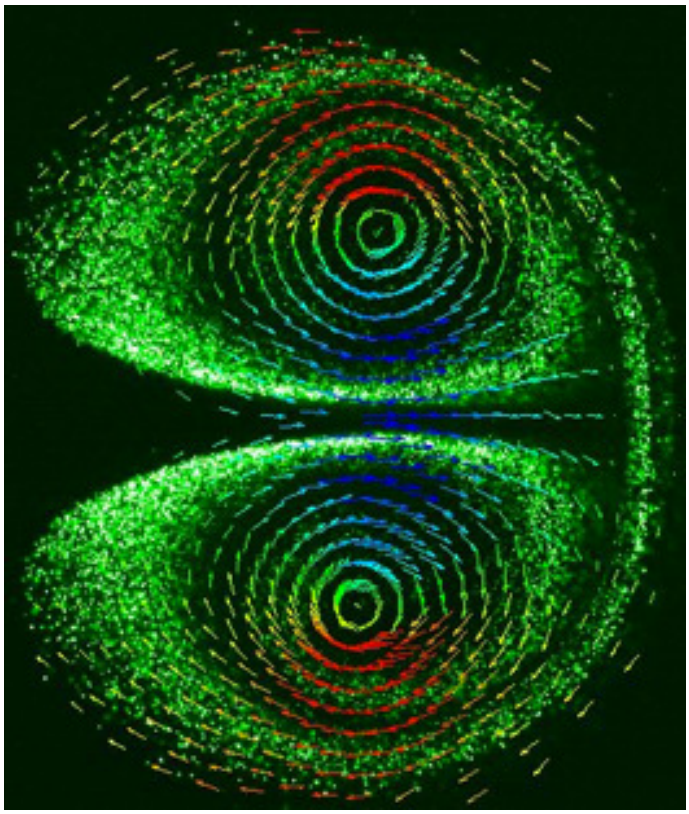


Figure 1.2: Example of a flow field estimated using PIV from [15].

the particles captured in the first image passed through the second plane. Results showed potential for high accuracy lateral flow component sensing, this method however, requires dense particulate clouds to get features for correlation as well as precise out of plane air velocity sensing to allow for the precise timing required to capture the same particles in successive images.

1.3.2.2 PTV

Particle tracking velocimetry (PTV) is similar to PIV but aimed at sparse particulate environments. Rather than use interrogation windows to compare features in particle clouds, PTV tracks the movements of individual particles through a sequence of images. This requires techniques to determine the path of particles through multiple images [41]. Three dimensional PTV is possible using multiple camera systems [31]. However these have the same limitations for an airborne flow sensor as in stereo PIV.

1.3.2.3 PSV

Particle streak velocimetry (PSV) is an imaging technique where a single long exposure image is taken of particles moving in the flow. This generates streaks in the image that can be detected and used to indicate the flow direction and speed.

Such techniques used for indoor air flow tracking were reviewed by Fu et al. [21]. These techniques used relatively large tracer particles, generally helium filled-soap bubbles, to track flows in interior spaces generated by ventilation equipment with multiple cameras, similar to the stereoscopic methods for PTV and PIV.

For a practical airborne system a single camera approach is expected to be superior to a multiple camera approach because of simplicity and lower cost. One such method for three-dimensional sensing using PSV is particle streak velocimetry optical coherence tomography (PSV-OCT), which uses profiles of streak intensities as they cross through a thin imaging volume to determine the out-of-plane velocity component in addition to the

typical two-dimensional components typically detected by PSV [60]. This method allows for a three-dimensional velocity field to be extracted from a single camera perspective, however it requires accurate measurements of streak intensity patterns which is difficult in a high background environment as encountered in the outdoors.

1.4 Differences in the Proposed Method Relative to Prior Art

The systems described in in the preceding sections can potentially provide very high accuracies with relatively simple calibration. However, existing optical wind sensing methods are relatively expensive and may require flow seeding and complex image processing, making them unsuitable for low-cost sUAS applications.

Compared to earlier particle imaging techniques, the method presented here is aimed at capturing accurate overall flow direction, rather than a two-dimensional velocity field. The flows of interest have high out-of-plane velocity components. As such the method is capable of dealing with deep imaging volumes rather than thin illuminated sheets. The new method, which is referred to as particle streak anemometry (PSA) to distinguish it from the velocity field measurements typically made in the various imaging velocimetry techniques, is also geared at imaging in environments with high background signal and heterogeneous particle sizes and species with sparse but variable densities. An illustration of the proposed method compared to classical PSV is shown in figure 1.3.

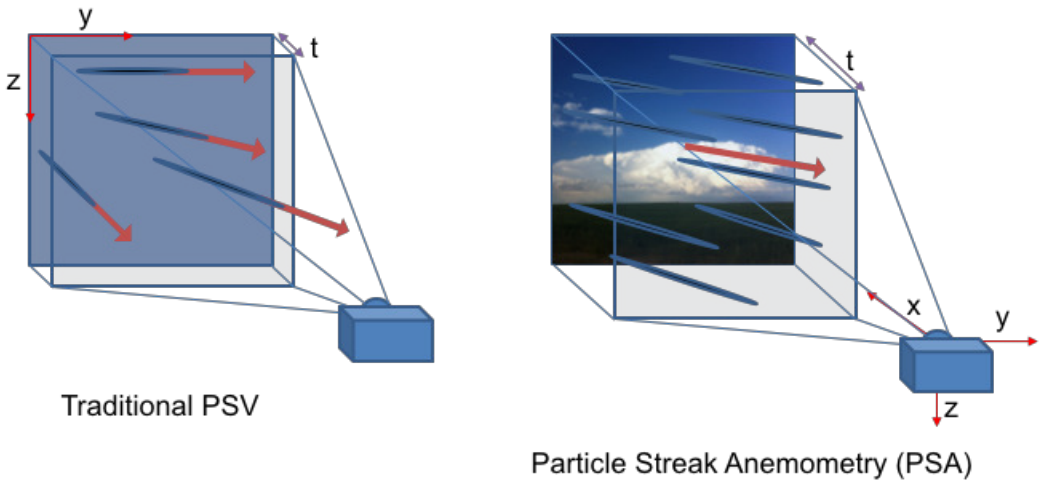


Figure 1.3: Conceptual comparison of PSA (right) to traditional PSV (left).

Chapter 2

Method Feasibility

2.1 Performance Requirements

The proposed system aims to achieve directional performance superior to existing pitot-static air data systems. An example system is the Aeroprobe micro air data system [13]. The published flow direction accuracy of this system is 1° in angle of attack and side slip. Air speed accuracy is shown in figure 2.1. Each curve is the calibration curve for the differential pressure transducer with the transducer range, indicated in the legend.

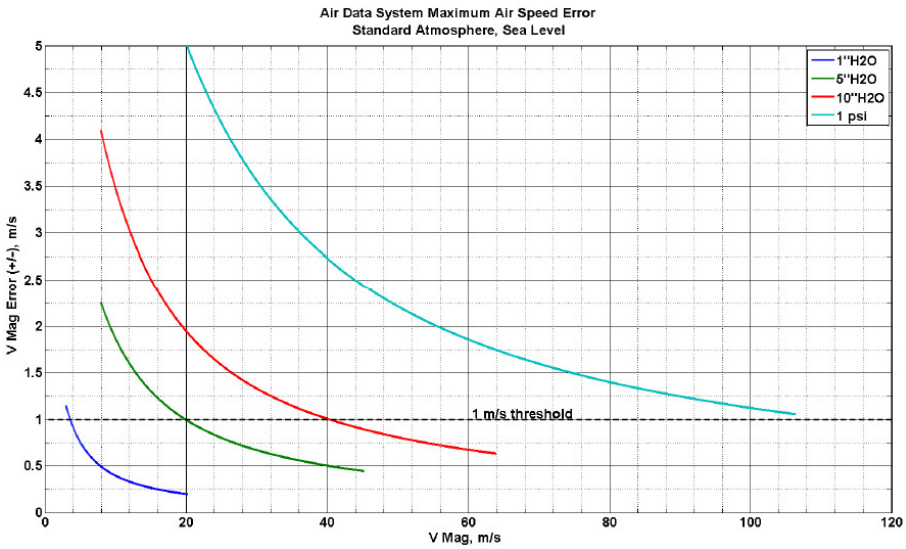


Figure 2.1: Air speed accuracy of commercial Aeroprobe, air data system from [13].

Because a driving motivation in this work is accurate sensing of vertical and lateral flow

components high directional accuracy is of greater importance than high air speed accuracy. For example, an aircraft raveling at 20m/s at a 5° angle of attack experiences a 0.34m/s error in vertical velocity for a 1° angle of attack error but only a 0.09m/s error in that component for a 1m/s airspeed error.

2.2 Validity of the Bulk Velocity Dominance Assumption

For the proposed method to function, it is necessary that particle motions within the imaged volume are dominated by the vehicle-relative air velocity such that turbulent fluctuations between particles play only a small role in determining a particle's trajectory during the exposure period. An analysis for a fixed-wing sUAS follows.

A typical flight airspeed for fixed wing sUAS developed by CU Boulder's Research and Engineering Center for Unmanned Vehicles (RECUV), and the Integrated Remote and In-Situ Sensing Program (IRISS), ranges from about 10m/s to 50m/s. For this analysis it is assumed that the imaged volume has side lengths on the order of 1m (a liberal upper bound for imaged volume given the need to illuminate and capture returns from particles across the extent of the volume).

Kolmogorov's isotropic turbulence theory gives the magnitude of velocity variance over a given distance as

$$(\sigma_{\delta v})^2 \sim C_k \left(\epsilon r\right)^{2/3} \tag{2.1}$$

where $C_k = 1.5$ is the Kolmorgorov constant [40]. Typical values of ϵ in the lower troposphere are below $10^{-1}\mathrm{m}^2/\mathrm{s}^3$ [10] so it is expected that:

$$\sigma_{\delta v} < 0.6\mathrm{m/s} \tag{2.2}$$

As this is between one and two orders of magnitude below the typical air speed, treating the flow as dominated by the relative wind velocity is a reasonable assumption. The impacts of turbulent fluctuations within the volume on the solution quality are shown in greater detail in chapter 4.

2.3 Imaging Feasibility

The proposed method relies on the capability of the sensor to image ambient particles in the atmosphere under powerful illumination. The PSA method detailed in chapter 3, describes the process of going from particle streak images to flow measurement. As such some discussion of the potential feasibility of acquiring such images is necessary.

2.3.1 Particulate Size Requirements

Particle images are captured using bright foreground illumination to create a contrast between scattered light from the particle and the background scene. In general, brighter illumination and enhanced contrast with the background, make particles more visible, giving a high signal to noise ratio against the background scene. Because larger particles generally scatter more light, they can be detected with less illumination power than smaller particles. Thus at a basic level, more illumination power leads to smaller minimum detected particle sizes. Additionally, because the method requires at least two imaged particle streaks to execute, but becomes more accurate with larger numbers of detected particles (chapter 4), the desired accuracy informs the minimum number of desired streaks and thus the choice of illumination power level. Sparse particle environments can potentially be compensated for by imaging larger volumes, although since this generally requires imaging at greater ranges, and reflected light intensity falls off with range, this is not as straightforward a solution as it initially appears.

Generally, as requirements for flow direction and speed accuracy increase, and particle environments become more sparse, smaller particles must be detected and thus brighter illumination must be used. Current expectations, based on the use of consumer grade imaging equipment, are that feasible imaging will target particles on the order of 10μm. While larger concentrations of particles exist at smaller sizes, imaging these from an sUAS is not thought to be practical.

2.3.2 Lower Troposphere Particulate Environment

sUAS typically operate in the lower troposphere. This region contains a wide variety of particles from a diversity of sources [42],[43]. A number size distribution of particles measured in a research flight over the North Pacific is shown in figure 2.2. Particle number densities fall several orders of magnitude between particle sizes on the low end of the size range, on the order of $10^{-2}\mu$m diameter, and those on the high end, on the order of 10μm diameter. This general trend of a steep fall off in number density with size is true for most particle species and environments; however, there may be significant differences in the overall scale of the number concentration (multiple orders of magnitude) [11],[5],[44]. As such, simple imaging may be effective in some environments but not in others.

Figure 2.3 from [42] shows the size ranges of a variety of particle species found suspended in indoor environments. While not all of these species are found outdoors, many clearly originate in the outdoor, lower troposphere environment (such as pollens), and are in suitable size ranges for capture with consumer-grade equipment.

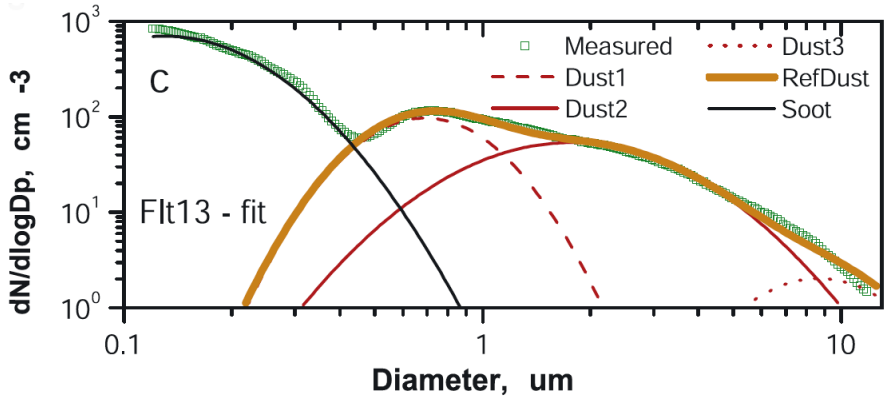


Figure 2.2: Number concentration of airborne particles against particle diameter from [11]. Data measured on a flight over the North Pacific is shown in green.

The larger atmospheric particles can be imaged relatively easily with consumer grade camera equipment. An example demonstrating this is shown in figure 2.4 showing suspended particles in an indoor environment captured with a Nikon D7200 SLR camera and 50mm lens,

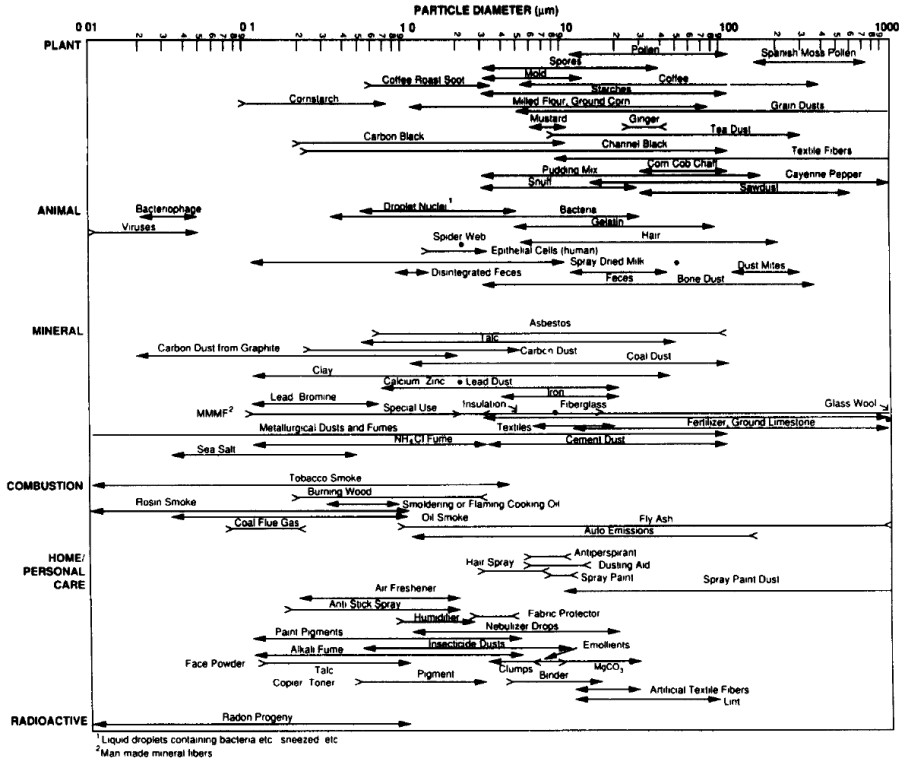


Figure 2.3: Size ranges of various types of airborne articles from [42].

illuminated by a shaft of sunlight against a background in shadow. Generally, consumer-grade digital single-lens reflex cameras (DSLRs) costing under $1000 can resolve detail as well or better than the human eye, which can observe particles down to the order of 10μm, as shown in figure 2.5.

Figure 2.4: Photograph of airborne particles in an indoor environment (the author's bedroom), illuminated by natural sunlight and photographed with a consumer-grade Nikon D7200 DSLR camera.

2.4 Particle light scattering

For particles of similar order of magnitude to the incident light, Mie scattering provides an accepted model of light scattering [14]. Since the particles of interest are only $1-2$ orders of magnitude larger in diameter than incident light wavelengths in the visible and near infrared spectra, where many camera systems function, a Mie scattering model was used to investigate the properties of light scattered by the target particles. The BHMIE code implemented in MATLAB by Christian Maetzler was used to determine Mie scattering coefficients and derive the phase functions for several possible airborne particulate materials

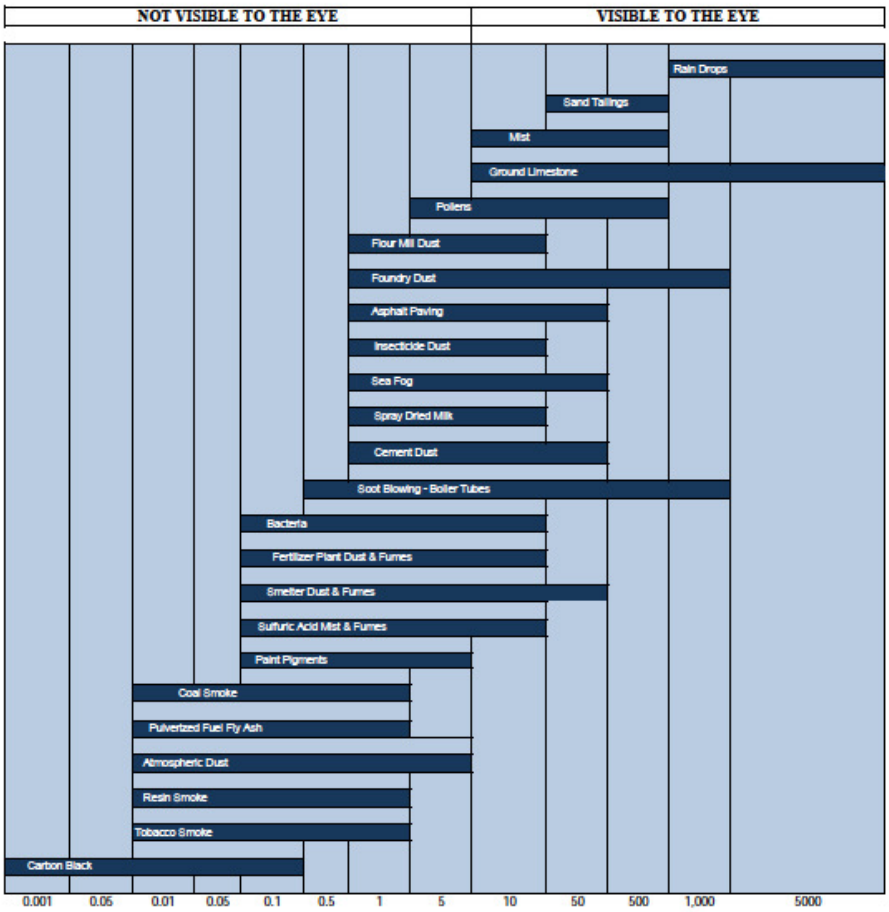


Figure 2.5: Visibility of various types of particles from [26].

[35]. There has been relatively little work done to fully characterize the refractive indicies of pollens so for the current analysis a particle of pure glycerol is simulated [9].

The simple model used here assumes spherical particles which are unlikely in reality. Additionally, soot species are investigated at these larger sizes for comparison but are rarely found in these very large diameters. As a result the outputs of these models should be taken only as loosely suggestive of the true particle scattering behaviors. Due to the complexity of accurate modeling of scattering the necessarily unknown size and shape of particles a definitive model predicting the intensity of light scattered from any given set of ambient particles is not possible, however the simple model used can provide some insights.

Scatter coefficients, representing the effective ratio of scattering cross section to true cross section for a number of materials at three different particle diameters are shown in figure 2.6, plotted against incident light wavelength. It is clear that there is relatively little variation in total scattering intensity between the species tested over the wavelengths of interest. The purpose of this plot is to show the insensitivity of the scatter coefficients to the species and diameter of the particle.

Phase functions, representing the intensity of scatter in different directions are shown in figure 2.7, these are plotted for 500nm incident light onto a 10μm particle diameter. The model shows strongest scattering in the forward direction with significant power also in the lateral direction (around $\pm 90^{\circ}$ scattering angles) for all species. Backscatter intensities vary widely by species, with some providing moderate scatter in the backwards direction and others providing nearly none. Between the backward and lateral direction there is extremely limited scattering intensity. This has potentially important consequences for illumination system geometry, though a full accounting of this is difficult without knowing the specific geometry, size and species of the particles being targeted. Importantly, scattering phase functions vary significantly based on species, particle size and incident light wavelength even though scatter coefficients are broadly similar in the particles examined.

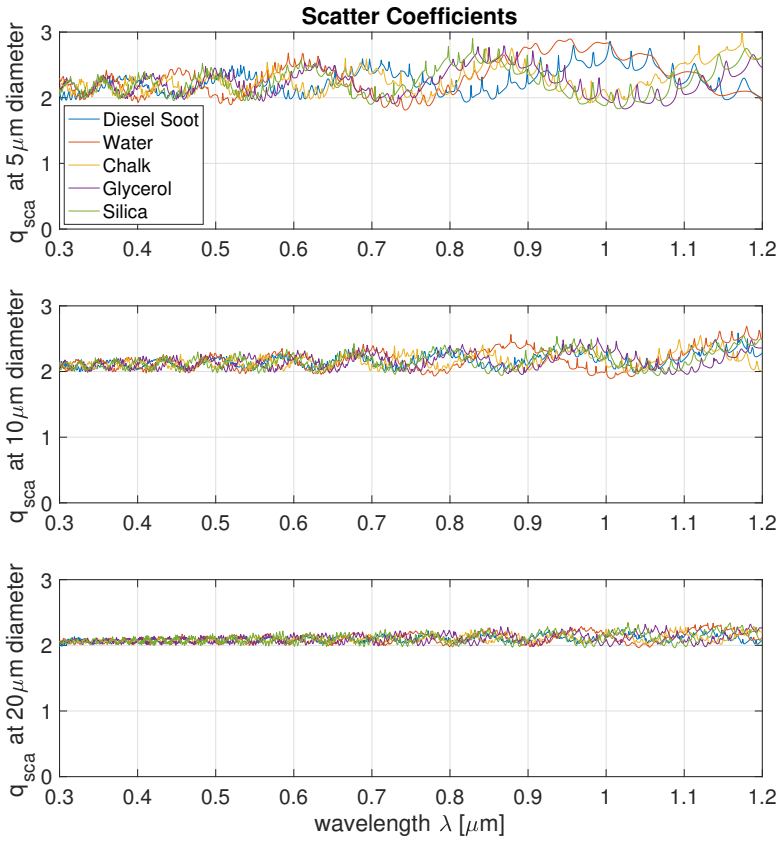


Figure 2.6: Scatter coefficients for several species of airborne particles

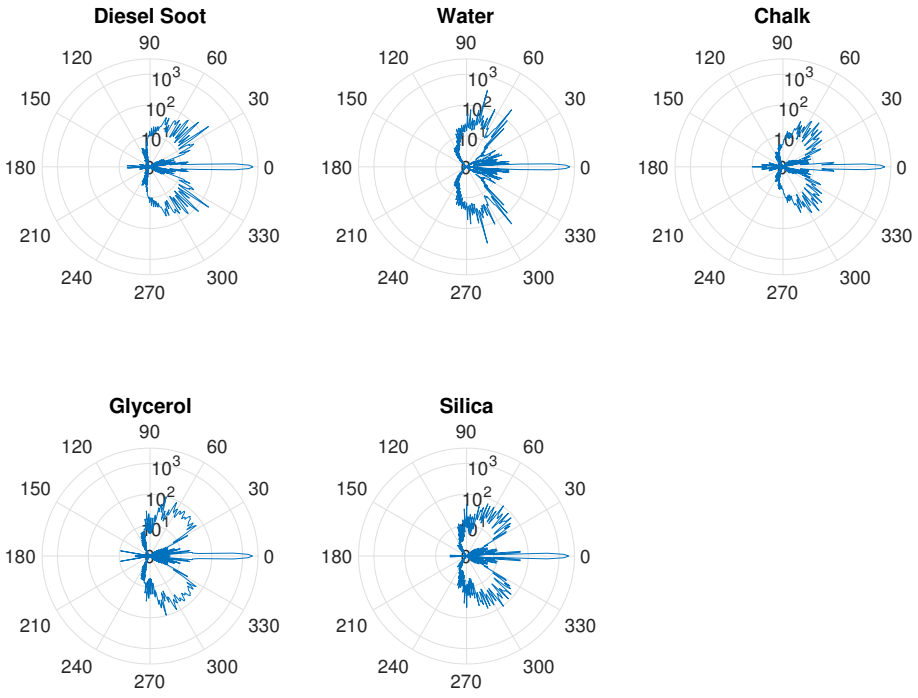


Figure 2.7: Scattering phase functions for several species of airborne particles

2.5 Feasibility Summary

These results indicate a reasonable possibility of capturing ambient outdoor particles on consumer digital imaging equipment in certain environments with a well-designed illumination-imaging system. Furthermore, while the cameras containing this imaging equipment are large for sUAS payloads, much of their weight lies in batteries, cases, LCD screens, viewfinders, auto-focus motors, flashes, and other features not needed for a system that could capture images for the proposed method. As such, it may be possible to construct a camera capable of capturing the imagery needed for the proposed method, using consumer sensors and electronics, that is substantially lighter than existing DSLR cameras.

Chapter 3

Method Description

The PSA method takes as input an image of particle streaks taken against a background, identifies the individual streaks, then estimates first flow direction, and then air speed. In order to get the greatest benefit from the extended standoff sensing range the technique offers compared to probes and vanes, looking forward of the aircraft into the flow is envisioned as the best employment scenario so is the focus of the following analysis and discussion.

The first stage in the method is to capture an image of particle streaks. A light source is applied to the incoming volume of air with the camera shutter open for sufficient time to see particle motions (at typical sUAS flight speeds this is on the order of 10ms). A simulated image of particle motion streaks is shown in figure 3.1.The simulated image is composite of a landscape photograph that is intentionally blurred with diminished intensity and with streaks superimposed to simulate particle traces.

Once an image is captured edge detection is used to identify potential streaks. The edge detection method used here is described in section 3.5.2. The results of this detection performed on figure 3.1 are shown in figure 3.2 which shows a binary image with white regions depicting possible streak edges. The edge detection step produces a binary image showing potential streak regions. These regions are then identified and tagged using connected component labeling (CCL) [51]. Once labeled, a size threshold is applied to reduce the effects of noise and background features.

Figure 3.1: Simulated streak image.

Figure 3.2: Edge detection on simulated image in figure 3.1.

Labeled features then have their endpoints detected using a principal component analysis (PCA) based approach. This step and the preceding CCL step are described in section 3.5.3. Figure 3.3 shows the streaks identified in figure 3.1. With streak endpoints detected the direction solution described in section 3.6 is applied. This method determines a direction unit vector that minimizes the error in a least-squares formulation based on the position of streak end points in the image. The final step is to determine the flow speed. This approach is described in detail in section 3.7.

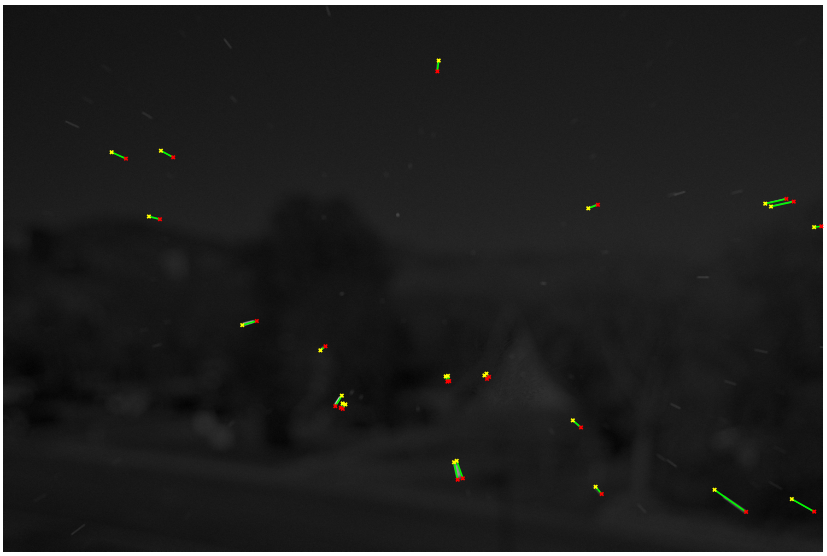


Figure 3.3: Streaks identified from 3.1.

3.1 Coordinate Frames

All vectors are defined in a three dimensional sensor frame where the origin is at the camera optical center. The x axis points forward. In the case analyzed here, this is the flight direction. The y axis points to the right of the image, and the z axis points downward. This axis system is shown in figure 3.4. Positions in an image are defined solely by their $y - z$ coordinates.

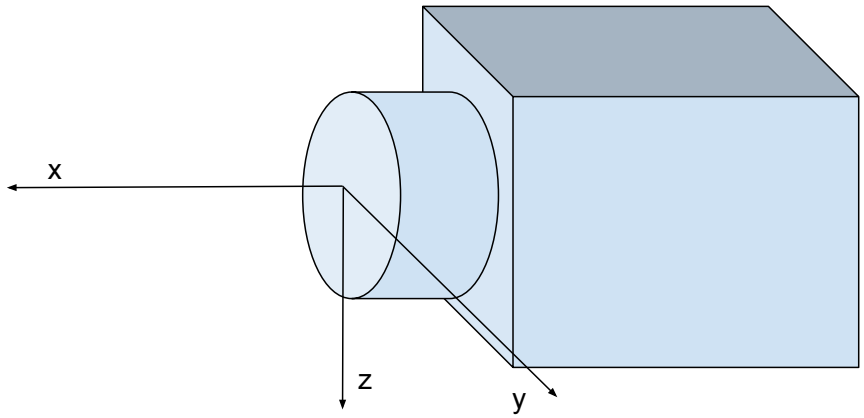


Figure 3.4: Sensor axes used for the method.

3.2 Image Capture

In order to capture images for processing, a three dimensional volume is illuminated for the entire period of the exposure to capture streaks as particles traverse the volume traveling with the flow. Images are captured with sufficient duration to capture particle motion as streaks.

The method is extremely flexible in terms of illumination type and geometry. While a wide-band, incoherent light source was used for wind tunnel testing, it is expected that a coherent, narrow band source with band-limited imaging, would be more appropriate in the outdoor environment to limit the influence of background imagery. Imaged volume edges may be defined either by the imaging system or by the limits of illumination. Generally illumination should cover the entire image to the y and z axis limits, though this is not necessary for a solution to be possible. Imaged space that is not illuminated, adds increased computational load with no performance benefit.

It is however necessary to set x axis limits, depth limits, on the imaged volume. Knowing these limits is vital to determination of the flow speed. This is explained in section 3.7.

For a monostatic sensor configuration, camera and illumination source co-located as shown in figure 3.6, the image depth is most practically set using the camera's focal depth. In this case a camera with a larger aperture will blur particles so much at ranges significantly longer and shorter than the focal distance that they will not be detectable despite being illuminated. This case will be referred to as *focus-limited*. A simulation of a focus-limited image is shown in 3.5.

In this focus-limited case the actual depth of feasible imaging can be estimated using depth of field relations [12]. The hyperfocal distance H is:

$$H = f + \frac{f^2}{Nc} \tag{3.1}$$

for focal length f, f-number N, and circle of confusion diameter c. The f-number is the ratio

Figure 3.5: Simulated image of streaks in a focus-limited image.

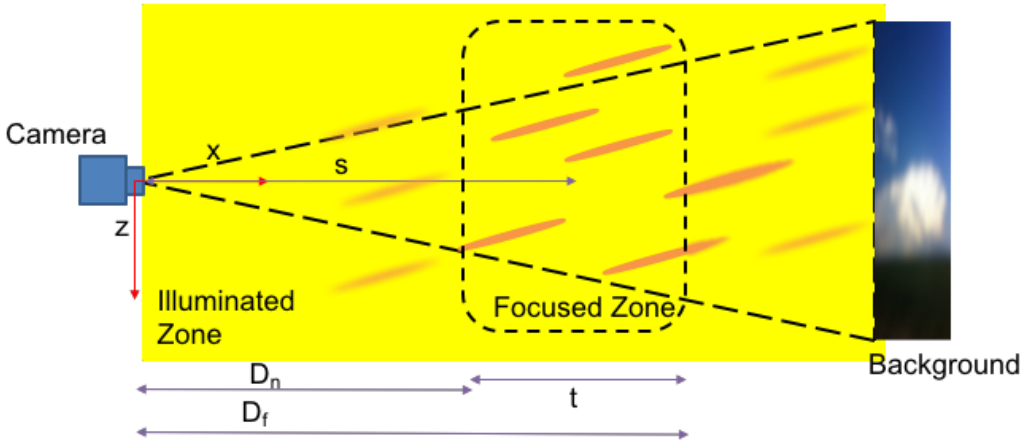


Figure 3.6: Diagram of monostatic configuration.

of focal length to aperture diameter. The circle of confusion is the acceptable diameter of a circle over which a feature can be blurred and still considered in focus. This is measured in length units on the camera sensor. For the specific case used here, the circle of confusion is defined by finding the largest blurred streak width that can still be detected using the streak identification method described here. In practice, this streak identification method detects streaks blurred out to diameters of roughly 0.1mm on the sensor (roughly 25 pixels on the Nikon APS-C sensor used in chapter 5). Some variation exists based on light return strength of the particle.

Based on the hyperfocal distance, the near and far limits, D_n and D_f, of the focused volume (the volume where any sharp feature will be defocused by no more than the circle of confusion) is then given by the following relations:

$$D_f = \begin{cases} \frac{Hs}{H-s}, & \text{if } s < H \\ \infty, & \text{otherwise} \end{cases} \tag{3.2}$$

$$D_n = \frac{Hs}{H+s} \tag{3.3}$$

where s is the distance to the plane of focus, a camera setting. This relation provides a method to determine the volume limits for the focus limited case.

A possible alternative to this method, for the focus limited case, is direct measurement. This requires having a target with features similar to expected particle sizes and then moving this target away from the focused subject distance s until the features are blurred beyond the abilities of the streak identification technique used here. This approach requires taking a large number of images with precise measured distances and creating a suitable simulated target. While it was not attempted in this work using real hardware, it was used in simulation. A comparison between the calculated depth limits, measured depth limits, and illumination-limited case is shown in chapter 4.

While the light return strength will fall off with distance, and could conceivably be used with an infinite depth of field, this is a poor method because in an uncontrolled particulate

environment, larger and/or more reflective particles will be visible at greater distances and the distance limit is hard to predict.

An alternative approach is a bistatic sensor configuration, with the camera and illumination sources physically separated. In this configuration, the image depth can be set by limiting the illumination depth. A schematic of a bistatic configuration is shown in figure 3.7. Limited illumination depth provides an advantage in creating sharper streak ends than the monostatic configuration and more clearly defined near and far image limits, however it requires that sensor platform allow for a bistatic configuration and good control over the illumination edges. A simulated particle streak image with an illumination limited image depth is shown in figure 3.8.

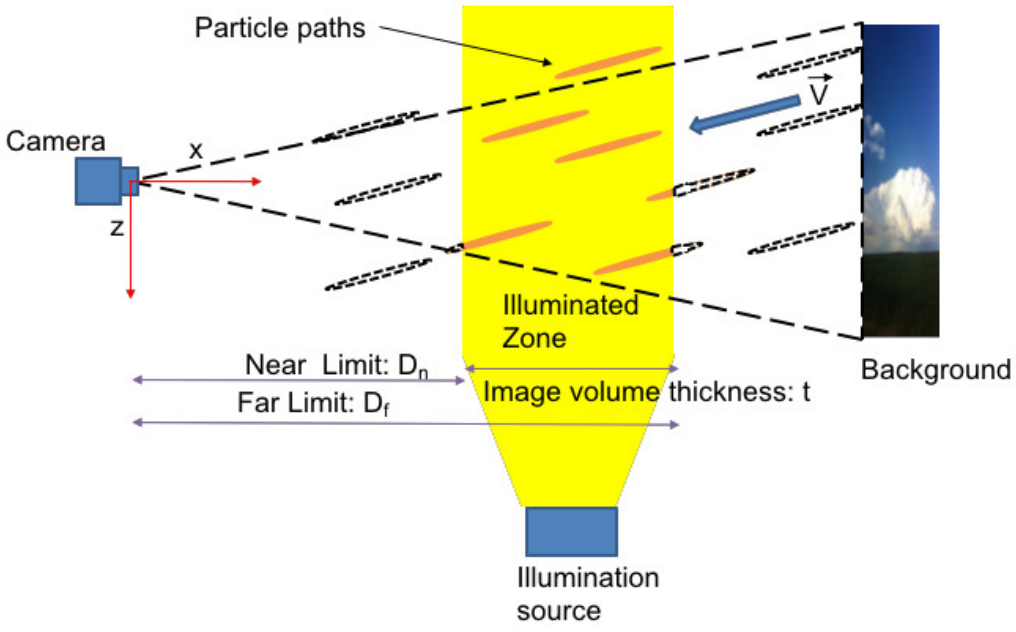


Figure 3.7: Diagram of bistatic configuration.

Figure 3.8: Simulated image of streaks in an illuminated-limited image.

3.3 Imaging System Trade-Offs

The imaging system parameters may be adjusted to achieve various results. A limited explanation of the trade-offs of these choices follows.

To get good streak returns the camera must give bright enough images to show returns from the particles. At the same time, increasing brightness of the overall image also increases the background brightness, so increasing image brightness can make the streak signals difficult to detect. There are two methods available to make an image brighter without altering exposure time.

The first is opening the camera aperture wider, decreasing the lens f-number. This brightens the image but it also narrows the depth of field. This is helpful to an extent, both for depth-limited processing (described later in this chapter) and to blur out background features, however, if the depth of field is too small, streaks will pass through the entire focused volume before the exposure time elapses, thus making accurate air speed solutions impossible. At the opposite extreme, increasing the f-number too much makes the streak returns too dark to detect, and may make background features sharper, leading to some being interpreted as particle streaks, potentially causing flow estimation errors.

The second method for improving brightness is to increase the image sensor sensitivity, or ISO number. This does not affect focus, but increasing it adds additional noise to the image. Additional noise makes the edge detection steps used for streak identification difficult.

As discussed in chapter 2.3.2, large particles are sparse in the lower troposphere. To capture more particles for a better solution, the imaging volume can be increased. To do this, there are three methods available. The first is to widen the image by shortening the focal length. This also increases the depth of field, further increasing the imaged volume but also sharpening any background imagery unless the f-number is reduced (aperture size increased). The second method is simply to increase the f-number with no other changes, this increases depth of field and thus image volume. This has the consequences previously

described (dimmer image, sharper background). The final method is to increase the subject distance, the range to the plane of focus. This increases image volume by both widening the imaged area (as the field of view is angular) and increasing the depth of field (see equation 3.2). However, it also moves the imaged volume farther from the camera, this makes particle returns weaker as light returns fall off with distance and again, sharpens the background.

Another key factor in imaging is the sensor size. The size of the sensor affects the field of view (shown under the pinhole model in equation 3.25). Smaller sensors narrow the field of view, while larger ones widen it. Larger sensors require lenses that can cover the entire sensor with light. This generally increases lens size and cost. The sensor pixel density also effects performance. Higher pixel densities improve sharpness down to the airy disk radius (see equation 4.1), improving the ability to resolve streaks. Again, these sensors bring higher costs.

3.4 Background Suppression

As described in section 3.5.2, edge detection is used to identify particle streaks in images. As such, it is important not only that particles are illuminated brightly enough to show up against a bright background, but that background features not dominate streaks in the streak identification steps. In order to accomplish this, the proposed method uses a shallow depth of focus to keep background features blurred so they do not create strong gradients that would potentially be detected as streaks. This shallow depth of field can also be used for focus-limiting the depth of the imaged volume.

Further approaches to reduce background signal in captured images for a potential practical field capable system are discussed in chapter 6.

3.5 Streak Detection

3.5.1 Overview

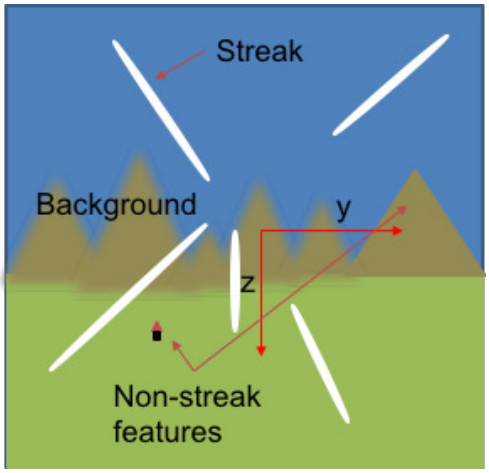
Once a suitable image has been captured the next stage of the method is to detect the endpoints of streaks in the image. Potential streak regions are identified by an edge detection process. The end points of streaks are then identified by a combination of connected component labeling and principal component analysis.

Figure 3.9 shows a conceptual overview of the streak identification process. Starting from a base image shown in figure 3.9a, an edge detection process, described in section 3.5.2, is used to produce a binary edge image, shown in figure 3.9b. Note that while most non-streak features are eliminated, some remain. Connected component labeling is then used to identify individual features, as shown in figure 3.9c. A pixel size threshold is applied at this stage to eliminate features likely to small to be streaks. Finally, principal component analysis is performed and used to identify the end points of probable streaks, as shown in figure 3.9d. Note that large non-streak features that are sufficiently sharp may make it through this process as shown. These are dealt with using the random sample consensus technique described in section 3.6.

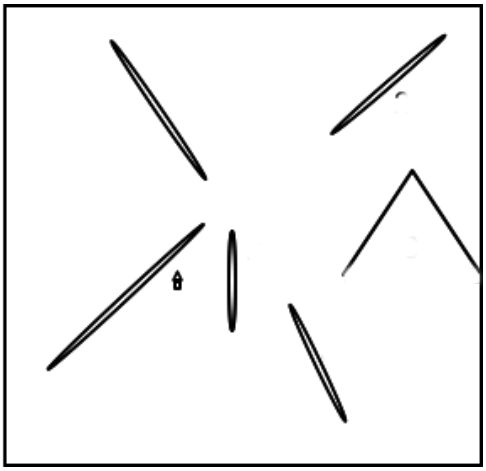
3.5.2 Edge Detection

While many PSV approaches use intensity thresholding to locate streaks [60],[32], this does not work well in cases with strong background signal, as can be expected when operating in outdoor environments under direct sunlight. Instead, PSA detects streaks by finding their edges in an image. This allows for detection as long as the streaks create sufficiently strong gradients in the image for background detection, which can be achieved even with significant amounts of background signal (simulation of the effect of increased background signal with this edge detection approach is shown in chapter 4).

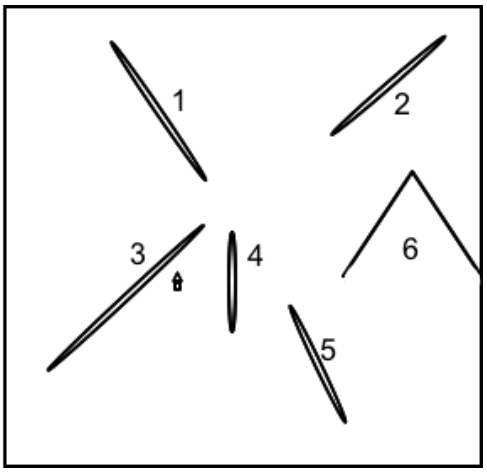
To detect edges that may be streaks, a modified Canny edge detector is used [7]. The



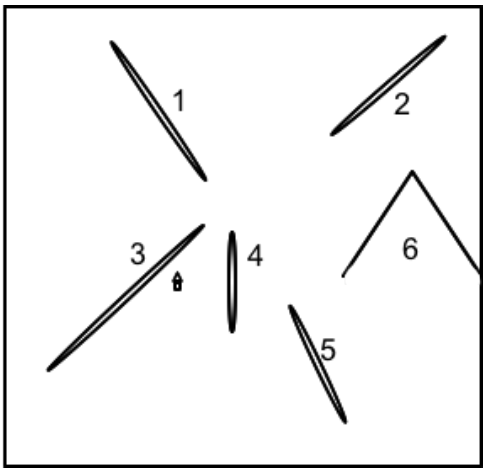
(a) Initial streak image.



(b) Edge detection.



(c) Connected component labeling.



(d) End point identification via principal component analysis

Figure 3.9: Conceptual summary of the streak detection technique.

image is first smoothed to reduce noise using a Gaussian filter with kernel size 2, and $\sigma = 1.4$. This kernel $K_g^{5\times5}$ is convolved with the original image I to give the smoothed image I_s.

$$K_g(i,j) = \frac{1}{2\pi\sigma^2} e^{\frac{(i-(k+1))^2+(j-(k+1))^2}{2\sigma^2}} \tag{3.4}$$

$$I_s = K_g * I \tag{3.5}$$

Gradients in the smoothed image are then measured with the Sobel operator [50],[53]. This relies on two kernels, one for each direction in the image, K_{sx} and K_{sy}. These are convolved with the smoothed image to give the gradient matrices G_x and G_y.

$$K_{sx} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \tag{3.6}$$

$$K_{sy} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \tag{3.7}$$

$$G_x = K_{sx} * I_s \tag{3.8}$$

$$G_y = K_{sy} * I_s \tag{3.9}$$

Here G_x and G_y are the components of the image intensity gradient vector at the corresponding points in the image matrix. Gradient magnitude and direction matrices can then be determined.

$$G = \sqrt{G_x^2 + G_y^2} \tag{3.10}$$

$$\Theta = tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{3.11}$$

Edges are identified as regions of strong gradient or regions of weaker strength bordering strong gradients through a process called double thresholding. Strong gradients are identified as areas above k_{strong} and stored in the binary matrix B_{strong} while weak gradients to be

counted only in proximity to strong gradients are identified as areas of gradient strength below k_{strong} but above k_{weak} and stored in the binary matrix B_{weak}.

$$B_{strong_{ij}} = \begin{cases} 1, & \text{if } G_{ij} \geq k_{strong} \\ 0, & \text{otherwise} \end{cases} \tag{3.12}$$

$$B_{weak_{ij}} = \begin{cases} 1, & \text{if } k_{strong} \geq G_{ij} \geq k_{weak} \\ 0, & \text{otherwise} \end{cases} \tag{3.13}$$

Once strong gradient areas are identified, a mask is created by convolving the kernel K_{near} with B_{strong}. All non-zero elements of the resulting matrix are set to 1 and zero elements left as 0 to create the mask M_{near}.

$$K_{near} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \tag{3.14}$$

$$N = K_{near} * B_{strong} \tag{3.15}$$

$$M_{near_{ij}} = \begin{cases} 1, & \text{if } N_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.16}$$

This mask is then multiplied element-wise with B_{weak} to yield a binary matrix B_{near} in which every non-zero element represents a weak gradient within two pixels of a strong gradient point. While the Canny edge detector generally only counts such gradients in direct contact with a strong gradient, it was found in this work, that for the streak detection problem in the presence of background, a two-pixel radius was more effective at capturing complete streaks without gaps in the detected edge.

$$B_{near} = M_{near} \circ B_{weak} \tag{3.17}$$

Here $\circ$ is the Hadamard product operator (element-wise multiplication). Adding B_{near} to B_{strong} and setting all non-zero elements to 1 gives the final binary matrix B showing all detected edge regions.

$$C = B_{near} + B_{strong} \tag{3.18}$$

$$B_{ij} = \begin{cases} 1, & \text{if } C_{ij} \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.19}$$

The standard Canny edge detector would then perform an edge-thinning step using non-maximum suppression. This would attempt to thin edge regions to a single pixel in width. For the specific problem of streak identification, this step proved to be unnecessary and counterproductive. As described in section 3.5.3, the streak end identification depends on the distribution of streak points and edge thinning does not provide any appreciable benefits in this process. Furthermore, edge thinning often creates gaps in the detected streaks that make it much harder to identify streaks via connected component analysis (described section 3.5.3). Finally the non-maximum suppression step proved to be the most computationally expensive element of the edge detection sequence and, as such, skipping it provides a significant boost in performance.

Figure 3.2 shows the results of the edge detection method applied to the simulated streak capture image in figure 3.1.

3.5.3 Streak Identification

With edge regions identified, it is possible to locate the streak ends. A connected component analysis is performed on B to identify individual features. The MATLAB function *bwconncomp* was used to carry out this analysis for eight-connected regions [34]. This function operates on binary images such as those that result from the edge detection step outlined in section 3.5.2.

To determine connected components, *bwconncomp* runs through all true pixels from the edge detection binary image output until it finds an unlabeled one. This pixel is then given the next label value and used as the start point for a flood-fill algorithm that fills all connected pixels with the same label.

Connected components are defined as lists of pixel positions within B. In order to limit analysis to actual streaks and reduce the impact of background features and noise that may have made if through the edge detection step, only connected regions above a threshold size are retained. The best value for this threshold is dependent on imaging system parameters. In general, the threshold should be large enough to eliminate small features, and background elements but small enough to retain most streaks.

Once connected components are identified and down-selected, a principal component analysis is performed on the remaining connected components [59]. This identifies the principal axes of each identified region. Principal component analysis (PCA) is a process that determines the vectors that align with the greatest variances in a set of data. The first component determined by the PCA is the component with the greatest variance, the second component is the second greatest variance and so on. For this application only the first component is needed.

This first component vector is referred to as the first loading vector, $\vec{w}$. It must satisfy the following relation.

$$\vec{w} = \underset{|\vec{w}|}{\operatorname{argmax}} \sum_{i} \left(X_{p(i)} \cdot \vec{w} \right)^2 \tag{3.20}$$

Here X_p is a $l \times 2$ matrix where l is the number of points in the connected component. This matrix contains the image positions of all the points in the connected component. $X_{p(i)}$ is the i^th row of the matrix.

The MATLAB function *pca* provides a solution to this problem as an orthogonal axis system where the first axis is this weighting vector. The output of this analysis is a 2×2 matrix R of unit vectors for the principal axes of each connected region. This matrix can be

used to transform the pixel positions stored as a $m \times 2$ matrix P from the image axes into the local principal axes. The position list in the principal axis frame is P_p. This rotation is performed for each component identified. For every connected element $i \in (1:n)$ where n is the total number of connected elements

$$P_p^i = P^i R^i \tag{3.21}$$

The ends of the i^{th} streak are now easily identified as the points on the principal axis with the largest and smallest values in the first column of P_p^i which corresponds to the longest principal direction of the identified feature. The indices of these points are now used to identify the end points of the streak in the image frame by finding the elements of P^i with the same indices. Figure 3.10 illustrates the image and principal axes for a single streak.

This process yields a list of streak endpoint pairs for each identified feature. These identified end points are used to estimate the flow velocity as described in the following sections.

3.6 Direction Determination

With streak ends identified, the flow direction can be estimated. To do this, a conceptual understanding of the imaging process is needed. A pinhole camera imaging model is used in this analysis. A diagram of this is shown in figure 3.11[24].

In this model, if an imaged object is at location (x_o, y_o, z_o), the corresponding displacement to the location of the object in the image plane, with the image flipped so that up in the image corresponds to up in the world and right in the image corresponds to right in the world, is then.

$$\vec{h} = \frac{f_d}{x_o} \begin{bmatrix} -x_o \\ y_o \\ z_o \end{bmatrix} \tag{3.22}$$

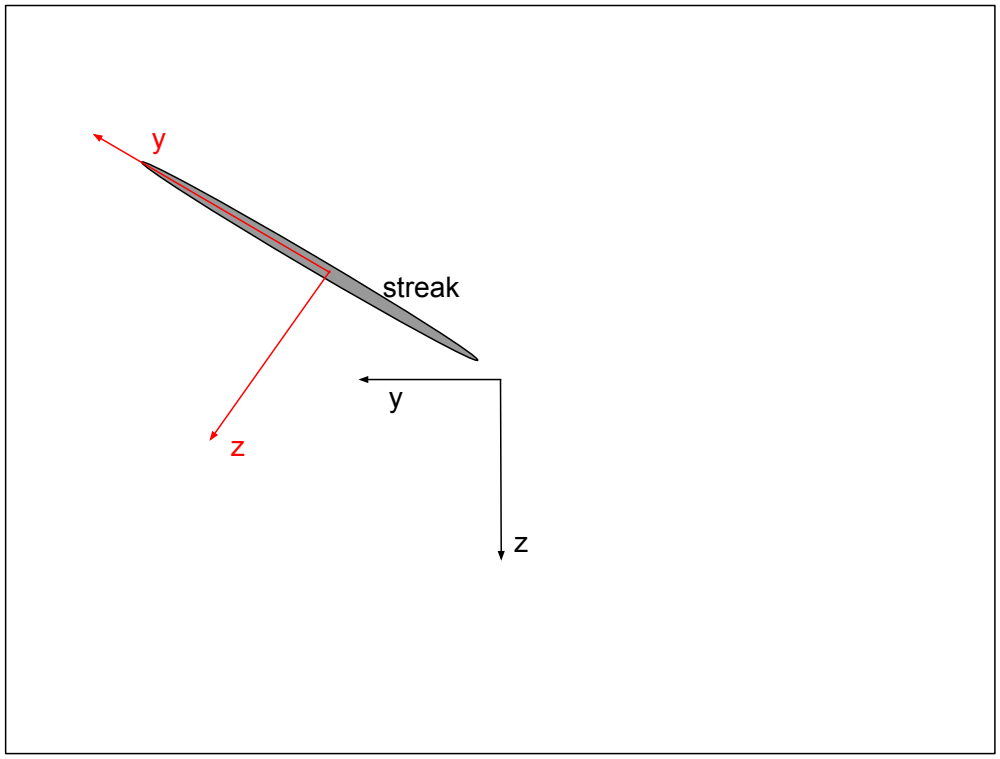


Figure 3.10: Diagram of principal component axes and image axes with principal component axis system shown in red.

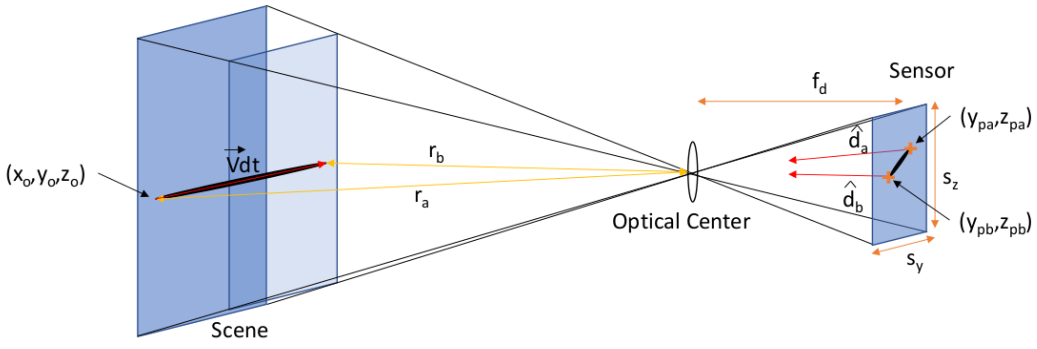


Figure 3.11: Diagram of camera pinhole model.

Here f_d is the focal length (distance to the sensor plane from the optical center). The result is that the $y-z$ coordinates in the image of the object are

$$y_i = y_o \frac{f_d}{x_o} \tag{3.23}$$

$$z_i = z_o \frac{f_d}{x_o} \tag{3.24}$$

where y_p and z_p are the coordinates of the resulting point in the image plane. Generally, points in the image are measured in units of y and z pixels while points in the world are measured in meters. Thus scale factors must be applied to convert from pixels to length units.

$$y_p = y_o \frac{p_y}{s_y} \frac{f_d}{x_o} \tag{3.25}$$

$$z_p = z_o \frac{p_z}{s_z} \frac{f_d}{x_o} \tag{3.26}$$

where y_p and z_p are the coordinates of the imaged object in the resulting image in pixels. Expressed as a function of $\vec{d}$ with the output as a vector, equation 3.25 becomes

$$\vec{P_i} = \begin{bmatrix} 0 \\ y_o \frac{p_y}{s_y} \\ z_o \frac{p_z}{s_z} \end{bmatrix} \frac{f_d}{\vec{x_o}} \tag{3.27}$$

Allow the displacement to the object position to be $\vec{p_o}$. Then the image displacement function can be expressed as a function.

$$\vec{P_i} = \mathrm{Im}\left(\vec{p_o}\right) \tag{3.28}$$

Here $\vec{P_i}$ is a vector displacement to the point in the image, expressed in pixels which may not be square ($\frac{p_y}{s_y} \neq \frac{p_z}{s_z}$) Equation 3.27 allows a position in the world to be converted to a pixel position in the image. Here p_z and p_z are the y and z pixel dimensions of the image sensor respectively while s_y and s_z are the dimensions of the sensor in in meters.

It is relatively simple to reverse the mapping. Because the image is two dimensional, the reverse mapping generates only direction information which can be stored in a unit vector

$\hat{d}$. For a point in the image (y_p, z_p), the reverse mapping is

$$\vec{d_i} = \begin{bmatrix} f_d \\ y_{pi}\frac{s_y}{p_y} \\ z_{pi}\frac{s_z}{p_z} \end{bmatrix} \tag{3.29}$$

$$\hat{d}_i = \frac{\vec{d_i}}{|\vec{d_i}|} \tag{3.30}$$

Using equation 3.29, any point in the image can be mapped to a unique unit vector in three space.

If a particle moves from position p_a to position p_b with velocity $\vec{V}$ (the flow velocity) over exposure time Δt and leaves a streak in the image its motion can be described as follows. Define the unit vector pointing from the image axis center in the direction of p_a as $\hat{d_a}$ (Determined from equation 3.29 for the pinhole model used here) and the vector pointing from the axis center in the direction of p_b as $\hat{d_b}$. Then the displacement to p_a from the axis center is $\hat{d_a}r_a$ and the displacement to p_2 is $\hat{a}_b r_b$ where r_b and r_b are the scalar ranges to p_a and p_b respectively. Therefore

$$\hat{d_b}r_b = \hat{d_a}r_a + \vec{V}\Delta t \tag{3.31}$$

$$\vec{0}^{3\times 1} = \hat{d_a}r_a + \vec{V}\Delta t - \hat{d_b}r_b \tag{3.32}$$

Equation 3.32 is the fundamental relation used in the determination of the velocity vector direction.

Note that a single streak gives a system of three equations for five unknown quantities (the three components of $\vec{V}$, r_1, and r_2). Adding another streak; with the first streak terms now defined by subscript 1 and the second by subscript 2; adds an additional two terms for the two streak end ranges to the system.

$$\hat{d}_{b1}r_{b1} = \hat{d}_{a1}r_{a1} + \vec{V}\Delta t \tag{3.33}$$

$$\hat{d}_{b2}r_{b2} = \hat{d}_{a2}r_{a2} + \vec{V}\Delta t \tag{3.34}$$

$$\vec{0}^{3\times1} = \hat{d}_{a1}r_{a1} + \vec{V}\Delta t - \hat{d}_{b1}r_{b1} \tag{3.35}$$

$$\vec{0}^{3\times1} = \hat{d}_{a2}r_{a2} + \vec{V}\Delta t - \hat{d}_{b2}r_{b2} \tag{3.36}$$

Rewriting this as a linear system gives:

$$\vec{0}^{6\times1} = \begin{bmatrix} \Delta t & 0 & 0 & \hat{d}_{a1}(1) & -\hat{d}_{b1}(1) & 0 & 0 \\ 0 & \Delta t & 0 & \hat{d}_{a1}(2) & -\hat{d}_{b1}(2) & 0 & 0 \\ 0 & 0 & \Delta t & \hat{d}_{a1}(3) & -\hat{d}_{b1}(3) & 0 & 0 \\ \Delta t & 0 & 0 & 0 & 0 & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) \\ 0 & \Delta t & 0 & 0 & 0 & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) \\ 0 & 0 & \Delta t & 0 & 0 & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) \end{bmatrix} \begin{bmatrix} \vec{V} \\ r_{a1} \\ r_{b1} \\ r_{a2} \\ r_{b2} \end{bmatrix} \tag{3.37}$$

$$\vec{0}^{6\times1} = Fy \tag{3.38}$$

A has six rows so $r_A \leq 6$, where r_A is the rank of A. The zero vector $\vec{0}^{6\times1}$ lies in the column space of F (the zero vector lies in the column space of any linear mapping) so a solution exists. y has 7 unknowns but the rank of F is ≤ 6 so the dimension of the right null space of F is ≥ 1. As such non-trivial solutions to $\vec{0}^{6\times1} = Fy$ exist. Thus there are infinite non-trivial solutions to equation 3.37.

To resolve the issue, divide all terms by r_{a1}:

$$\vec{0}^{6\times1} = \begin{bmatrix} \Delta t & 0 & 0 & \hat{d}_{a1}(1) & -\hat{d}_{b1}(1) & 0 & 0 \\ 0 & \Delta t & 0 & \hat{d}_{a1}(2) & -\hat{d}_{b1}(2) & 0 & 0 \\ 0 & 0 & \Delta t & \hat{d}_{a1}(3) & -\hat{d}_{b1}(3) & 0 & 0 \\ \Delta t & 0 & 0 & 0 & 0 & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) \\ 0 & \Delta t & 0 & 0 & 0 & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) \\ 0 & 0 & \Delta t & 0 & 0 & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) \end{bmatrix} \begin{bmatrix} \vec{V} \\ r_{a1} \\ r_{b1} \\ r_{a2} \\ r_{b2} \end{bmatrix} \frac{1}{r_{a1}} \tag{3.39}$$

Now shift the terms multiplying r_{a1} to the LHS:

$$\begin{bmatrix} -\hat{d}_{a1} \\ \vec{0}^{3\times 1} \end{bmatrix} = \begin{bmatrix} \Delta t & 0 & 0 & -\hat{d}_{b1}(1) & 0 & 0 \\ 0 & \Delta t & 0 & -\hat{d}_{b1}(2) & 0 & 0 \\ 0 & 0 & \Delta t & -\hat{d}_{b1}(3) & 0 & 0 \\ \Delta t & 0 & 0 & 0 & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) \\ 0 & \Delta t & 0 & 0 & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) \\ 0 & 0 & \Delta t & 0 & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) \end{bmatrix} \begin{bmatrix} \vec{V} \\ r_{b1} \\ r_{a2} \\ r_{b2} \end{bmatrix} \frac{1}{r_{a1}} \tag{3.40}$$

Now let $r_{a1} = 1$:

$$\begin{bmatrix} -\hat{d}_{a1} \\ \vec{0}^{3\times 1} \end{bmatrix} = \begin{bmatrix} \Delta t & 0 & 0 & -\hat{d}_{b1}(1) & 0 & 0 \\ 0 & \Delta t & 0 & -\hat{d}_{b1}(2) & 0 & 0 \\ 0 & 0 & \Delta t & -\hat{d}_{b1}(3) & 0 & 0 \\ \Delta t & 0 & 0 & 0 & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) \\ 0 & \Delta t & 0 & 0 & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) \\ 0 & 0 & \Delta t & 0 & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) \end{bmatrix} \begin{bmatrix} \vec{V} \\ r_{b1} \\ r_{a2} \\ r_{b2} \end{bmatrix} \tag{3.41}$$

(3.42)

This leaves a standard linear equation of the form:

$$\vec{b} = M\vec{x} \tag{3.43}$$

3.6.0.1 Direction Solution Considerations

The result of the above is a linear system with a unique solution, when M has full rank, where all solution distance units are r_{a1}. To see the cases where M is not full rank, consider the determinant of M.

$$det(M) = \frac{\Delta t^3 \, f_d \, s_y \, s_z \, {p_y}^3 \, {p_z}^3 \, \left(y_{a2}\, z_{b1} - y_{b1}\, z_{a2} - y_{a2}\, z_{b2} + y_{b2}\, z_{a2} + y_{b1}\, z_{b2} - y_{b2}\, z_{b1}\right)}{q} \tag{3.44}$$

where:

$$q = p_y\, p_z \sqrt{{f_d}^2\, {p_y}^2\, {p_z}^2 + {p_y}^2\, {s_z}^2\, {z_{a2}}^2 + {p_z}^2\, {s_y}^2\, {y_{a2}}^2} \ldots$$
$$\sqrt{{f_d}^2\, {p_y}^2\, {p_z}^2 + {p_y}^2\, {s_z}^2\, {z_{b1}}^2 + {p_z}^2\, {s_y}^2\, {y_{b1}}^2}\, \sqrt{{f_d}^2\, {p_y}^2\, {p_z}^2 + {p_y}^2\, {s_z}^2\, {z_{b2}}^2 + {p_z}^2\, {s_y}^2\, {y_{b2}}^2} \quad (3.45)$$

As long as f_d,p_y and p_z are non-zero positive values, as they must in a real camera system, q has a non-zero positive value. Similarly the scale factor $\Delta {t}^3\, fd\, sy\, sz\, {py}^3\, {p_z}^3$ in the numerator is also positive as long as Δt, s_y and s_z are also positive and non-zero, which again, they must be in a real camera system. As such M is singular when:

$$(y_{a2}\, z_{b1} - y_{b1}\, z_{a2} - y_{a2}\, z_{b2} + y_{b2}\, z_{a2} + y_{b1}\, z_{b2} - y_{b2}\, z_{b1}) = 0 \quad (3.46)$$

Consider a line that passes through the two endpoints of the first streak with slope m_1 and z-intercept p_1

$$z = m_1 y + p_1 \quad (3.47)$$

So that

$$z_{a1} = m_1 y_{a1} + p_1 \quad (3.48)$$

$$z_{b1} = m_1 y_{b1} + p_1 \quad (3.49)$$

Now consider a line that passes through the two endpoints of the second streak with slope m_2 and z-intercept p_2

$$z = m_2 y + p_2 \quad (3.50)$$

So that

$$z_{a2} = m_2 y_{a1} + p_2 \quad (3.51)$$

$$z_{b2} = m_2 y_{b1} + p_2 \quad (3.52)$$

Substituting these relations into equation 3.46 and simplifying gives:

$$(y_{a2} - y_{b2})\, (p_1 - p_2 + m_1\, y_{b1} - m_2\, y_{b1}) = 0 \quad (3.53)$$

Note that this relation is true if $y_{a2} = y_{b2}$, for cases of finite m_1 and m_2 (infinite cases will be considered below, in which case, the second streak has zero length. It is also true if $p_1 = p_2$ and $m_1 = m_2$, the case where the streaks are collinear. In this case, no unique solution exists for equation 6.2.2. More broadly, no unique solution exists if:

$$p_1 - p_2 + m_1\, y_{b1} - m_2\, y_{b1} = 0 \tag{3.54}$$

Allow m_1, y_{b1}, p_1 to be arbitrary finite values (for an arbitrary first streak). Then the determinant is 0 if:

$$m_1\, y_{b1} + p_1 = m_2\, y_{b1} + p_2 \tag{3.55}$$

$$z_{b1} = m_2\, y_{b1} + p_2 \tag{3.56}$$

That is, if point b on the first streak lies on the line passing through the second streak. This set of cases obviously includes the collinear case. At this point, note that in the non-collinear variants of this case, the streak endpoints can be swapped (a for b) and M becomes invertible without any change to the actual streaks (because which end of a streak is the beginning and which is the end is unknown). However, in the collinear case, this switch results in point b from streak 1 still lying on the line of streak 2 and thus does not make M invertible. Therefore only the collinear case is truly without a unique solution.

Now consider the case of infinite slope streaks, that is streaks that are vertical in the image plane. If streak 1 is vertical

$$y_{b1} = y_{a1} \tag{3.57}$$

and z_{a1} and z_{b1} are arbitrary. Using this relation in equation 3.46 with the relations from equations 3.51 and 3.52 the determinant of M is zero if

$$(y_{a2} - y_{b2})\ (p_2 - z_{b1} + m_2\, y_{a1}) = 0 \tag{3.58}$$

Which again occurs if $y_{a2} = y_{b2}$, which is the case where streak 2 has zero length when m_2

is finite.

$$0 = p_2 - z_{b1} + m_2\, y_{a1} \tag{3.59}$$

$$z_{b1} = m_2\, y_{a1} + p_2 \tag{3.60}$$

$$z_{b1} = m_2\, y_{b1} + p_2 \tag{3.61}$$

Note that this is the same condition as in the finite m_1 case. So the above statement is unchanged. Now consider a case where m_2 is infinite and $m1$ is finite such that:

$$y_{b2} = y_{a2} \tag{3.62}$$

Then the determinant of M, determined using this relation and the relations from equations 3.48 and 3.49 is 0 when:

$$(y_{a2} - y_{b1})\ (z_{a2} - z_{b2}) = 0 \tag{3.63}$$

In this case again, if streak 2 has zero length ($z_{a2} = z_{b2}$) M is not invertible. If $y_{a2} = y_{b1}$ the determinant of M is again 0. In this case point b is again on the line that passes through streak 2, the same condition found in the other cases. Again if the lines are not collinear, the endpoints can be swapped and M becomes invertible.

The final case to consider is one where both streaks are vertical, $m_1 = m_2 = \infty$. Then:

$$y_{b1} = y_{a1} \tag{3.64}$$

$$y_{b2} = y_{a2} \tag{3.65}$$

In this case the determinant of M is zero when:

$$(y_{a1} - y_{a2})\ (z_{a2} - z_{b2}) = 0 \tag{3.66}$$

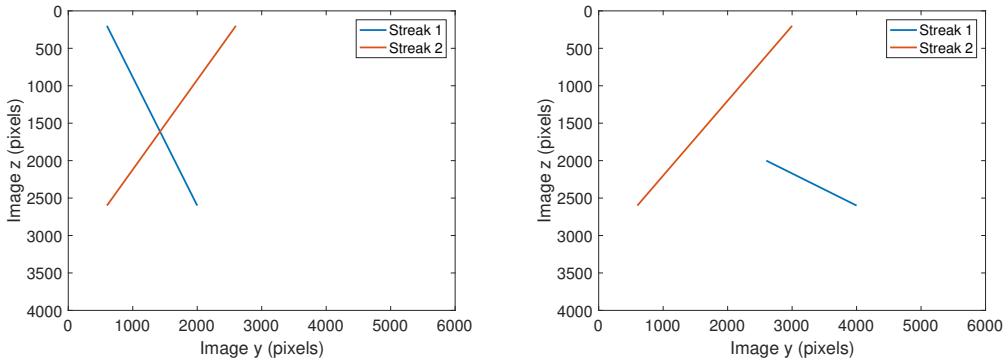
As such the determinant of M is zero only when streak 2 has zero length or $y_{a2} = y_{a1}$, which is the same as the streaks being collinear since, with both being vertical, they are parallel by definition.

Thus for all streak direction cases the determinant of M is zero for arbitrary direction and length streak 1, if streak 2 has zero length or streak 2 is collinear with streak 1. If the endpoint of streak one falls on the line that passes through the endpoints of streak 2 then the determinant of M is zero, but M can be made invertible by swapping the endpoints of streak 1 if streak 2 is not collinear with streak 1. Since the selection of which endpoint is a and which is b is arbitrary (streak direction is ambiguous), this can be done. Note that in for streaks that are near collinear and result in poorly conditioned M, the resulting poor solution is typically discarded by the RANSAC method used in the method implementation described in section 3.6.0.2.

Thus a matrix M determined from two streaks cannot be made invertible if either the two streaks are collinear or the second streak has zero length. Therefore, as long as the streaks are not collinear and the second streak has non-zero length, M has full rank and equation 6.2.2 has a unique solution. This allows solution of $\vec{V}$ using any pair of non-collinear streaks with finite length (note that zero-length streaks cannot be detected by the streak identification method described in section 3.5.3) scaled by the unknown range, r_{a1}. For clarity, the velocity expressed with this unknown unit will hereafter be referred to as $\vec{V}_s$. The solved velocity from this equation has units of r_{a1}/s. Since r_{a1} is unknown, the velocity magnitude cannot be determined from this in known units, but the direction can be. Resolving the flow speed uncertainty is handled in section 3.7.

With the non-invertible cases thus described, the next consideration is the physical feasibility of solutions. Scaled ranges are given relative to the camera optical center. Note that it is not possible for a particle to be detected if it has a negative range under this definition. However, because the streak directions are ambiguous a valid physical solution may give some negative scaled ranges. To see this recall equation 3.32:

$$\vec{0}^{3\times 1} = \hat{d}_a r_a + \vec{V}\Delta t - \hat{d}_b r_b \tag{3.67}$$



(a) Crossing Streaks.

(b) Streaks where the line through one intersects the other.

Figure 3.12: Examples of streak images with non-physical solutions.

Switch the signs on r_a and r_b

$$\vec{0}^{3\times1} = -\hat{d}_a r_a + \vec{V}\Delta t + \hat{d}_b r_b \tag{3.68}$$

$$\vec{0}^{3\times1} = \hat{d}_b r_b + \vec{V}\Delta t - \hat{d}_a r_a \tag{3.69}$$

Note that this is equivalent to swapping points a and b. Because the streak directions are ambiguous this swap can easily occur in the endpoint identification steps. Thus allowing negative ranges is necessary to account for the streak direction ambiguity.

While negative ranges are thus potentially physically valid there is one important case in which they are not. This is the case where the scaled ranges to the two streak endpoints have opposite signs. In this case regardless of the role of streak direction ambiguity one of the points would be associated with a particle position behind the optical center, but such a position cannot be imaged. This leads to a class of cases where M is full rank and a solution is possible but the solution is not physically possible. Generally this is associated with streaks that cross or where the line passing through the endpoints of one streak intersects another streak. Example cases are shown in figure 3.12

By solving the system in equation 6.2.2 symbolically in MATLAB, the scaled ranges

are found as follows:

$$r_{b1} = -\frac{h_{b1}\,(y_{a1}\,z_{a2} - y_{a2}\,z_{a1} - y_{a1}\,z_{b2} + y_{b2}\,z_{a1} + y_{a2}\,z_{b2} - y_{b2}\,z_{a2})}{k\,(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.70}$$

$$r_{a2} = \frac{h_{a2}\,(y_{a1}\,z_{b1} - y_{b1}\,z_{a1} - y_{a1}\,z_{b2} + y_{b2}\,z_{a1} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})}{k\,(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.71}$$

$$r_{b2} = -\frac{h_{b2}\,(y_{a1}\,z_{a2} - y_{a2}\,z_{a1} - y_{a1}\,z_{b1} + y_{b1}\,z_{a1} + y_{a2}\,z_{b1} - y_{b1}\,z_{a2})}{k\,(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.72}$$

$$h_{b1} = \sqrt{{f_d}^2\,{p_y}^2\,{p_z}^2 + {p_y}^2\,{s_z}^2\,{z_{b1}}^2 + {p_z}^2\,{s_y}^2\,{y_{b1}}^2} \tag{3.73}$$

$$h_{a2} = \sqrt{{f_d}^2\,{p_y}^2\,{p_z}^2 + {p_y}^2\,{s_z}^2\,{z_{a2}}^2 + {p_z}^2\,{s_y}^2\,{y_{a2}}^2} \tag{3.74}$$

$$h_{b2} = \sqrt{{f_d}^2\,{p_y}^2\,{p_z}^2 + {p_y}^2\,{s_z}^2\,{z_{b2}}^2 + {p_z}^2\,{s_y}^2\,{y_{b2}}^2} \tag{3.75}$$

$$k = \sqrt{fd^2\,py^2\,pz^2 + py^2\,sz^2\,{z_{a1}}^2 + pz^2\,sy^2\,{y_{a1}}^2} \tag{3.76}$$

The h_i and k factors are positive for real values and $r_{a1} = 1$, which is positive, so a non-physical case occurs if

$$0 \geq -\frac{(y_{a1}\,z_{a2} - y_{a2}\,z_{a1} - y_{a1}\,z_{b2} + y_{b2}\,z_{a1} + y_{a2}\,z_{b2} - y_{b2}\,z_{a2})}{(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.77}$$

$$\tag{3.78}$$

or if

$$\frac{(y_{a1}\,z_{b1} - y_{b1}\,z_{a1} - y_{a1}\,z_{b2} + y_{b2}\,z_{a1} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})}{(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.79}$$

and

$$-\frac{(y_{a1}\,z_{a2} - y_{a2}\,z_{a1} - y_{a1}\,z_{b1} + y_{b1}\,z_{a1} + y_{a2}\,z_{b1} - y_{b1}\,z_{a2})}{(y_{a2}\,z_{b1} - y_{b1}\,z_{a2} - y_{a2}\,z_{b2} + y_{b2}\,z_{a2} + y_{b1}\,z_{b2} - y_{b2}\,z_{b1})} \tag{3.80}$$

have opposite signs. In these cases a solution is possible but the solution represents a physically impossible situation.

3.6.0.2 Solution Implementation

The matrix M is extensible to any number of streaks. For n streaks, this extended system is:

$$\begin{bmatrix} -\hat{d}_{a1} \\ \vec{0}^{3(n-1)\times 1} \end{bmatrix} = \begin{bmatrix} \Delta t & 0 & 0 & -\hat{d}_{b1}(1) & 0 & 0 & \dots & 0 & 0 \\ 0 & \Delta t & 0 & -\hat{d}_{b1}(2) & 0 & 0 & & 0 & 0 \\ 0 & 0 & \Delta t & -\hat{d}_{b1}(3) & 0 & 0 & & 0 & 0 \\ \Delta t & 0 & 0 & 0 & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) & & 0 & 0 \\ 0 & \Delta t & 0 & 0 & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) & & 0 & 0 \\ 0 & 0 & \Delta t & 0 & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) & & 0 & 0 \\ \vdots & & & & & & \ddots & & \\ \Delta t & 0 & 0 & 0 & 0 & 0 & & \hat{d}_{a2}(1) & -\hat{d}_{b2}(1) \\ 0 & \Delta t & 0 & 0 & 0 & 0 & & \hat{d}_{a2}(2) & -\hat{d}_{b2}(2) \\ 0 & 0 & \Delta t & 0 & 0 & 0 & & \hat{d}_{a2}(3) & -\hat{d}_{b2}(3) \end{bmatrix} \begin{bmatrix} \vec{V} \\ r_{b1} \\ r_{a2} \\ r_{b2} \\ \vdots \\ r_{an} \\ r_{bn} \end{bmatrix} \tag{3.81}$$

$$\vec{b}_e = M_e x_e \tag{3.82}$$

Adding a new streak adds two unknown range terms and three new rows. Thus the size of M_e is $3n \times 2n + 2$.Note that in this system the addition of more data, in the form of more streaks, also adds to the solution complexity by adding unknowns.

The number of rows of M_e is $3n$ but the rank is necessarily $\leq 2n + 2$, the number of columns of M_e. For $n > 2$, the number of rows is thus greater than the rank. As such, when $n > 2$ the left null space is non-trivial. Therefore it is possible that, for some systems, no solutions exist. However, as long as M has full column rank, the Gramian matrix, $M_e^T M_e$ has full rank and a linear least squares estimate is possible. Preliminary study indicates this is the case as long as any two of the streaks used to build M_e are not collinear and do not have zero length.

To see this, note that when the two streaks making up M are collinear the rank of M,

determined using the MATLAB symbolic toolbox, is 5. When extending M_e to three streaks adding a streak collinear to a previous streak increases the rank of M_e by 2 whereas adding a streak that is not collinear to a previously used streak, increases the rank of M_e by 3. This was again the case when adding a fourth streak. Further study is needed to determine if this is the case generally.

If this is the case in general then the following conclusion could be reached. Because each streak added to M_e adds two unknowns, if the initial streak pair is not collinear (so the rank of M_e with the initial pair is 6) then M_e will never become rank deficient no matter how many additional collinear streaks are added. If the initial streak pair is collinear (so the rank of M_e with the initial pair is 5) the rank of M_e will be one less than the number of columns of M_e if only collinear streaks are added, as each streak adds two unknowns and increases the rank by two. Thus if a single non-collinear streak is added to a rank deficient M_e it will become full column rank as the rank will increase by three making the rank equal to the number of columns.

While a least squares solution is possible, to reduce the impact of any image artifacts of linear background features a random sample consensus (RANSAC) approach is used [39]. In this approach, a random pair of streaks is created from the streak list and the scaled velocity is solved by performing $\vec{x} = M^{-1}\vec{b}$. This result is then tested for validity with the remaining streaks. This is done by testing to see if the line created in the image plane that runs through one end of the streak and the point reached by $\hat{d}_1 + \vec{V}_s dt$ projected into the

image plane is parallel to the streak itself. This allows the definition of a cost for each streak:

$$\hat{S}_p = \frac{\text{Im}\left(\hat{d}_a + \vec{V}_s \Delta t\right) - \text{Im}\left(\hat{d}_a\right)}{\|\text{Im}\left(\hat{d}_a + \vec{V}_s \Delta t\right) - \text{Im}\left(\hat{d}_a\right)\|} \tag{3.83}$$

$$\hat{S}_m = \left(\begin{bmatrix} y_{p1} \\ z_{p1} \end{bmatrix} - \begin{bmatrix} y_{p2} \\ z_{p2} \end{bmatrix} \right) \frac{1}{\sqrt{(y_{pb} - y_{pa})^2 + (x_{pb} - x_{pa})^2}} \tag{3.84}$$

$$\vec{c} = \hat{S}_p \times \hat{S}_m \tag{3.85}$$

$$J = |\vec{c}(1)| \tag{3.86}$$

where Im is the image projection function defined in equation 3.27. An illustration of this cost function is shown in figure 3.13. The cost function tests how close to parallel the direction vectors $\hat{S}_m$ and S_p are in the image plane.

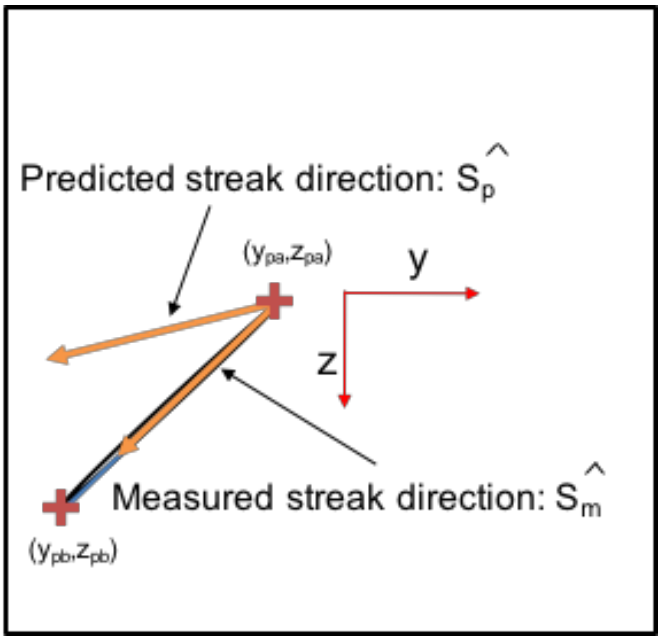


Figure 3.13: Illustration of the RANSAC cost function. The cross product of $\hat{S}_m$ and S_p is taken to determine how close to parallel the two are.

A tunable cost threshold is set and the number of streaks whose costs are below this threshold is counted up. If the total number of such streaks exceeds a tunable threshold

amount, a least-squares solution is run using all the streaks that fell below the cost threshold (as noted, the matrix M can be arbitrarily extended to any number of streaks) and the solution is stored as a candidate. This candidate's cost is then assessed by applying equation 3.83 to every streak.

The cost and solution are stored and the above method rerun until a suitable number (another tunable threshold) of solutions have been found. The one with the lowest cost is retained as the solution.

One final issue must be confronted at this point. The issue with using a streak method is that the sign of the flow direction is arbitrary. Two equally valid solutions are possible, one positive and one negative. Because this method is aimed at fixed wing sUAS applications, a basic assumption that the vehicle travels forward can be used to assume that the flow vector points into the rear hemisphere of the sensor axis system. This is sufficient information to give the appropriate sign to the solution scaled velocity vector. For applications where flow direction could come from any angle, potential methods of resolving this ambiguity are discussed in section 6.2

This solution direction vector must then be scaled to get the relative air velocity into the selected units. That is, the air speed must be solved for.

3.6.1 Image Distortion Effects

The preceding derivation relied on a distortion-free pinhole camera model. In reality, optical systems distort images. The general distortion model of Brown provides a model for this distortion that is widely used [6]. The Brown model provides a functional relationship between the position of a pixel in a distorted image and its position in an undistorted image of the same scene. Under this model y_d and z_d, the distorted positions, have the following

relationship to the undistorted image positions:

$$r = \sqrt{(y_u - y_c)^2 + (z_u - z_c)^2} \tag{3.87}$$

$$y_d = y_u \left(1 + K_1 r^2 + K_2 r^4 + \ldots\right) + \left(P_2 \left(r^2 + 2y_u^2\right) + 2P_1 y_u z_u\right) \left(1 + P3r^2 + P_4 r^4 + \ldots\right) \tag{3.88}$$

$$z_d = z_u \left(1 + K_1 r^2 + K_2 r^4 + \ldots\right) + \left(P_2 \left(r^2 + 2z_u^2\right) + 2P_1 y_u z_u\right) \left(1 + P3r^2 + P_4 r^4 + \ldots\right) \tag{3.89}$$

Here (y_c, z_c) is the distortion center, generally taken as the principal point of the image. The $K_1, K_2, \ldots$ terms are the radial distortion terms while the $P_1, P_2, \ldots$ terms are the decentering distortion terms.

The effects of distortion are to invalidate the relationships in equations 3.27 and 3.29. This can create substantial error if the distortion is large. The effects of a simple first-order radial distortion are shown in section 4.9.

Fortunately many commercial tools exist to both determine and correct camera distortion [1]. As a result distortion is not a major obstacle to the proposed method, however, a low-distortion optical system may not require correction and as such could provide a computation time benefit.

3.7 Flow Speed Determination

With the flow direction determined, the final step is to determine the flow speed. This is accomplished by requiring that streak start and end points fall within the imaged volume. This is a single degree of freedom optimization problem.

In practice however, some particles will leave the imaged volume during the exposure. Those that leave the lateral edges (y and z directions) of the volume are easily identified by having end location located on or very near the image edges. Streaks with one endpoint at the image edge can thus be eliminated from the optimization. However those that exit the front or back of the volume are harder to identify. As such the associated streaks cannot be

eliminated, thus ensuring that getting all endpoints to fall within the volume is not possible.

Let $\vec{r}$ be a vector of all the scaled range elements of $\vec{x}$ that correspond to streaks that do not have endpoints on the image edges. All these ranges are scaled by the length of r_{a1}. The objective then is to pick a value of r_{a1} that places all elements of $r_{1a}\vec{r}$ within the depth limits of the imaged volume.

Thus, if n is the total number of elements in $\vec{r}$, a basic cost function can be defined as follows

$$\vec{J}_{r_i}(r_{a1}) = \begin{cases} 1, & \text{if } r_{a1}\vec{r}_i \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{3.90}$$

$$J_r(r_{a1}) = \sum_{i=1}^{n} \vec{J}_{r_i}(r_{a1}) \tag{3.91}$$

Then

$$\underset{r_{1a}}{\operatorname{argmax}}\, \vec{J}_{r_i}(r_{a1}) := \{r_{a1} \in \mathbb{R}_{\geq 0}\} \tag{3.92}$$

yields the optimal range scale.

While 3.92 describes the optimization problem generally, implementing a practical solution is somewhat more complex. Simply minimizing the relation created issues in practice when the first streak end, r_{a1} happened to be on a streak that left either the front or back of the volume, had its start and end points improperly identified, or was eliminated because it crossed an image edge.

Instead an iterative minimization approach was used. First $\vec{r}$ was sorted from shortest to longest element, yielding the column vector $\vec{r}_s$. Because streaks entering or leaving the volume have their length shortened, their scaled distances are longer (due to the nature of the motion, shorter streaks are solved as being farther away). Thus, if any streaks fall completely within the volume, it is likely the streak with the closest apparent scaled range does. Points are then processed by moving down $\vec{r}_s$ from the shortest to longest element.

For each element of $\vec{r}_s$ to be processed, the next step is to create a range scale vector $\vec{s}$. This row vector contains a range of scale factors from smallest to largest, where the smallest value multiplied by the current element of $\vec{r}_s$ places the product position on the near side (smallest x component) of the volume and the largest value. Increasing the resolution of this vector (number of elements in $\vec{s}$) increases the precision of the resulting air speed solution at the expense of computation time.

To keep this relation easy to handle in practice, $\vec{s}$ is created by first creating a vector of ranges between the minimum and maximum ranges in the volume with a resolution in distance equivalent to the desired resolution in air speed (for 0.01m/s resolution in air speed a 0.01m range resolution in this vector would be used). This vector is then divided by the current element of $\vec{r}_s$ to give the appropriate scaling vector with desired precision.

The product of the sorted range and the scaling vector is a matrix R where each column is a list of ranges to the streak ends for each scaling.

$$R_{ij} = \vec{r}_{s_i}\vec{s}_j \tag{3.93}$$

With R determined, a binary matrix can be created that reflects if any point in R is inside the volume. Allow r_{max} to be the far extent of the volume andr_{min} to be the near extent of the volume.

$$B_{r_{ij}} = \begin{cases} 1, & \text{if } R_{ij} \geq r_{max} \\ 1, & \text{if } R_{ij} \leq r_{min} \\ 0, & \text{otherwise} \end{cases} \tag{3.94}$$

Because the streaks with shorter scaled ranges are more likely not to be truncated, it is more important that their end points lie inside the volume when selecting an appropriate scaling. A simple weighting vector $\vec{w}$ is created that weights solutions that keep the nearer (more likely not to be truncated) streaks in the volume:

$$\vec{w}_i = r_i^{-1} \tag{3.95}$$

This weight vector allows a weighted sum of the number of streak ends falling outside the volume for each range scaling option in $\vec{s}$ to be determined:

$$\vec{j} = \vec{w}B_r \tag{3.96}$$

The minimum value in $\vec{j}$ gives the cost for the current point being processed and the optimal scale factor based on the processed point. If there are multiple minima in $\vec{j}$, the scale factor chosen is the one closest to the mean value of all the scale factors associated with minima in $\vec{j}$.

The chosen scale factor for the point being processed then has a new cost determined as the total fraction of end points that fall within the volume using the scale factor determined $J_s = \frac{l_i}{n}$ where l_i is the number of line end points in the volume using the determined scale factor. Each processed point thus receives a cost and scale factor, points are marched through from shortest to longest scaled range until a the line fraction cost is below a user-tunable threshold. The first point have J_s above this threshold has its associated scale factor s_f taken as the solution.

With the scale factor determined the final estimated velocity is:

$$\vec{V_e} = s_f \vec{V_s} \tag{3.97}$$

Chapter 4

Simulation Results

To test the method with a variety of imaging conditions and parameter alterations, a simulation environment was created in MATLAB. This environment creates simulated images using the pinhole camera model described in chapter 3. These simulated images allow for application of varying lens focal lengths, subject distances, f-numbers, sensor sizes and exposure times, as well as adding distortion effects, random noise, background imagery and particle velocity fluctuations (providing a crude model of turbulence).

4.1 Simulation Environment

The simulation model assumes an environment of particles that have a single pixel diameter when imaged. Thus when in sharp focus, the particles produce an image on the sensor whose size is determined by their Airy disk radius [3]:

$$r_{airy} = 1.22\lambda N \tag{4.1}$$

for f-number N.

For the purposes of the simulation, the light intensity was assumed to peak at roughly $\lambda = 500$nm. Particles not in sharp focus were simulated as providing a uniform light intensity distributed across their blur disk. The radius of the blur disk from [28] is given as follows:

$$r_{blur} = \frac{fm_s}{N}\frac{|r-s|}{r} \tag{4.2}$$

again for a subject focus distance s and blurred object at range r.

To create images, a random cloud of particles, with a uniform spatial distribution and user-tunable density, is created inside a rectangular prism. This prism's dimensions are set as the largest limits of the imaged volume (the depth limits described in chapter 3 along with the largest y and z values covered by the pinhole model of the camera with the given parameters). In the focus-limited case, the simulated depth limits were extended by a factor of two in the simulation to allow particles to naturally come into and out of focus, while in the illumination-limited case, particles were shown only when actually present inside the imaged volume. These limits are then extended to cover the volume that will move through the imaged volume during the exposure.

A bulk velocity is specified and individual particles have their velocities perturbed with a simple Gaussian additive noise with a variance from section 2.2. While not a true turbulence model, this does allow for a limited ability to explore the effects of velocity fluctuations parameterized by the dissipation rate.

Particle motion is then simulated with a forward Euler integrator with fixed time step (as the particles all move in straight lines in this model). The integrator is run for the duration of the exposure time. The fixed time step allows for modeling the particle light returns by simply adding the returns at each step, this accurately portrays how particles with lower lateral motion on the sensor generate brighter streaks.

To model particle light return strength, the model assumes a fixed strength C at a 1m distance from the lens. The light source is assumed to be at a distance from the particles equal to the distance from the particles to the camera. The strengths of individual particles then follow an inverse quartic profile (in agreement with the radar range equation [17]) scaled by this fixed strength. Because the real particulate environment is heterogeneous, attempting to model particle light return strengths in greater detail would be complex, and necessarily inaccurate, and so was not attempted.

Particle light return strength S_p for a particle at range r is then:

$$S_p = Cr^{-4} \tag{4.3}$$

The return strength for each particle is then spread evenly over the larger of the blur disk radius r_{blur} or airy disk radius r_{airy}. While this is not an accurate depiction of the the airy pattern, the airy disk sizes for the camera systems modeled are very small and in practice the airy limit is only achieved for relatively small segments in only some streaks, thus it has minimal effect on simulation accuracy.

An image matrix is created that sums all these strength returns from each time step at each pixel, applies a simple saturation model, and then normalizes the result to create a gray-scale image of particle streaks. Such an image for a focus-limited case is shown in figure 4.1. This serves as a base to apply various error sources.

Figure 4.1: Particle streaks simulated accounting for light fall-off and depth of field.

First a background image is added. To provide this image, the simulation relies on

photographs. The best practice is to take a photograph with settings as close as possible to those being used in the simulation. This photo provides the background signal matrix B. This is scaled by t_b, a scale factor defined such that when $t_b = 1$ the strongest return in the background is equal to the maximum value in the streak image I_s. The scaled B matrix is then added to I_n to produce the simulated image with background I_b:

$$I_b = t_b B + I_s \tag{4.4}$$

Figure 4.2 shows the simulated image from figure 4.3 with background imagery added.

Figure 4.2: Streak image from 4.3 with background added.

Next, a simple additive Gaussian noise is applied to simulate camera sensor noise. This is parametrized with a variance scaled to the strongest streak return. Let Z be a matrix of the same size as I_b with components distributed according to a zero-mean Gaussian. For a

streak image with background I_b, the resulting noisy image I_n is then:

$$\sigma = t_n \max(I_s) \tag{4.5}$$

$$Z = \mathcal{N}(0, \sigma^2) \tag{4.6}$$

$$I_n = I_b + Z \tag{4.7}$$

where $\mathcal{N}(0, \sigma^2)$ is the normal distribution with zero mean and standard deviation σ, and t_n is a scale factor for the noise, that is set as a simulation parameter. Figure 4.3 shows a simulated image with camera noise added.

Figure 4.3: Streak image from 4.1 with noise added.

Finally I_n is distorted using only the first order radial term from the Brown model described in section 3.6.1. This is a crude representation but allows some exploration of the effects of image mapping errors in a quantifiable manner. The distorted image I is then processed using the method described in chapter 3. A final image with severe distortion (for illustration) is shown in figure 4.4 compared to the undistorted image from figure 4.3.

Figure 4.4: Streak image from 4.3 shown on the left with distortion added on the right.

4.2 Simulation Parameter Settings

Tables 4.1, 4.2, and 4.3 list the parameters used in the simulation by default, with adjustments made as described in the following sections. Default imaging system parameters were chosen to match the system available for wind tunnel testing (described in the next chapter)

4.3 Error Quantification

Because the direction and airspeed are solved separately, errors are quantified separately for direction and air speed. Direction errors are described in terms of simple vertical and horizontal angles:

$$\Delta\theta_{vert} = \arctan\left(\frac{\vec{V}_e(3)}{\vec{V}_e(1)}\right) - \arctan\left(\frac{\vec{V}_{true}(3)}{\vec{V}_{true}(1)}\right) \tag{4.8}$$

$$\Delta\theta_{horz} = \arctan\left(\frac{\vec{V}_e(2)}{\vec{V}_e(1)}\right) - \arctan\left(\frac{\vec{V}_{true}(2)}{\vec{V}_{true}(1)}\right) \tag{4.9}$$

Airspeed error is simply:

$$\Delta AS = |\vec{V}_e| - |\vec{V}_{true}| \tag{4.10}$$

For the varying error parameters, a Monte Carlo approach was used to determine relations between the mean and variance of each error quantity and the values of the error

Table 4.1: Simulation Environment Imaging Parameters

Parameter	Value
focal length	50mm
f-number	8
aperture diameter	6.25mm
sensor size (mm)	25.3×15.6mm
sensor size (px)	6000×4000
exposure time	0.05s
illumination wavelength	500nm
particle density	100particles/m^3
subject distance	1m

source parameters listed in 4.2. A span of 100 values of each parameter was tested with 100 simulated images for each value. Plots were then made comparing the error quantities to the error source parameters. This allowed for determination of relationships between error sources and method accuracy. For the default particle density used here, the number of detected streaks was typically on the order of 10. The expected accuracy under ideal conditions (no error sources) for this number of streaks can be seen in the following section. Sections 4.6, 4.7, 4.8, and 4.9 show the impacts of velocity fluctuations, bakcground signal, image noise and mapping errors (as described in chapter 3) on the error quantities.

4.4 Accuracy Convergence

To investigate the impact of the number of streaks in an image on solution accuracy, a single image with a high streak density was created using the simulation environment. 242 streaks were identified in the image. Starting with a set size of two streaks and increasing the set size to 242, 242 random sets were selected and the velocity solution performed on each (Allowing unique sets in the 241 streak case). Note that in the 242 streak case this amounts to using the same set every time but due to the random start set of the RANSAC approach this still results in different solutions.

Figure 4.5 shows the impact on angle accuracy of increasing the streak size. Errors

Table 4.2: Simulation Environment Error Source Parameters

Parameter	Value
t_b	0
t_n	0
radial distortion	0
turbulence dissipation rate	$10^{-2}\frac{\mathrm{m}^2}{\mathrm{s}^3}$

were near zero mean so were parameterized only by RMSE. Note that sub degree accuracy which is good relative to existing systems, is possible with small numbers (less than 10) streaks, but larger sets (over 50) can deliver RMSE values below 0.1°.

Figure 4.6 shows the impact on air speed accuracy of increasing the streak set size. Note that while increasing the set size results in a decreasing random air speed error it does not have nearly any significant impact on the air speed bias. As a result, RMSE does not drop significantly with larger sample sizes beyond numbers in the low tens of streaks used for the solution. This indicates that further improvements are needed in the air speed estimation approach to reduce this bias.

4.5 Depth-Limiting Approaches

Using the simulation method described in section 4.1, the different depth-limiting approaches derived in chapter 3 can be compared. For each possible approach 1000 simulated images were tested. In the illumination limited case an f-number of 25 was used.

Figure 4.7 shows the distribution of air speed errors using a focus-limited approach with the limits calculated using equation 3.2. Note that the errors are large and significantly biased to overestimates with significant outliers beyond 15m/s (the spike in the largest bin represents the total of values falling beyond the plotted range). This relatively poor performance results from a combination of factors in the focus-limited case. Firstly, extremely blurred streaks appear longer than they actually are (because focus blur extends each end), this leads to solutions for the air speed that are too high (longer streaks are associated with higher air

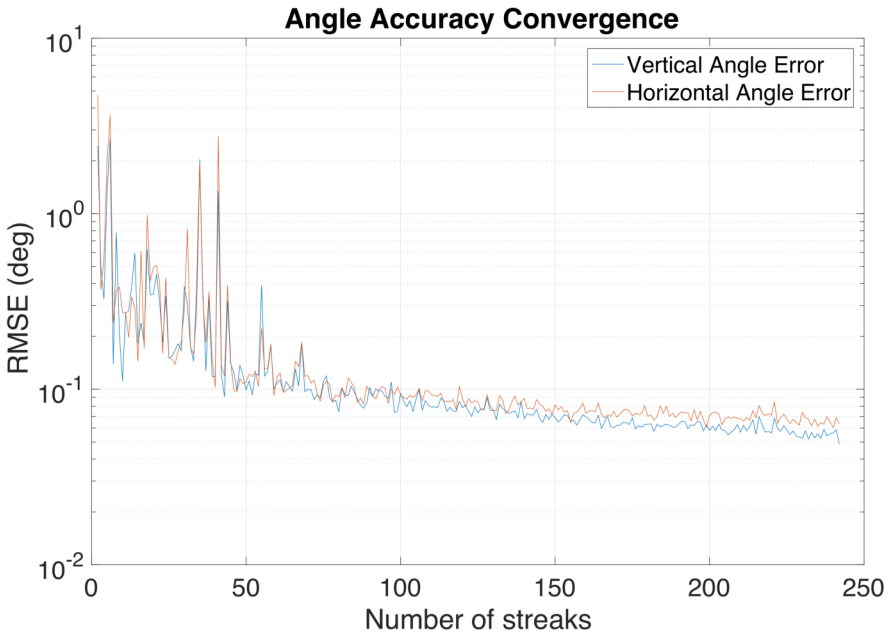


Figure 4.5: RMSE of flow angle errors against the number of detected streaks available for the solution.

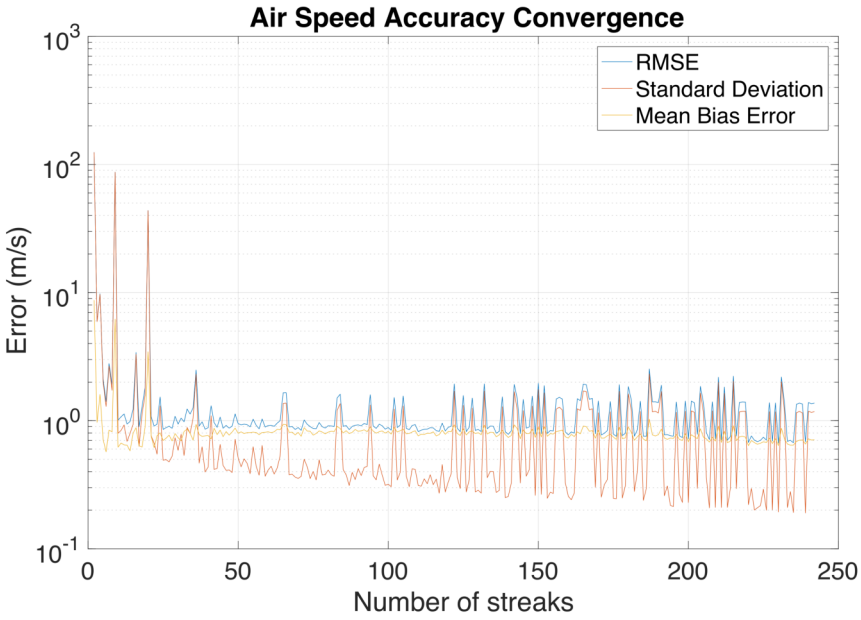


Figure 4.6: RMSE, standard deviation, and mean bias error of air speed errors against the number of detected streaks available for the solution.

Table 4.3: Simulation Environment Solver Parameters

Parameter	Value
edge detection Gaussian kernal size	2 pixels
edge detection Gaussian filter σ	1.4
edge detection weak gradient threshold	$\frac{\max(G)}{30}$
edge detection strong gradient threshold	$\frac{\max(G)}{15}$
RANSAC direction solver minimum sets to test	100
RANSAC direction solver minimum solution sets to reach	20
RANSAC direction solver minimum number of streaks for solution set	5
RANSAC direction solver maximum cost for set inclusion	0.25
air speed solver minimum distance from streak end to image edge	50 pixels
air speed solver minimum fraction of line ends in image depth	0.7

speeds). The second issue is that due to light falloff, heavily blurred streaks near the camera (those near D_n) still have strong enough gradients to appear in the edge detection, but equivalently blurred streaks far from the camera (those near D_f) do not. This effectively shifts the true imaged volume closer to the camera, making the calculated limits inaccurate. This shift will generally bias the results to be too high.

Figure 4.8 shows the distribution of vertical angle errors using the same calculated depths. The method delivers accuracy well under 1∘ in this case. A handful of outlying cases are noted which result from cases where all the streaks present in the image are closely aligned. This makes solutions very difficult as multiple aligned streaks do not provide more information than a single streak. The frequency of such cases is inherently dependent on the particle density as larger numbers of identified streaks are less likely to align.

Figure 4.9 shows the distribution of vertical flow angle errors. The distribution is very similar to the horizontal angles. Again there are some outliers resulting from streak alignment but these are minimal and the standard deviation remains below 0.2∘.

Figure 4.10 shows the distribution of air speed errors with measured image depth. Results are similar to the calculated-limits case, and in fact slightly poorer. Because the depth limits were measured in simulation as the ranges where streaks blurred beyond detectable

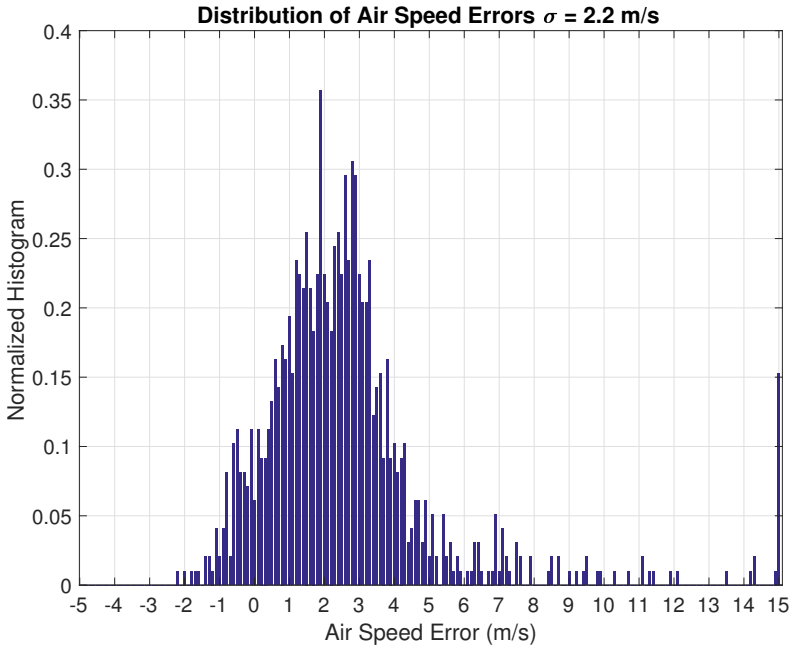


Figure 4.7: Normalized histogram of air speed errors using focus-limited depth with calculated limits. Note that outliers in the edge bins have been excluded from the standard deviation.

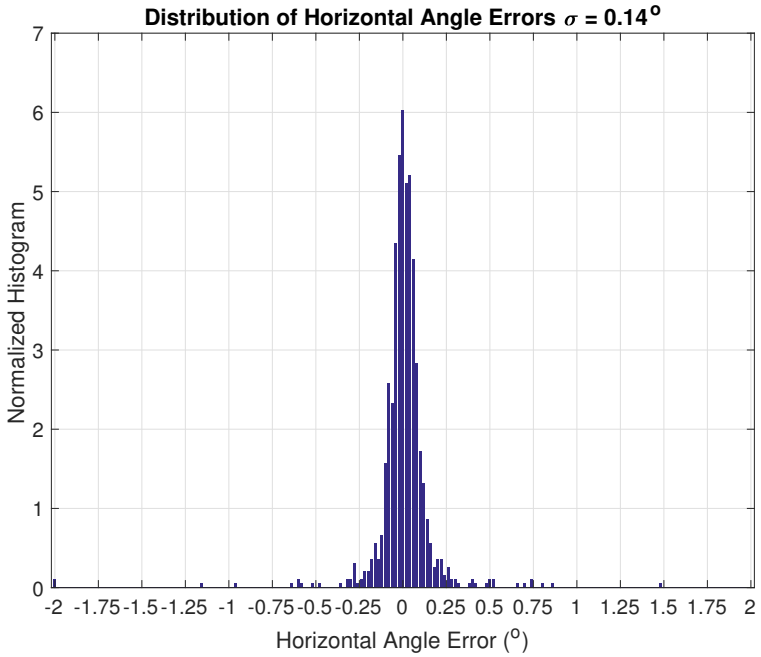


Figure 4.8: Normalized histogram of vertical angle errors using focus-limited depth with calculated limits.

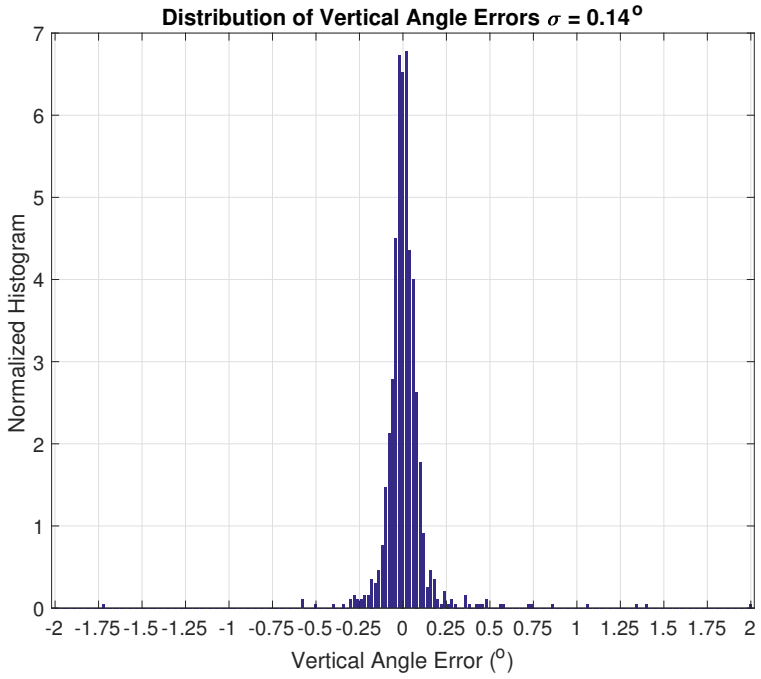


Figure 4.9: Normalized histogram of vertical angle errors using focus-limited depth with calculated limits.

thresholds, it is likely the same issues of light fall-off are in play here. Additionally the measurements relied on a test image and it is possible that results from this image cannot be translated broadly. Overall these results suggest little if any benefit to attempting direct measurement of image depth limits.

Angular error distributions shown in figures 4.11 and 4.12 again yield excellent accuracy. The small differences are likely driven by differing numbers of samples with larger errors (due to the narrow distribution, limited numbers of these cases have measurable effects on the standard deviation).

The air speed error distribution for the illumination-limited case is shown in figure 4.13. Performance is markedly improved over the focus-limited cases, with fewer outliers and a narrower distribution. A bias to positive values is again noted. This is likely a consequence of prioritizing the smaller scaled ranges in the solution resulting in a tendency to ensure their points are always solved with enough range to fall inside the illumination range. However, for streaks that have short scaled ranges and short lengths; those that point almost directly at the camera at close range; the streak lengths may be slightly extended by the edge detection, resulting in an overestimate of air speed.

Again, direction accuracy is very good, though slightly poorer than the focus limited cases. This is likely from a combination of two factors. Firstly, longer streaks give better direction accuracy as the PCA can more accurately align to the true streak direction. In focus limited cases some streaks are bright enough to be detected beyond the anticipated volume (noted earlier as a source of air speed error). This allows for somewhat longer streak lengths that may contribute a slight improvement in accuracy.

A second potential cause is that some defocused streaks are detected by the edge detection algorithm as two streaks which point towards an intersection where the streak would be focused in the image. This can limit the angle errors when all streaks in the image are aligned, reducing the amount of error in such images. So even though such images would not provide accurate direction in either case, in the focus-limited case, this false intersection

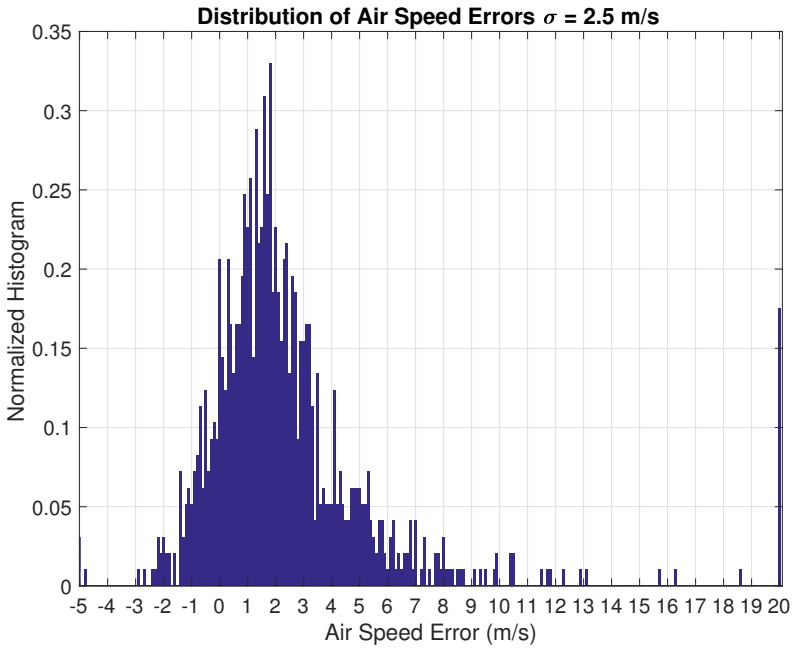


Figure 4.10: Normalized histogram of air speed errors using focus-limited depth with measured limits.

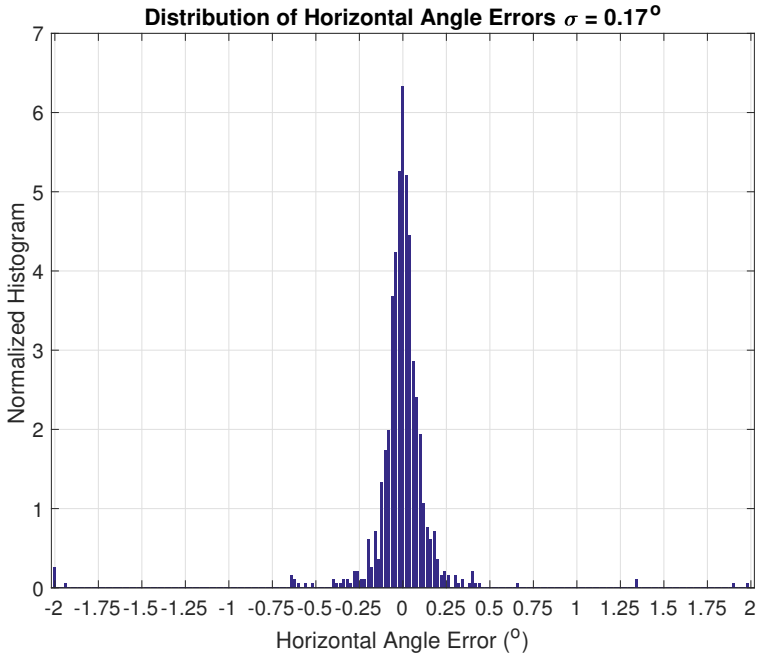


Figure 4.11: Normalized histogram of vertical angle errors using focus-limited depth with measured limits.

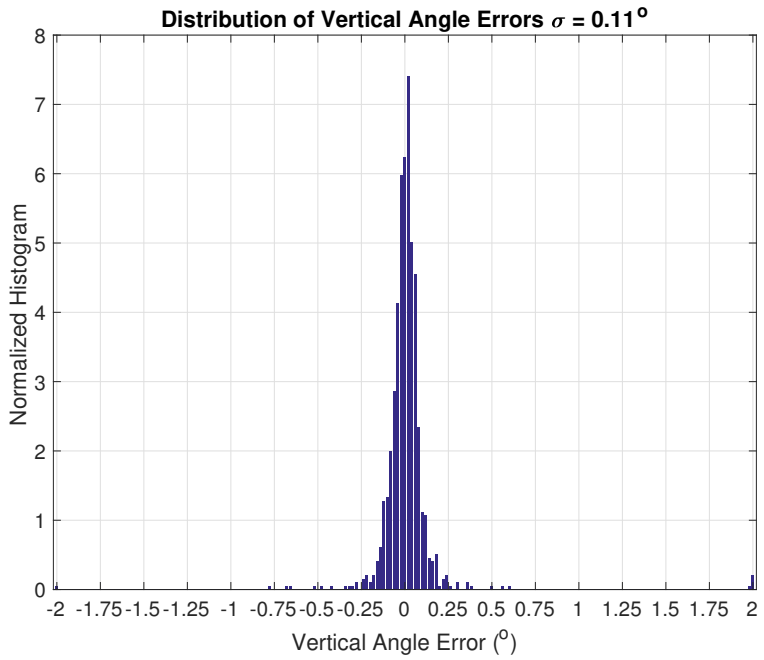


Figure 4.12: Normalized histogram of vertical angle errors using focus-limited depth with measured limits.

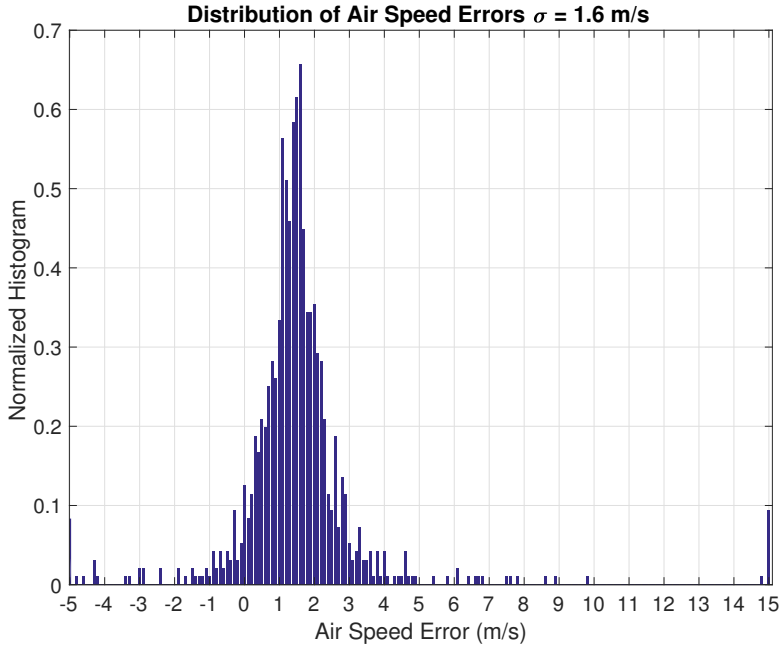


Figure 4.13: Normalized histogram of air speed errors using illumination-limited depth limits.

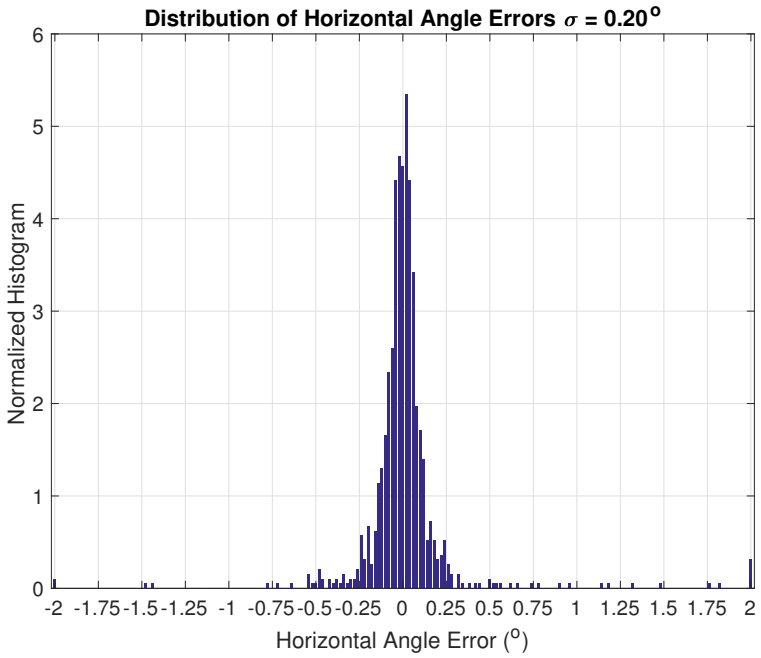


Figure 4.14: Normalized histogram of vertical angle errors using illumination-limited depth limits.

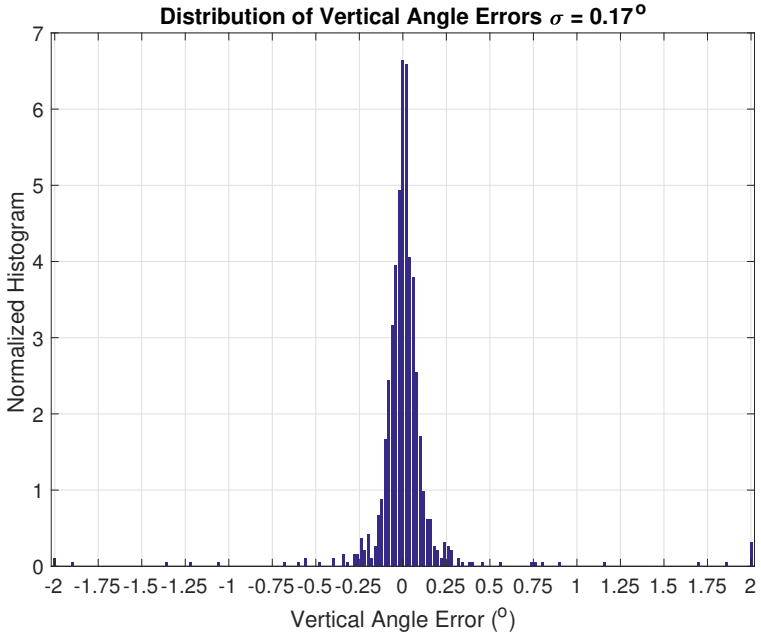


Figure 4.15: Normalized histogram of vertical angle errors using illumination-limited depth limits.

may lead to the estimation of smaller (but still incorrect) direction angles in some of these images, and thus smaller overall variance.

Overall, the direction accuracy predicted here is similar for all cases and compares very favorably to existing flow sensing systems commonly used on sUAS systems. However, air speed errors remain a major issue in all cases. While the primary aim of this work was to improve flow direction sensing, improved air speed sensing should be a focus of future work. Some potential approaches to improve the method performance in this regard are described in chapter 6.

For the remainder of the simulations presented, an illumination-limited approach is used so that the impacts of the error sources on air speed are not dominated by errors resulting from the depth-limiting approach.

4.6 Turbulence

The impacts of increasing the simulated dissipation rate are shown in figures 4.16, and 4.17. Directional errors remain generally on the order of 0.1° up to $\epsilon \approx 3 \times 10^{-2}\text{m}^2/\text{s}^3$. This is near the top of measured values of the dissipation rate in the lower troposphere [10] Beyond this point variance increases though mean error remains in the sub 0.1° range up to $\epsilon \approx 1\text{m}^2/\text{s}^3$ which is roughly two-orders of magnitude above the typical top of the physical range for turbulence in the lower troposphere. For these simulations, 1000 images were utilized rather than 100 to clarify the trends.

Results show that in lower turbulence environments the angular accuracy delivered by the method can be exceptional, slightly above 0.1°. In higher turbulence environments, there is minimal biasing allowing the potential to average multiple measurements to keep errors low.

Air speed errors are larger but show a similar trend in error behavior with increasing ϵ. Air speed accuracy may be acceptable for some applications at low levels of turbulence but at higher levels the errors climb into the same range as the measurement itself.

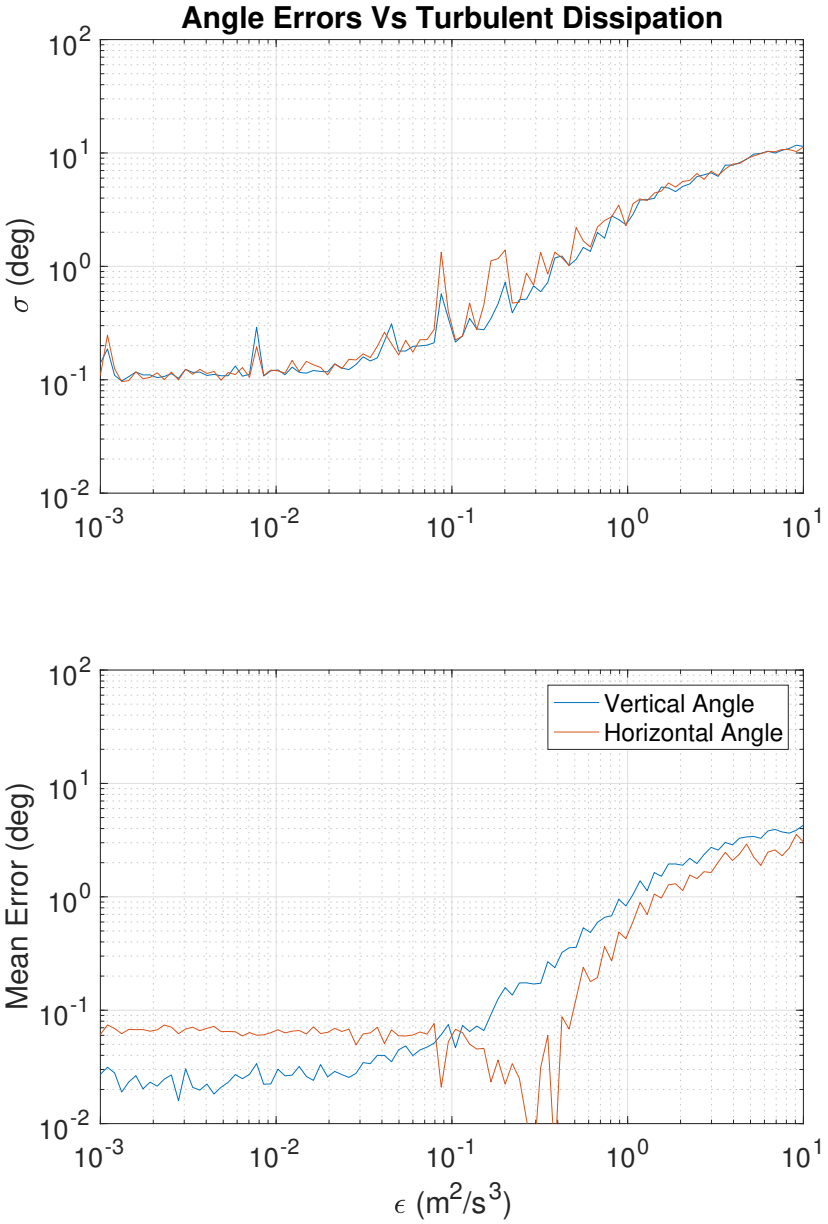


Figure 4.16: Plots of standard deviation and mean errors in flow angles against the turbulent dissipation rate determined from simulation.

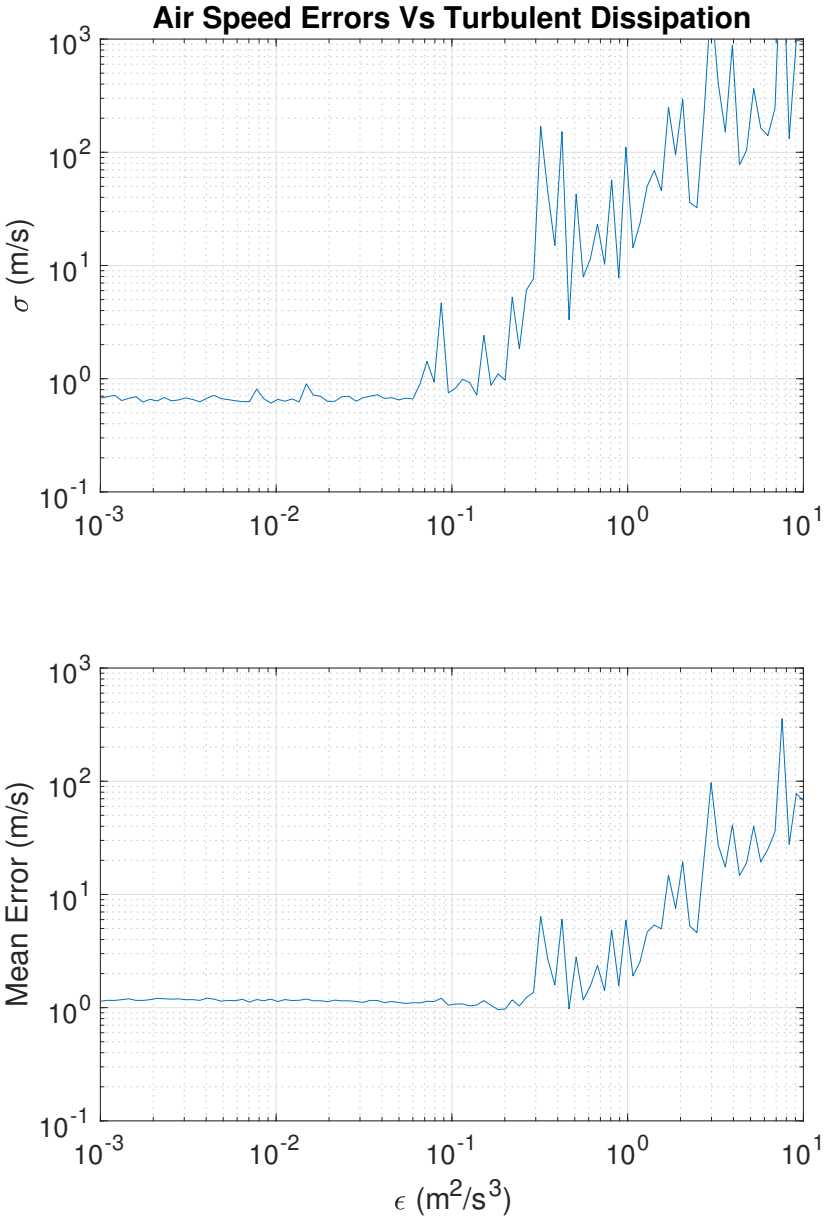


Figure 4.17: Plots of standard deviation and mean errors in air speed against the turbulent dissipation rate determined from simulation.

4.7 Background Effects

The background image used for this test is shown in figure 4.18. The image is focused at a short range as would be done for a measurement system, hence it is blurred. The impacts of increasing the simulated background strength ratio, t_b, are shown in figures 4.19, and 4.20.

Figure 4.18: Photograph used for background image in simulation

The angle errors resulting from increasing t_b are generally small. They remain in the low sub-degree range through $t_b = 1$ and then begin to increase as t_b climbs above the streak returns strength. Results show that the method is relatively robust against t_b, a necessary quality for a potentially field-capable system.

Air speed errors are also shown to be relatively insensitive to background signal strength with the image used here. Accuracy across the range remained on the order of $1\frac{\mathrm{m}}{\mathrm{s}}$. This performance is similar to what was predicted in the base case in section 4.5. Curiously, at higher values of t_b, bias decreases somewhat. This may be a result of streaks getting effectively shortened as brighter background imagery reduces streak contrast and thus makes

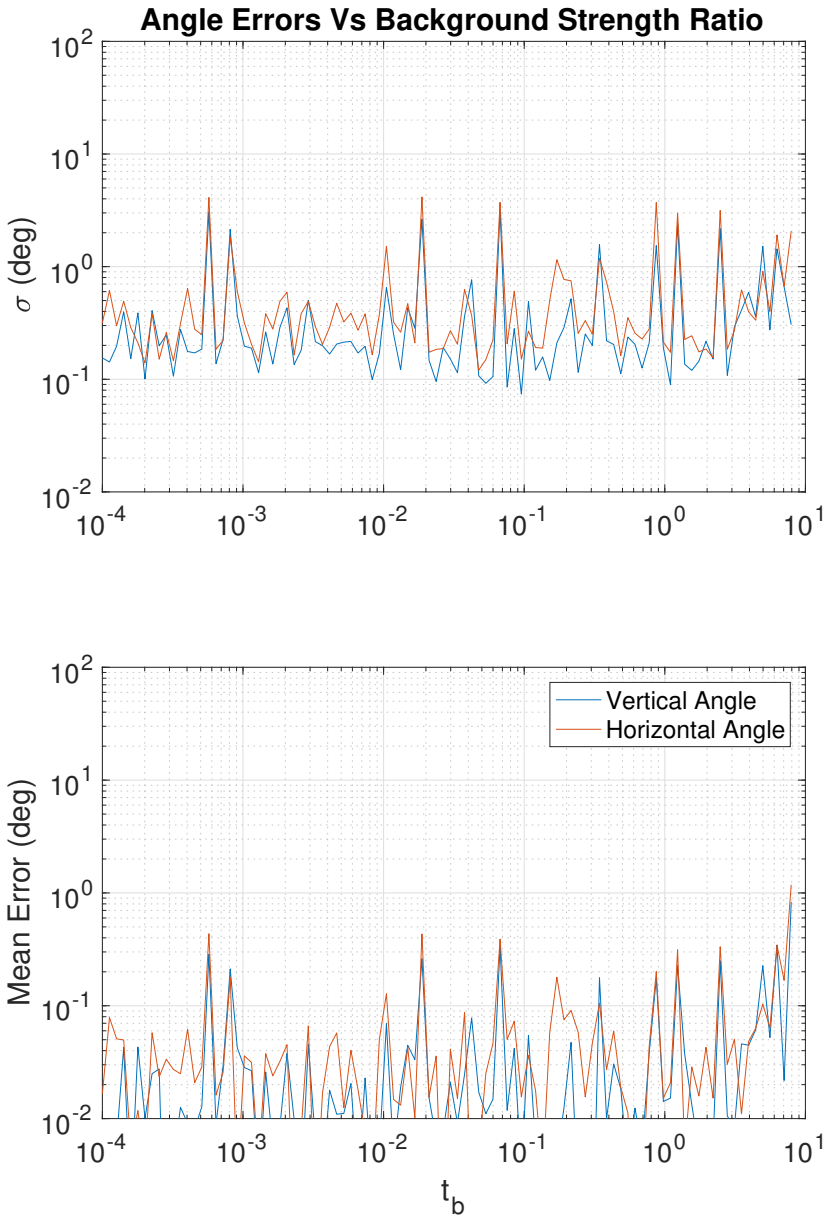


Figure 4.19: Plots of standard deviation and mean errors in flow angles against t_b determined from simulation.

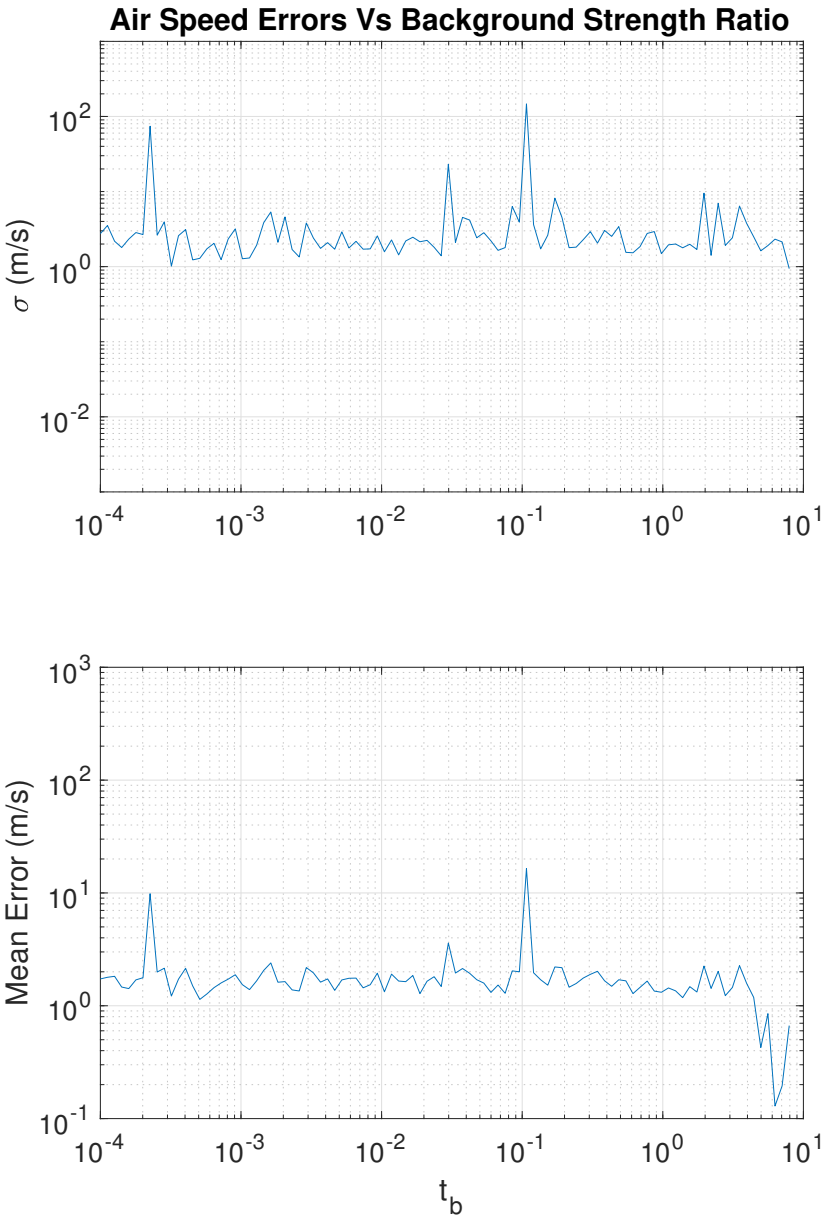


Figure 4.20: Plots of standard deviation and mean errors in air speed against t_b determined from simulation.

the generally more blurred streak ends weaker and results in edge detection not picking up the entire streak. This may effectively shorten many detected streaks introducing a negative bias that counters the typical positive bias.

Because background signals may vary widely based on environment and camera settings a definitive statement about the method's sensitivity to all background signals cannot be made, but these simulations do indicate that the method at least may be relatively robust to this error source. A future testing campaign to provide more definitive results could make use of a variety of background images captured from sUAS in flight.

4.8 Image Noise

The impacts of increasing the simulated image noise are shown in figures 4.21, and 4.22. The data show that the method is relatively robust to noise over the ranges tested. However, the values of t_n used in these simulations was limited by an abrupt breakdown in the method slightly above the upper end of the range shown in figures 4.21, and 4.22. This breakdown results from the point where the noise is so strong that even after the Gaussian blurring in the edge detection, the noise is still detected as possible streak returns. This results in an abrupt jump in the number of streaks from a few seen real streaks to many thousands of spurious returns. This results in the solution method requiring a massive amount of processing time, making Monte Carlo runs prohibitively expensive.

Air speed errors were similar to those seen in the base case with no discernible trends against t_n observed over the range tested. Again the method breaks down slightly above the range shown. A possible future improvement would be to quantify the noise in the image prior to processing and either skip extremely noisy images or increase the strength of the Gaussian blur step in the edge detection.

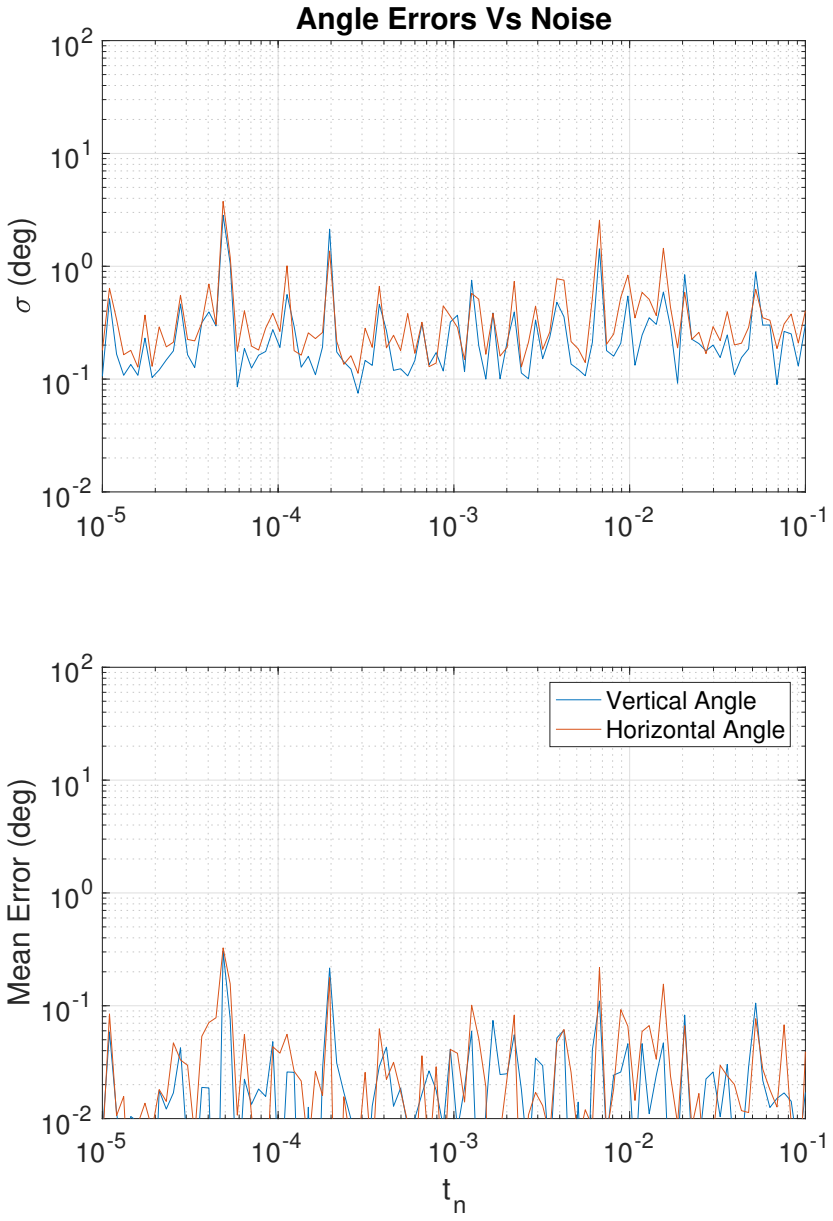


Figure 4.21: Plots of standard deviation and mean errors in flow angles against Gaussian noise parameterized by t_n determined from simulation.

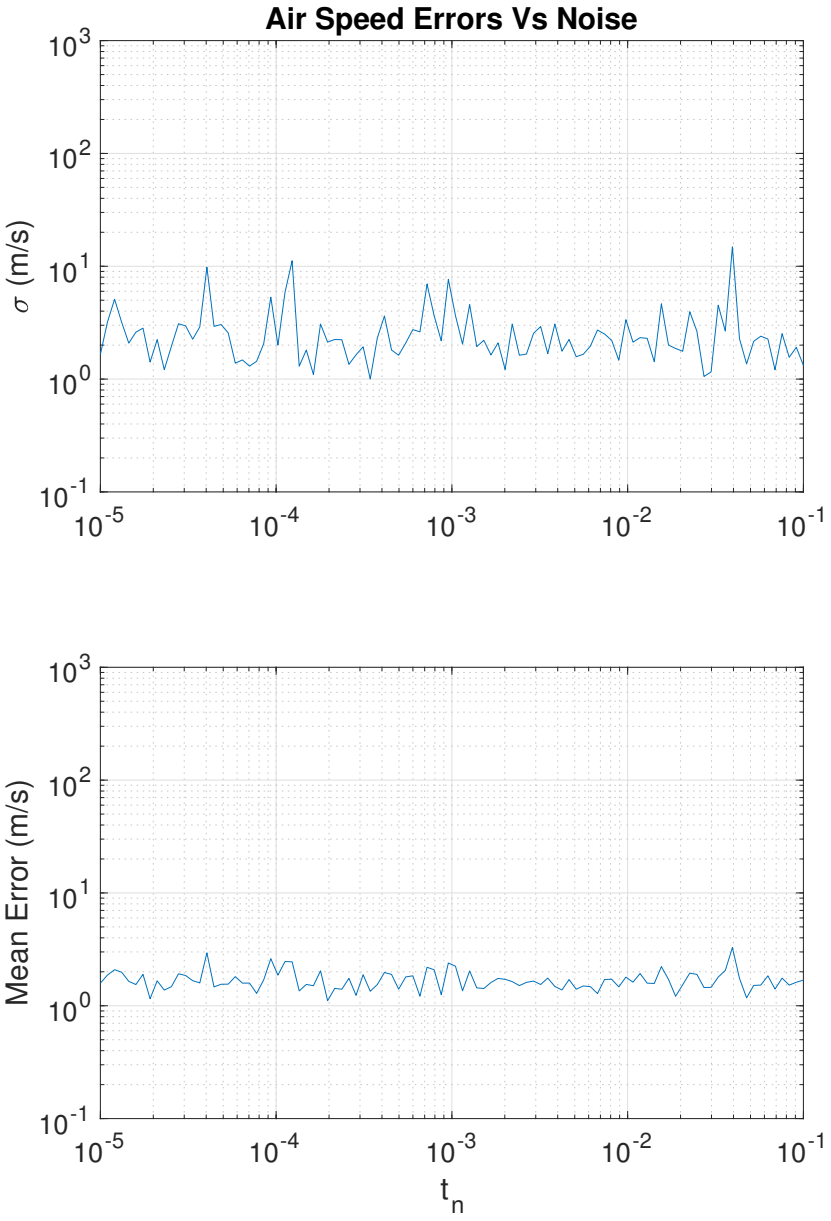


Figure 4.22: Plots of standard deviation and mean errors in air speed against Gaussian noise parameterized by t_n determined from simulation.

4.9 Lens Distortion

The impacts of a basic 1st order lens distortion are shown in figures 4.23, and 4.24. Negative 1st order values lead to barrel distortion, as illustrated in figure 4.25 showing the most extreme negative case simulated at $K_1 = -1.5 \times 10^{-8}$. Positive 1st order values lead to pincushion distortion as illustrated in figure 4.26 showing the most extreme positive case simulated at $K_1 = 1.5 \times 10^{-8}$. Because the distortion is purely radial in this model a small angle of attack was given to simulate a realistic flight condition so that the streaks would not be oriented along the direction of distortion.

The horizontal and vertical angle accuracy show a similar trend, increasing away from the zero distortion case. The errors in the vertical angle have greater magnitude, since the flow is off the center axis in the vertical but not the horizontal direction. Both also show larger errors for the negative values (barrel distortion) than the positive values (pincushion distortion). This may be because extreme levels of barrel distortion pull streaks out of the imaged area, while extreme pincushion distortion pushes streaks into the imaged area. The loss of streaks in strong barrel distortion could account for the larger errors.

Air speed error trends are less clear. Some extremely large errors occur in the barrel distortion case (likely a consequence of losing streaks off the image edges) but there is not a clear trend overall. This unaccounted for distortion is a form of mapping error. It can be corrected for by replacing the pinhole model with a model correctly matching the distorted lens.

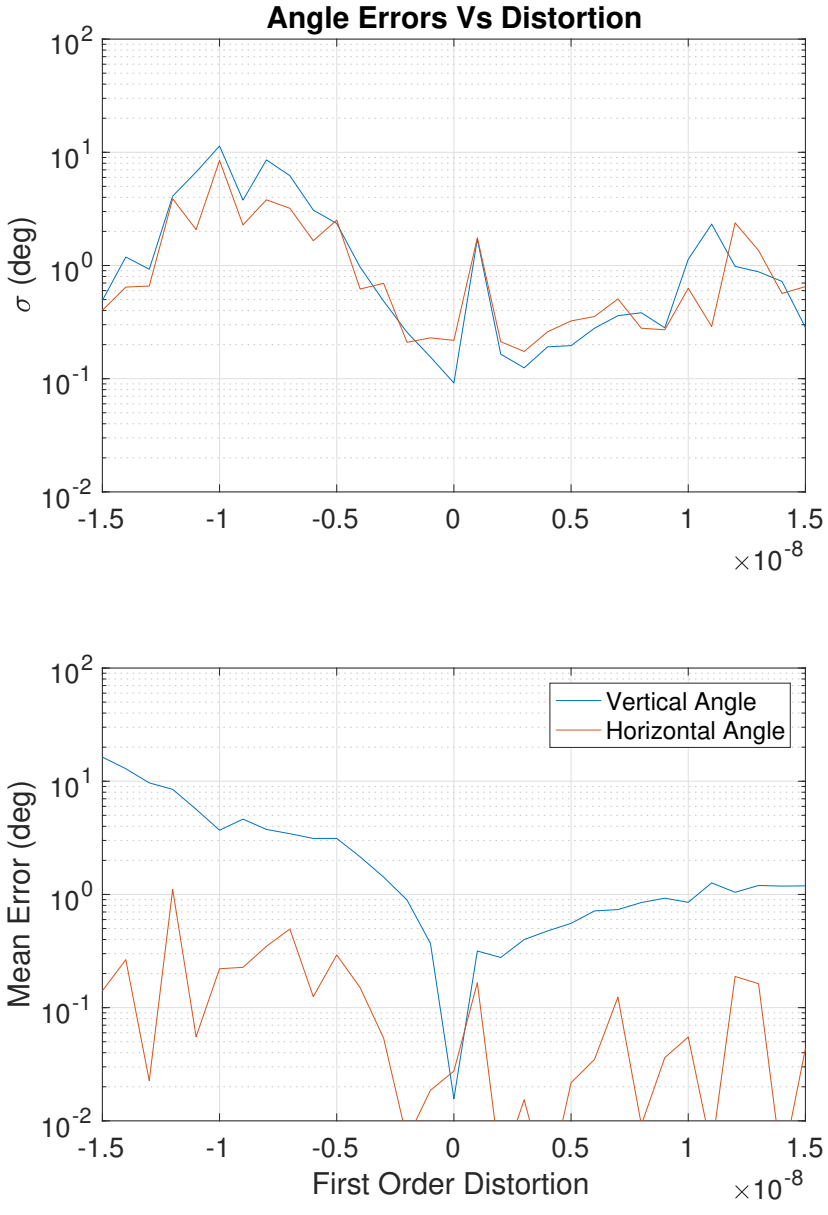


Figure 4.23: Plots of standard deviation and mean errors in flow angles against the first order distortion parameter determined from simulation.

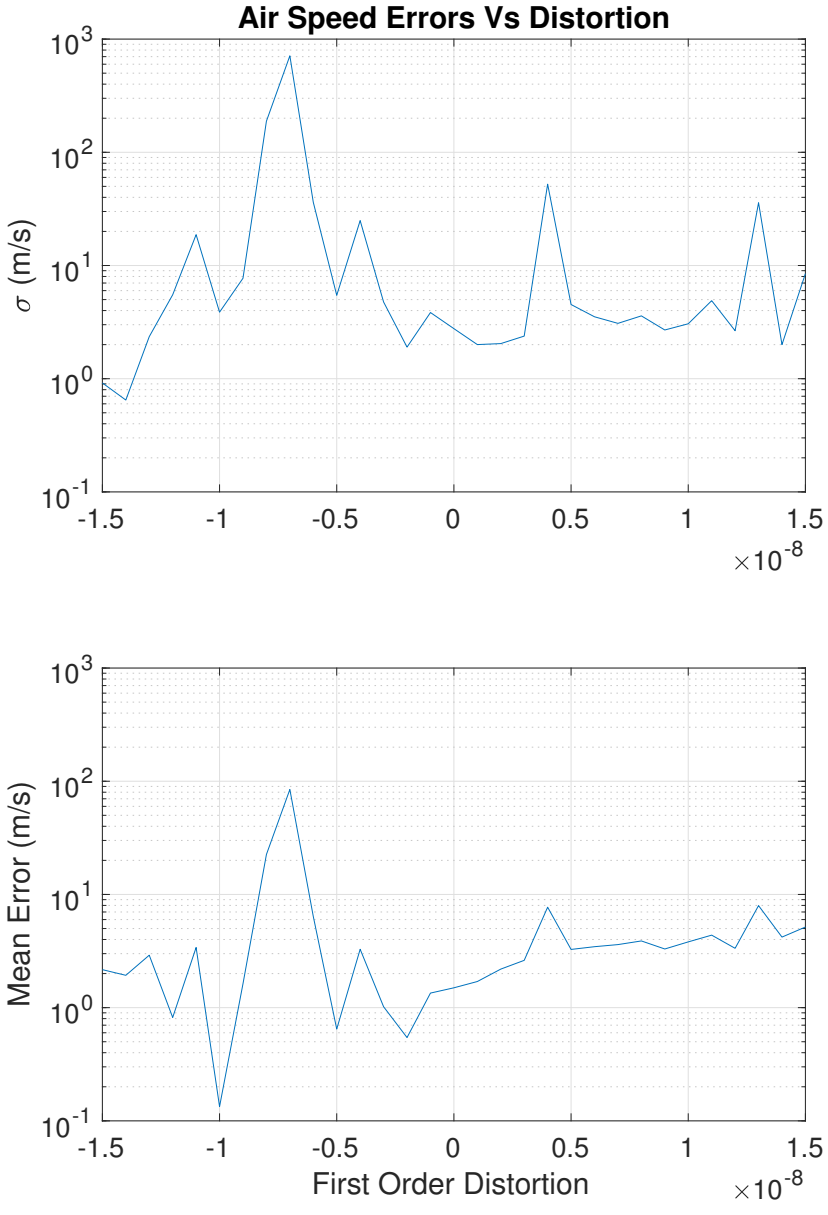


Figure 4.24: Plots of standard deviation and mean errors in air speed against the first order distortion parameter determined from simulation.

Figure 4.25: Barrel distortion under first order model used in the simulation at the extreme negative case: $K_1 = -1.5 \times 10^{-8}$.

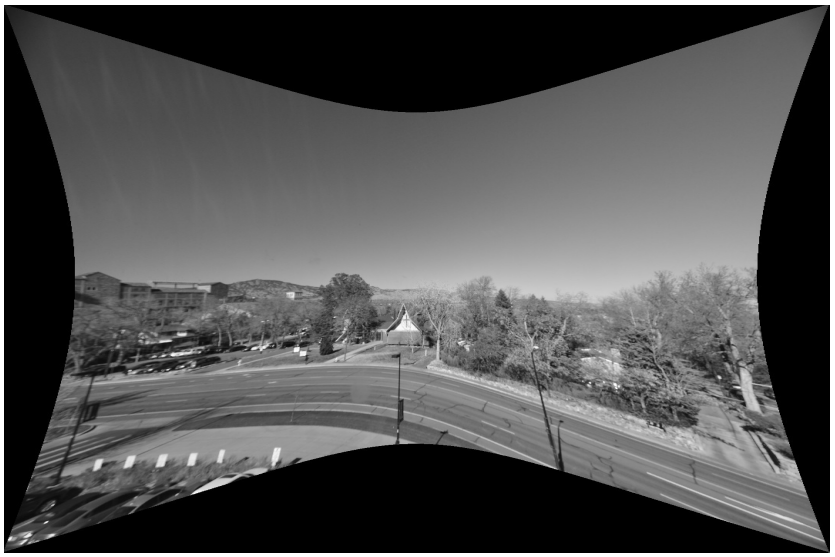


Figure 4.26: Pincushion distortion under first order model used in the simulation at the extreme positive case $K_1 = 1.5 \times 10^{-8}$.

Chapter 5

Simulation Verification

In order to verify the simulation results a limited wind tunnel testing campaign was conducted. Due to limited resources a full system characterization was not possible. Instead the tests aimed to determine if samples taken in the wind tunnel delivered performance similar to what was expected based on the simulation results shown in chapter 4.

Tests were conducted in the National Center for Atmospheric Research low speed wind tunnel. This tunnel uses a suck-through design for low-turbulence. The tunnel measurement equipment is capable of measuring air speed with a resolution of 0.01m/s and an accuracy of 0.1m/s [37]. The tunnel has a mount allowing single-degree sideslip angle accuracy.

A photograph of the test set up is shown in figure 5.1 and diagram is shown in figure 5.2. A Nikon D7200 camera was used with a 50mm Nikkor AF lens to capture images. The camera was mounted to a hinged piece of 80-20 aluminum attached to a long vertical piece of 80-20 that fit directly into the adjustable mount for the tunnel. The angle of attack was measured with a digital inclinometer allowing 0.1° accuracy. Though control of the angle was strictly by hand adjusting the hinge, limiting control to roughly ±1°. Camera control was performed via computer commanding a sequence of images (generally 30 or 60 images in a sequence). A pair of 60W LED lights were used to provide illumination in the volume ahead of the camera. Due to the diffuse nature of the light source, a focus limited approach was used with f-number 8 and subject distance 0.49m. This resulted in predicted image depth limits $D_n = 0.42$m and $D_f = 0.59$m.

Figure 5.1: The image capture portion of the test setup. The flocking paper was used to reduce bright reflections off the tunnel walls.

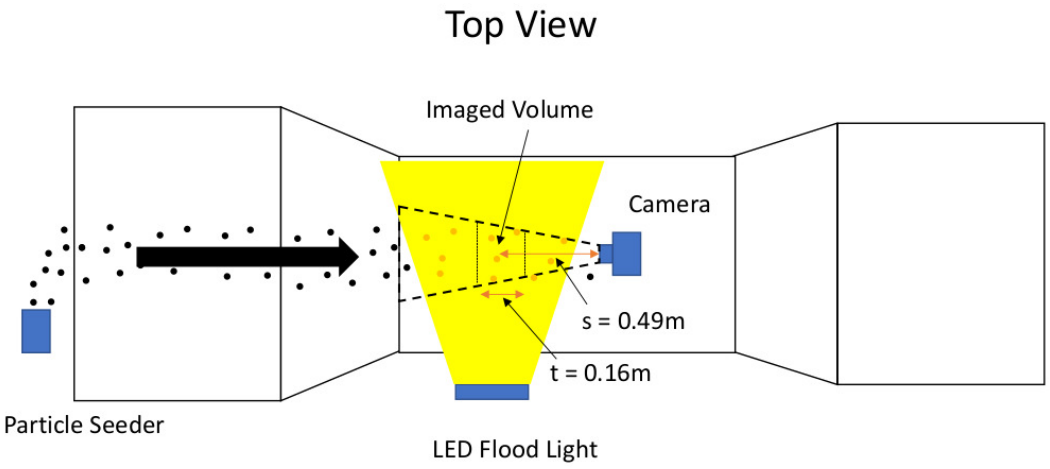


Figure 5.2: Diagram of the wind tunnel test setup.

Early studies found an insufficient number of large particles present in the wind tunnel room to use only ambient particles with the imaging and illumination equipment available, so additional particles were added through seeding. Seeding was provided by a crude fluidized bed particle seeder [55]. A photograph of the seeder is shown in figure 5.3 and a conceptual overview of the device is shown in figure. 5.4. This device used a battery powered pump to push air into a cavity below the seeding particles which were suspended on a layer of screen material. Particles were thus forced upward through a second screen to prevent large clumps from being thrown out of the device.

In practice it was difficult to reliably send the resulting particle stream through the image zone. The seeder was moved around throughout the sampling period to ensure at least some images were captured with particle returns.

For seeding particles, chalk was used. Due to its tendency to clump, it forms heterogeneous sized particle clouds providing an approximation to ambient atmospheric particulate. Additionally chalk was affordable within the limited test resources available. One issue with the seeding system used, was that it generally over-seeded the flow (providing an excessive particle density). The large number of particles accreted on the lens eventually blocking out particle returns. As a result, image sequence length had to be limited to allow regular lens cleaning.

Three different angles of attack $-5.9°$,$-0.8°$, and $9.8°$ were tested at two different sideslip angles $0°$ and $10°$ ($8°$ at $9.8°$ angle of attack, to avoid catching too much of the tunnel wall). These angles were limited due to the geometry of the tunnel and illumination setup. Angles had to be chosen that did not include significant amounts of either the brightly illuminated tunnel walls or the lights. These elements are so bright that they overpower the streak returns. The lights cause excessive lens flare if the camera is pointed close to their direction.

Three air speeds were tested at each orientation 5m/s, 10m/s, and 15m/s. The upper air speeds were limited to avoid damage to flocking paper used to reduce reflections from

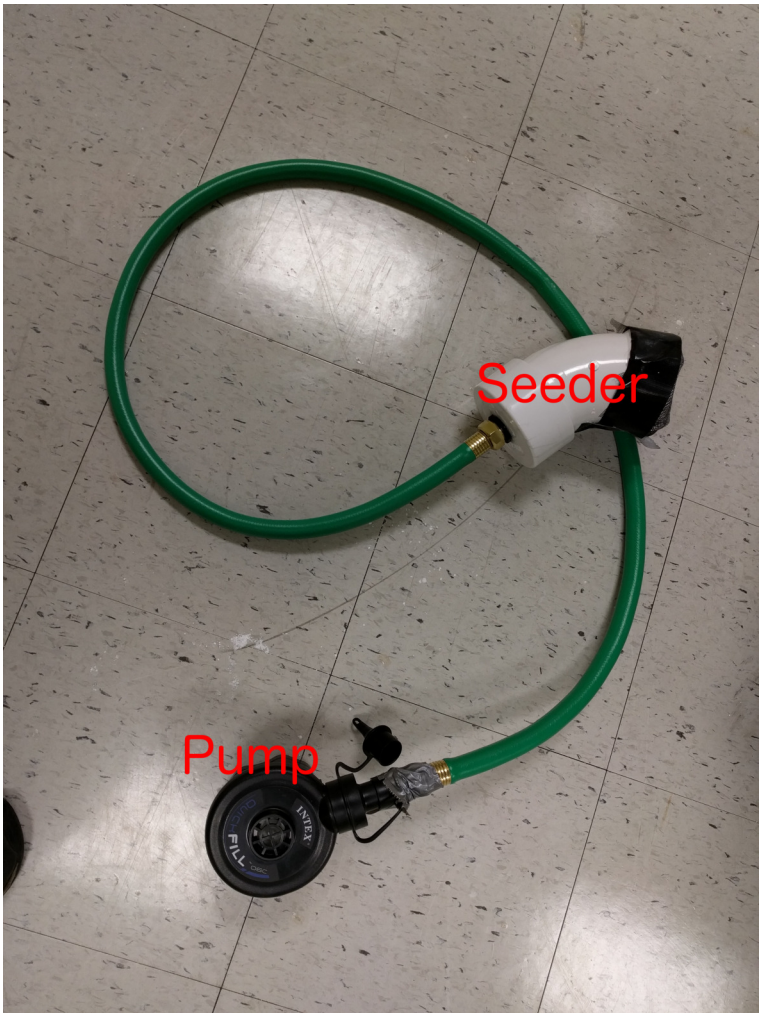


Figure 5.3: The particle seeder used for testing with chalk dust.

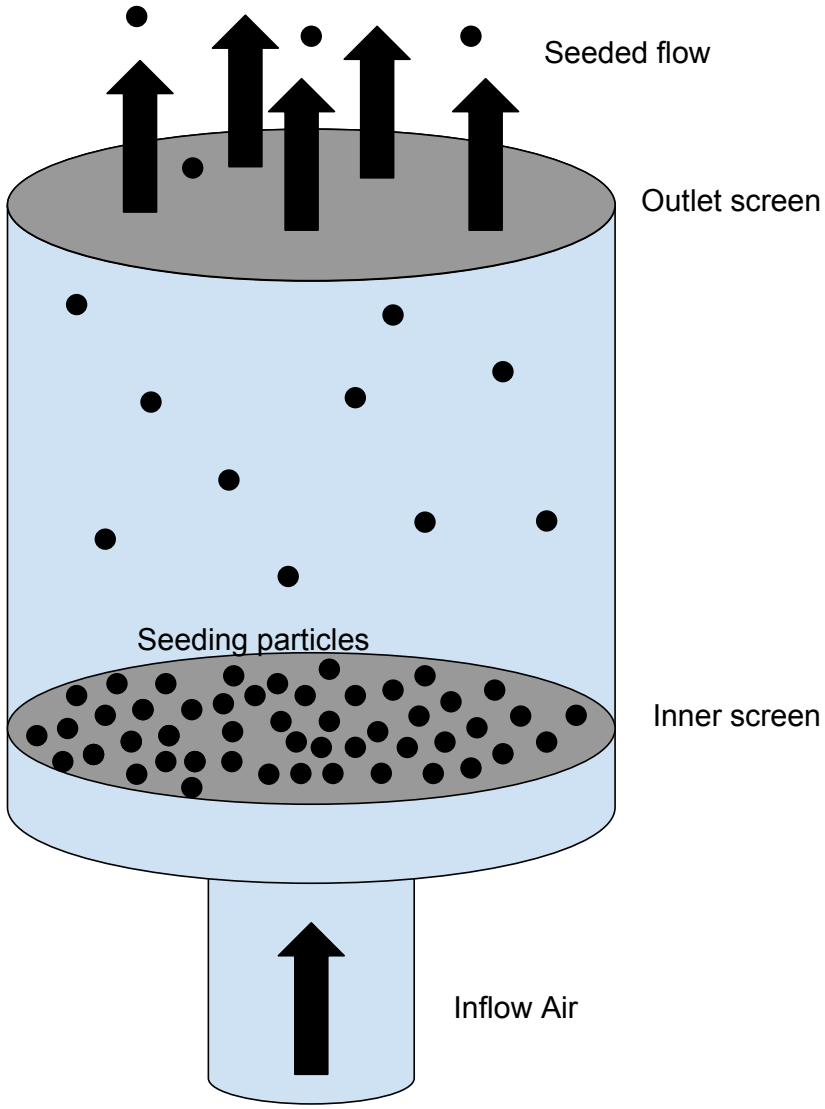


Figure 5.4: Diagram of the particle seeder used for testing.

the tunnel wall.

5.1 Testing Issues

The testing campaign was limited in scope and the techniques used limited the total number of samples captured. The biggest single limitation was getting the stream of seeding particulate to pass through the imaged volume consistently. This led to a number of images that lacked any particle returns. An improved seeding approach with a lower seeding frequency covering a wider cross-section of the tunnel would be necessary to provide a higher rate of successful sampling (images with particle streak returns).

A secondary issue was catching reflections from the tunnel wall. The white tunnel walls led to very strong returns in images taken at the positive angle of attack and positive side slip condition. Using a camera with automatic exposure control on the sensor this effectively reduced the brightness of streaks making them more difficult to detect and was analogous to the high background signal condition explored in the simulations in chapter 4.

A third issue was generally weak particle returns in all positive side slip angle conditions. This is potentially a result of the particle scattering phase functions being generally low between the narrow cone of backscattering around 180° and 90°. However since the scattering phase functions for coarse mode chalk particles shown in this work were simulated on spheres the accuracy of these simulations is necessarily questionable for real particles that do not necessarily have spherical shape. These weak returns led to many images lacking streaks that could be detected with te settings used. While adjusting the streak detection parameters may have allowed for detection of these streaks, the goal of the tests was to match simulation conditions, and manual tuning of streak detection is not realistic for a system to be used in the field.

Due to the limited number of samples, testing results are not adequate to make definitive conclusions on method performance;, however, the preliminary data collected did show promising performance. These results are shown in the following sections.

5.2 Centered Flow Results

Figure 5.5 shows an example image of particle streaks taken with the camera pointed at a $-0.8°$ angle of attack and $0°$ side slip angle. Figure 5.6 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.7 shows a detailed view of the points near the true angles. Points in gray are those measured at other positions. Figure 5.8, shows the distribution of air speed errors from the estimates made in this position.

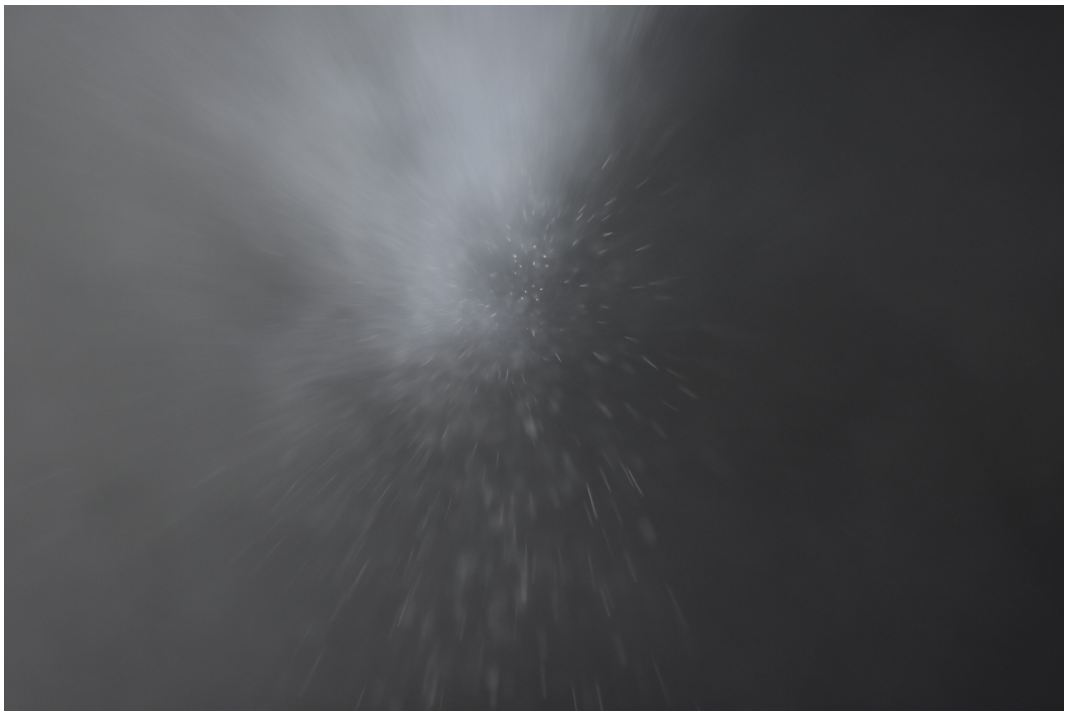


Figure 5.5: Example image of particle streaks in the centered flow case.

The results show a relatively tight clustering in angle errors. The spread of errors is a bit wider than predicted in simulation Most errors are in the sub-degree range. The horizontal angle measurements show a slight bias (around $-1°$); however, this angle could only be measured to an accuracy on the order of $1°$ so this may simply be an error in the measurement of the camera position.

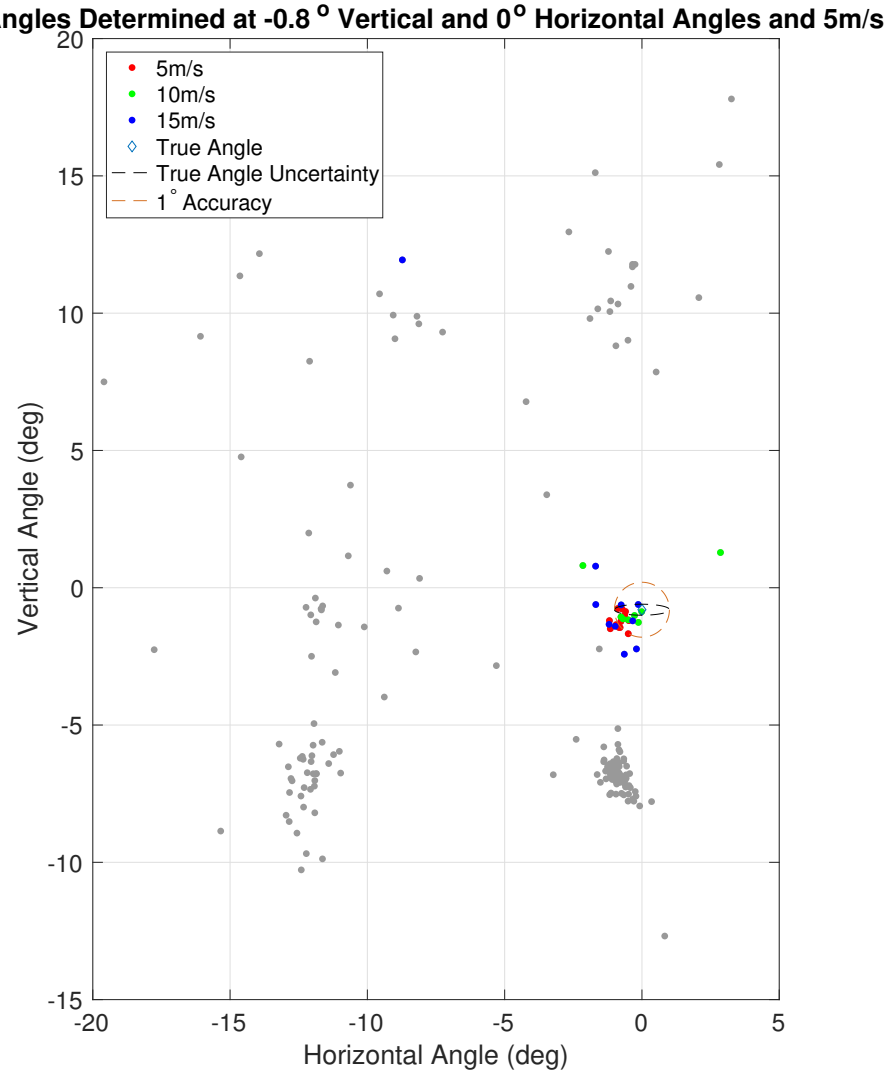


Figure 5.6: Flow angle estimates in centered position wind tunnel tests.

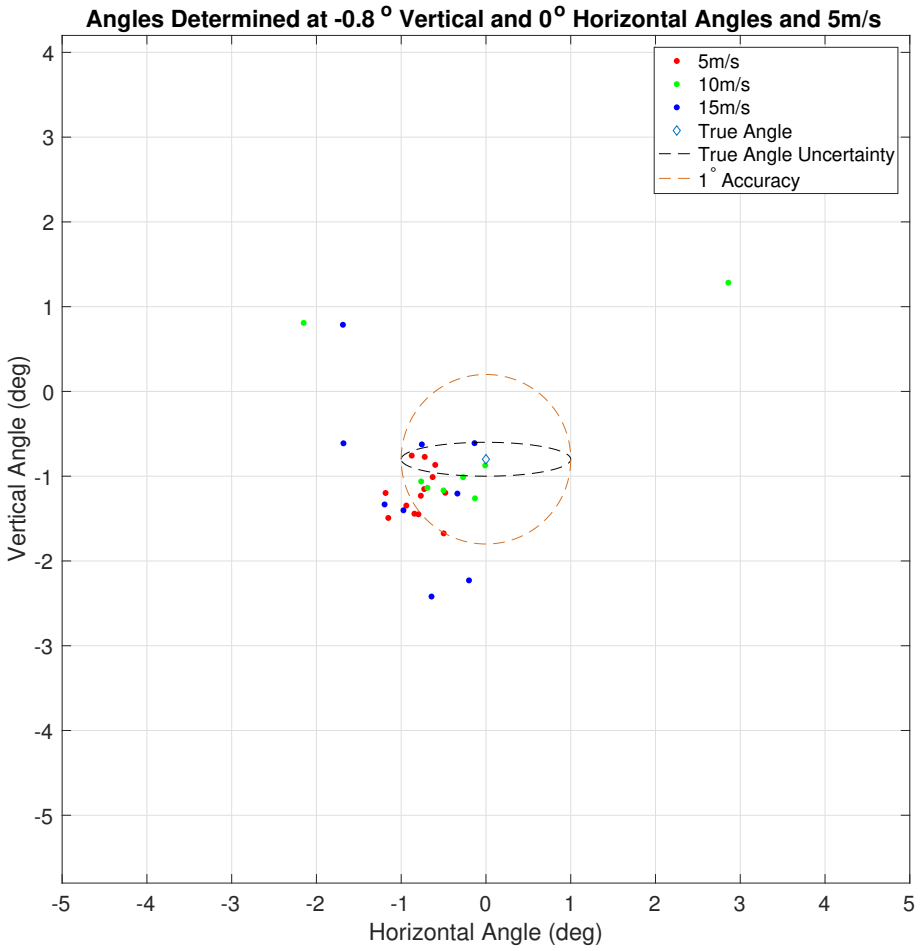


Figure 5.7: Flow angle estimates in centered position wind tunnel tests, detailed view.

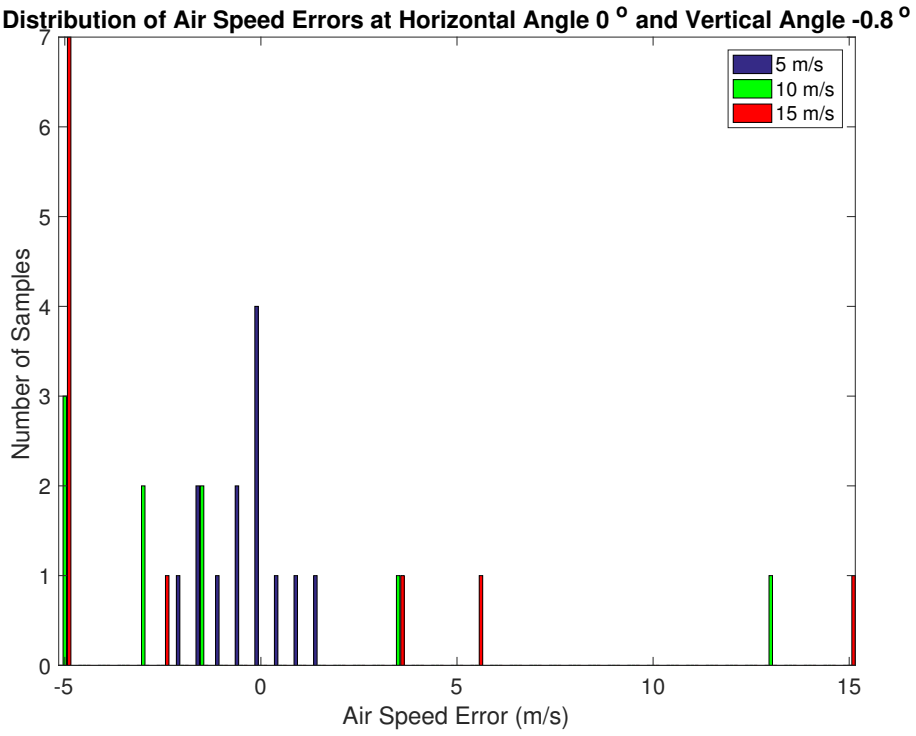


Figure 5.8: Distribution of air speed errors in centered position wind tunnel tests.

There is a slight bias to negative readings in the vertical angle, roughly. This could be the result of a small measurement error in the vertical angle (the measurement was made at the front glass of the camera which may not be perfectly aligned with the sensor). Another possibility is that the larger particles have a slight downward trajectory due to gravity. This would also explain why the errors shift more positive at higher flow speeds. Angle errors are somewhat larger than the simulation results predict, but some amount of additional turbulence is injected into the flow by the seeding device and the seeding operator creates a minor blockage at the tunnel entrance which contributes additional turbulence. This added turbulence likely results in a slightly larger standard deviation of angle errors as predicted in section 4.6.

Air speed errors are generally large, however at low flow speeds the errors are smaller, generally within $1 - 2$m/s. This compares favorably to simulation results which predicted a bias of roughly 1m/s and standard deviation of around 2m/s for a focus-limited case. Interestingly, the bias seen in the simulation is not apparent in the 5m/s air speed results, however far more data is needed to make a determination about whether the real results are or are not biased. The lower speed results are likely superior for two main reasons. Firstly, the slower flow speeds lead to brighter streaks and thus more accurate detection of streak end points. Secondly, particle displacements are shorter and thus the probability of a streak being shortened by the particle entering or leaving the volume during the exposure are lower.

5.3 10° Side Slip and -0.8° Angle of Attack Flow Results

Figure 5.9 shows an example image of particle streaks taken with the camera pointed at a -0.8° angle of attack and 10° side slip angle. Figure 5.10 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.11 shows a detailed view of the points near the true angles. Points in gray are those measured at other positions. Figure 5.12, shows the distribution of air speed errors from the estimates made in this position.

Figure 5.9: Example image of particle streaks in the 10° side slip and −0.8° angle of attack flow case.

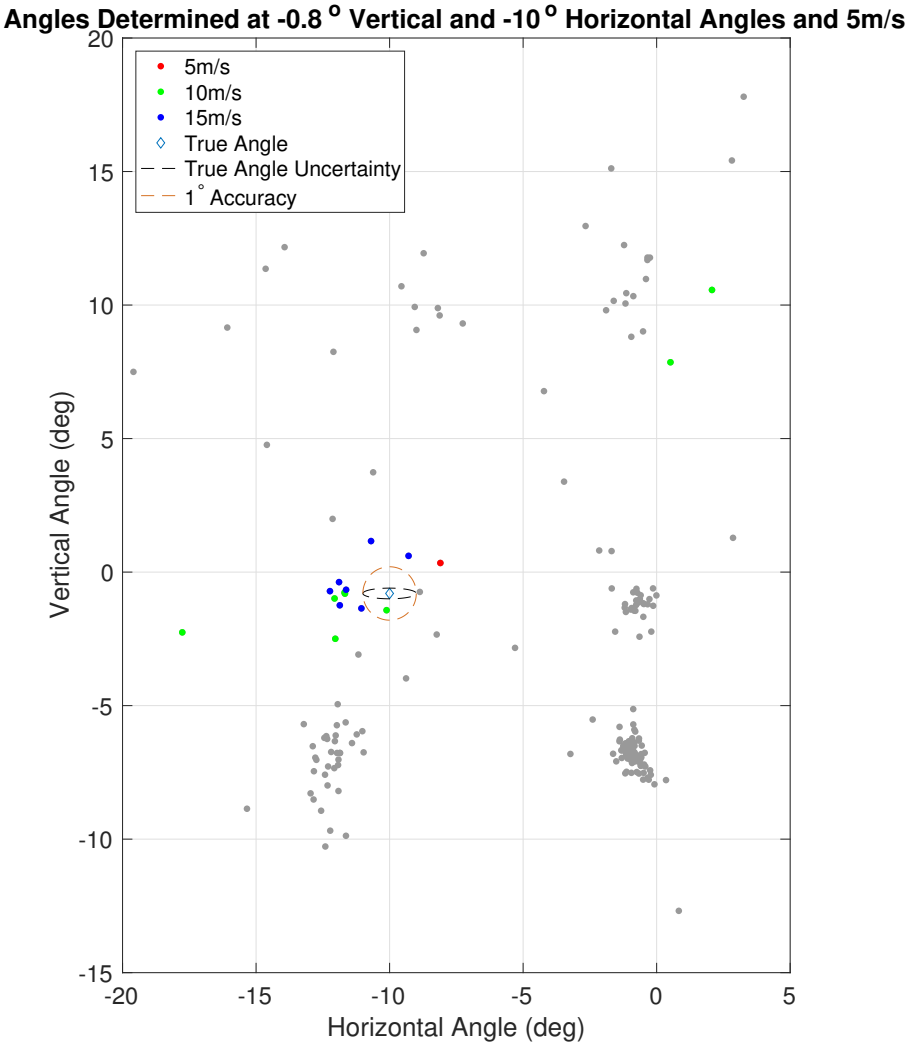


Figure 5.10: Flow angle estimates in 10° side slip and −0.8° angle of attack wind tunnel tests.

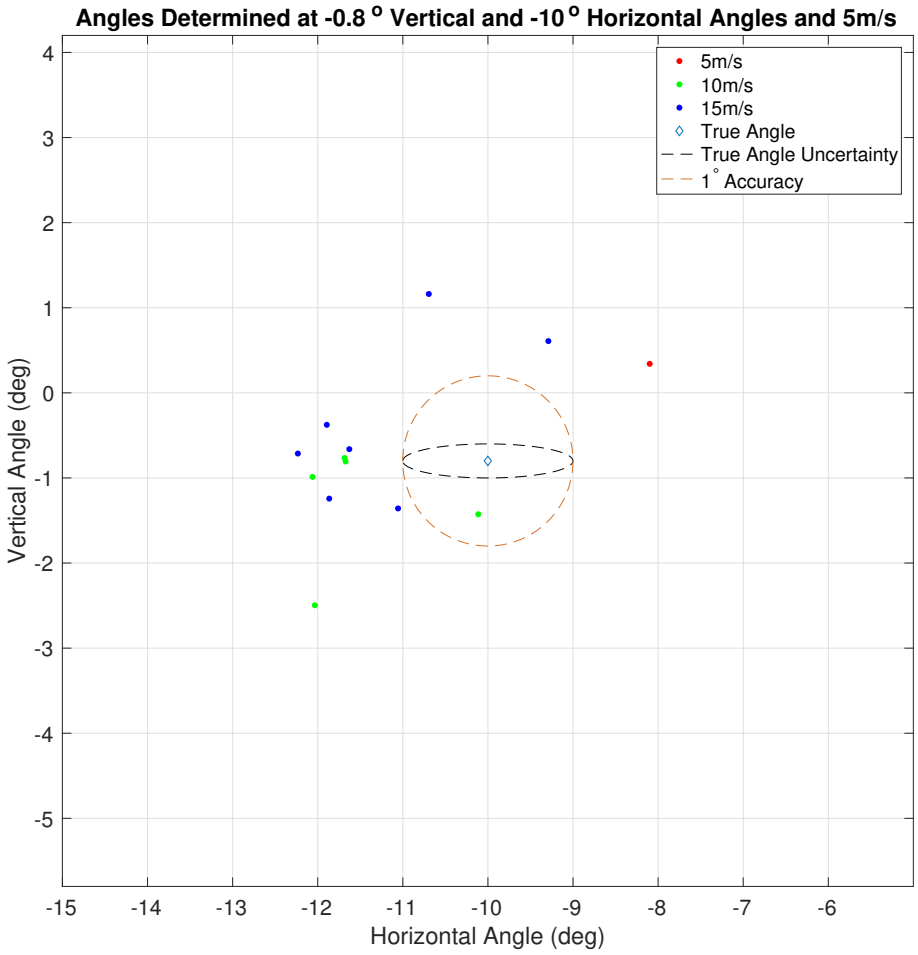


Figure 5.11: Flow angle estimates in 10° side slip and −0.8° angle of attack wind tunnel tests, detailed view.

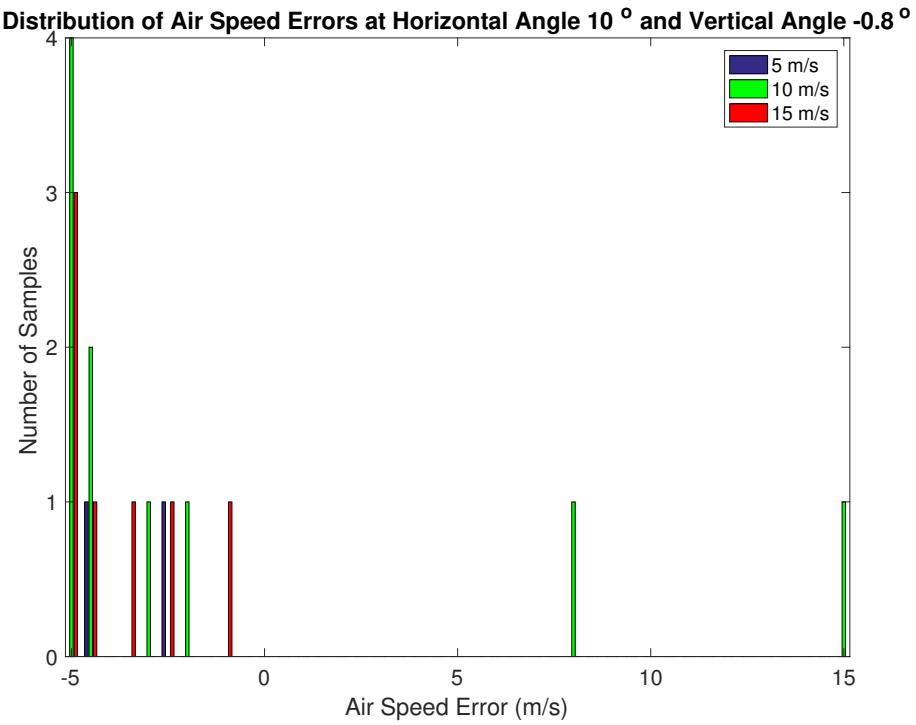


Figure 5.12: Distribution of air speed errors in 10° side slip and −0.8° angle of attack wind tunnel tests.

Horizontal and vertical angle errors are somewhat larger in this case, generally falling outside the 1° envelope and showing some potential bias. The horizontal angle errors are mostly around 2° off which is substantially greater than the predicted errors from simulation. this error also appears to be more of a bias than merely a wider random distribution (though the limited number of samples precludes a definitive answer on this). The probable reason for this is that the pinhole model used for mapping image positions to three-dimensional directions is not perfect for this real camera system. The pinhole model was used for a simple implementation in this method but can easily be replaced with a different mapping. As was shown in chapter 4, mapping errors can lead to both increased bias and a wider distribution of estimates. A possible further source of error in this orientation is reduced intensity of light scattering. The scattering angle at this orientation to the camera is roughly 100°, which lies in a region of reduced scattering for chalk particles in the tens of micron range under the assumption of spherical particles as described in section 2.4. As a result, light returns from particles in this test configuration may be dimmer leading to reduced streak detection accuracy.

Air speed errors are also much larger than the 5m/s case in the centered flow alignment, with many being 5m/s or more too low. While the predicted performance from the simulation for the focus-limited approach was positive-biased air speeds, the airspeeds in this case are mostly estimated too low. Likely reasons for this are the the same mapping errors affecting the angle estimates combined with a generally lower quality of images captured at this angle resulting in streak lengths being inaccurately detected.

5.4 8° Side Slip and 9.8° Angle of Attack Flow Results

Figure 5.13 shows an example image of particle streaks taken with the camera pointed at a 9.8° angle of attack and 8° side slip angle. Figure 5.14 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.15 shows a detailed view of the points near the true angles. Points in gray are those

measured at other positions. Figure 5.16, shows the distribution of air speed errors from the estimates made in this position.

Again, angle errors are larger than in the centered case which likely results from mapping errors. Angle errors are generally near or above 1° which is poorer than the simulation prediction. Few samples were collected at this orientation and to what degree measurements are biased as opposed to simply random cannot be stated. Air speed errors remain significant and again, tend to be too low, mostly from $1-4$m/s. Again this disagrees with the predicted performance of positive bias from simulation. Streak detection was complicated in this case by the bright tunnel wall, as can be seen in figure 5.13 but still allowed for direction sensing in most cases. The main issue this caused was that the camera's automatic exposure adjustment, with the bright tunnel wall in the field of view, dimmed streaks, making for more difficult detection and thus likely contributed to increased error. The potentially reduced scatter returns at this side slip angle also may have contributed.

5.5 0° Side Slip and 9.8° Angle of Attack Flow Results

Figure 5.17 shows an example image of particle streaks taken with the camera pointed at a 9.8° angle of attack and 0° side slip angle. Figure 5.18 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.19 shows a detailed view of the points near the true angles. Points in gray are those measured at other positions. Figure 5.20, shows the distribution of air speed errors from the estimates made in this position.

Angle and air speed accuracies were poor in this case. A wide distribution of angle estimates is seen in the data, with multiple estimates off by 5° or more, far worse than predicted under the simulation. The 15m/s data seems to be the poorest, likely because the fainter streaks in this case lead to inaccurate detection. The apparent bias in horizontal angle of around 1° may again result from simple incorrect measurement of the horizontal angle. While mapping error likely played a part, overall low quality of captured imagery

Figure 5.13: Example image of particle streaks in the 8° side slip and 9.8° angle of attack flow case.

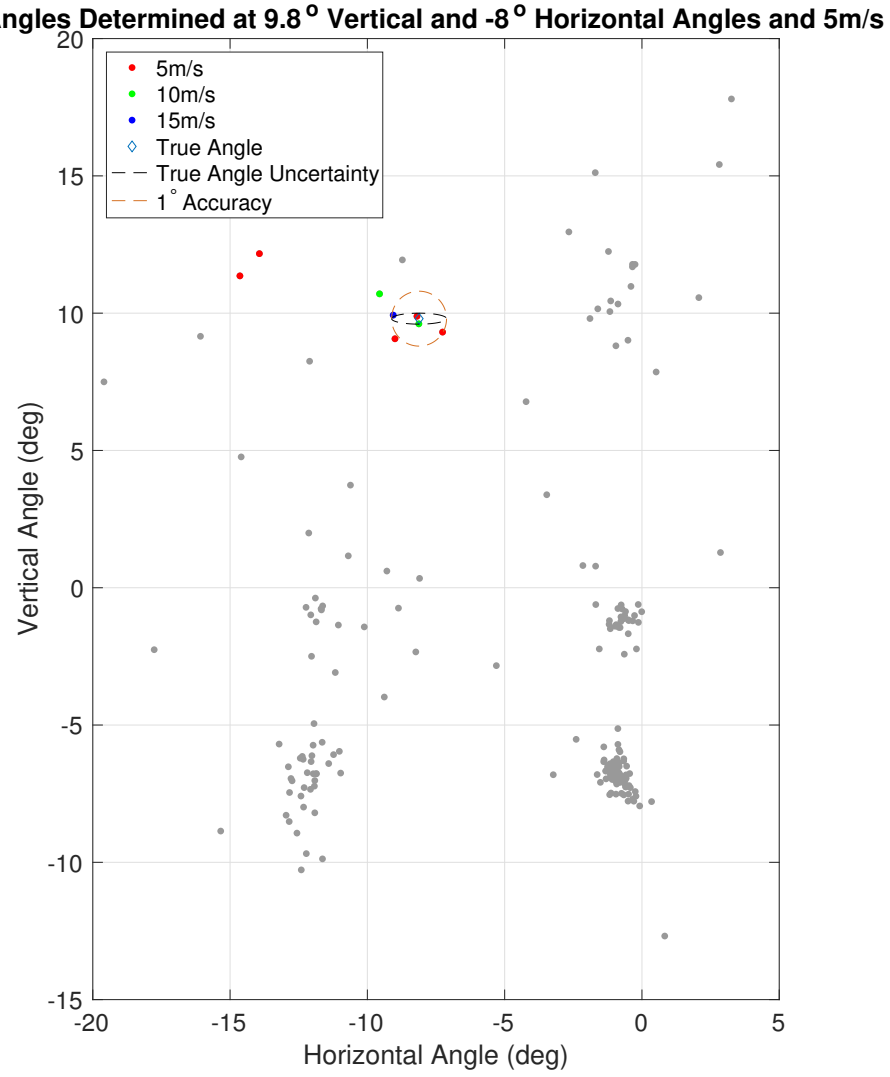


Figure 5.14: Flow angle estimates in 8° side slip and 9.8° angle of attack wind tunnel tests.

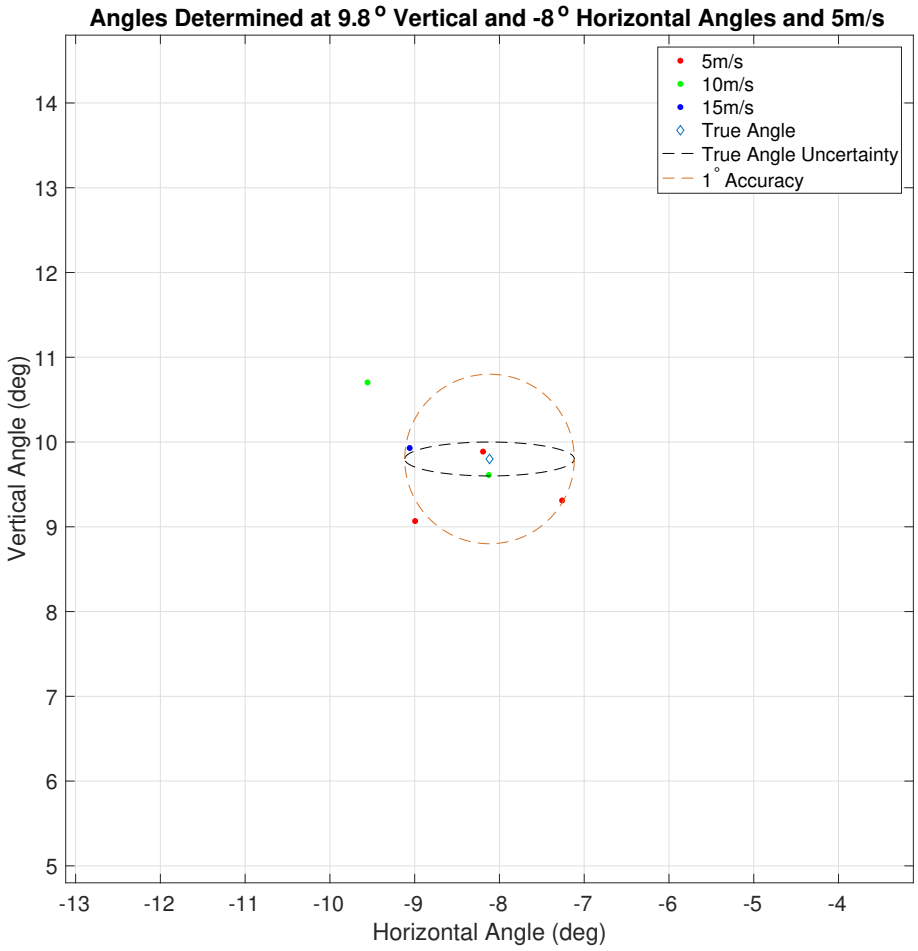


Figure 5.15: Flow angle estimates in 8° side slip and 9.8° angle of attack wind tunnel tests, detailed view.

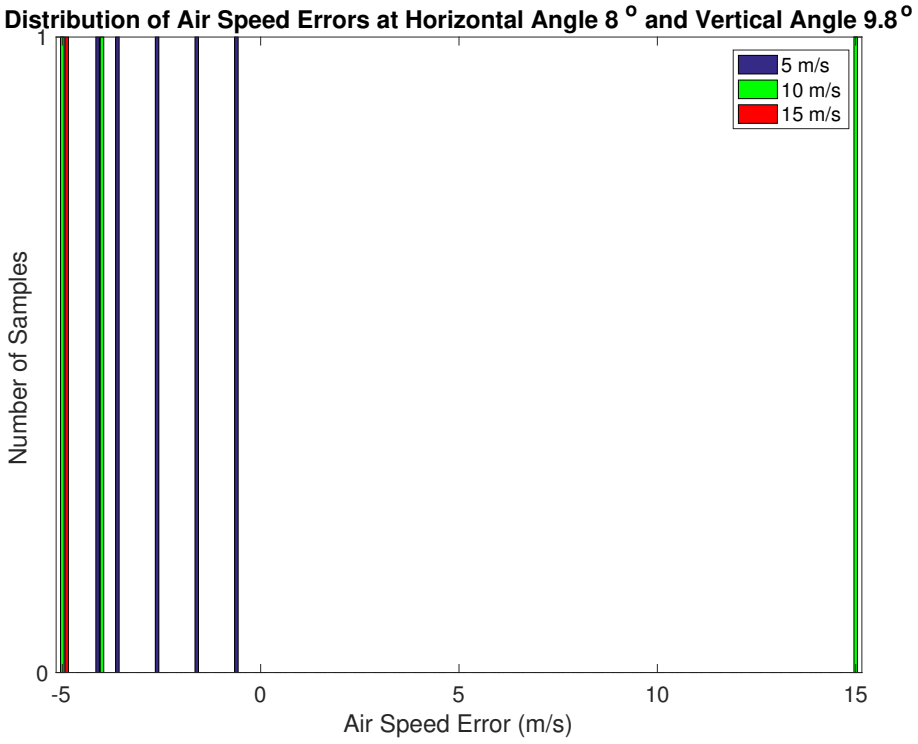


Figure 5.16: Distribution of air speed errors in 8° side slip and 9.8° angle of attack wind tunnel tests.

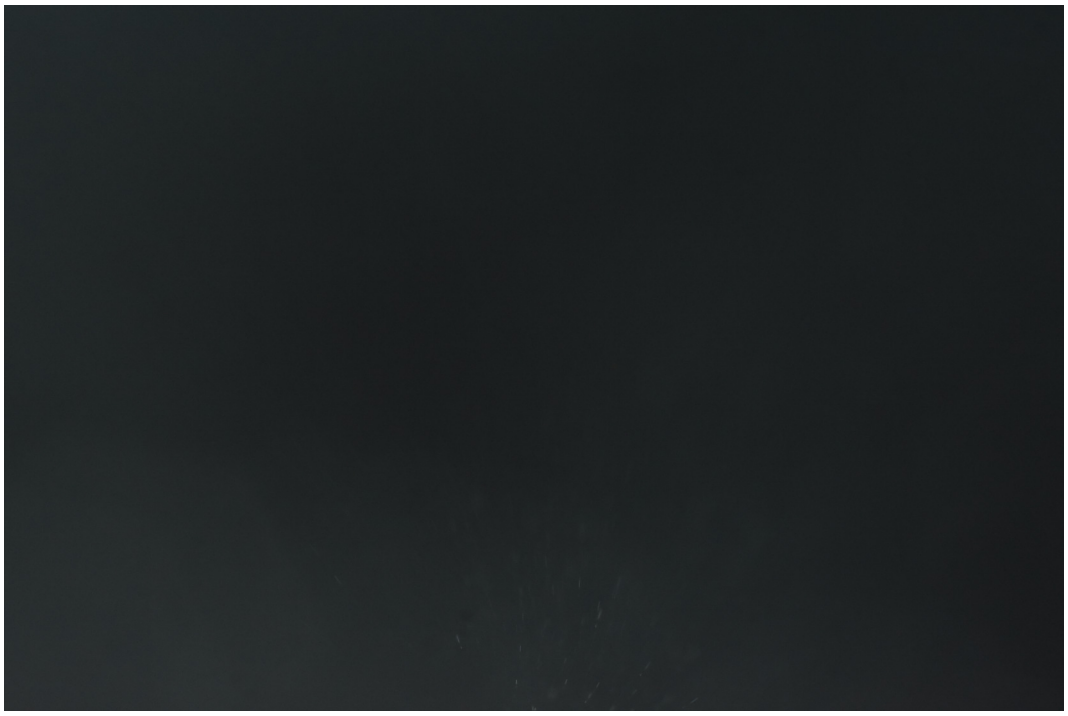


Figure 5.17: Example image of particle streaks in the 0° side slip and 9.8° angle of attack flow case.

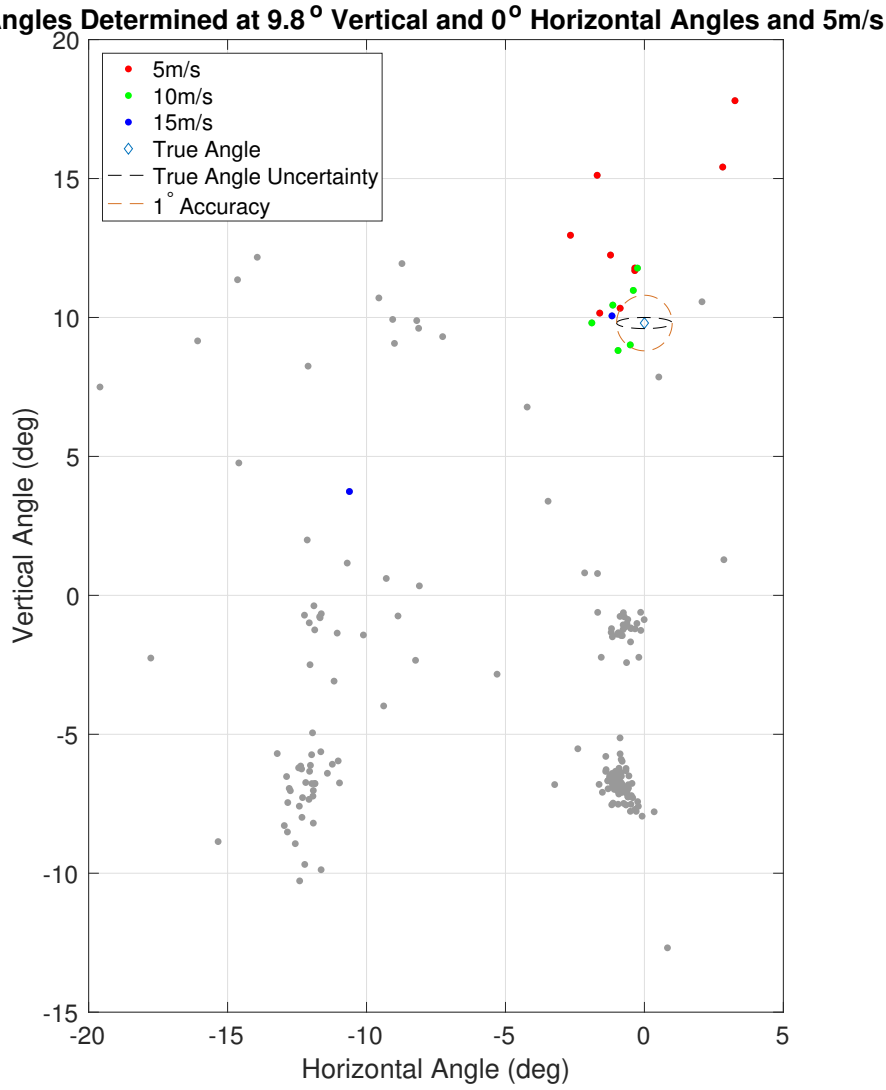


Figure 5.18: Flow angle estimates in 0° side slip and 9.8° angle of attack wind tunnel tests.

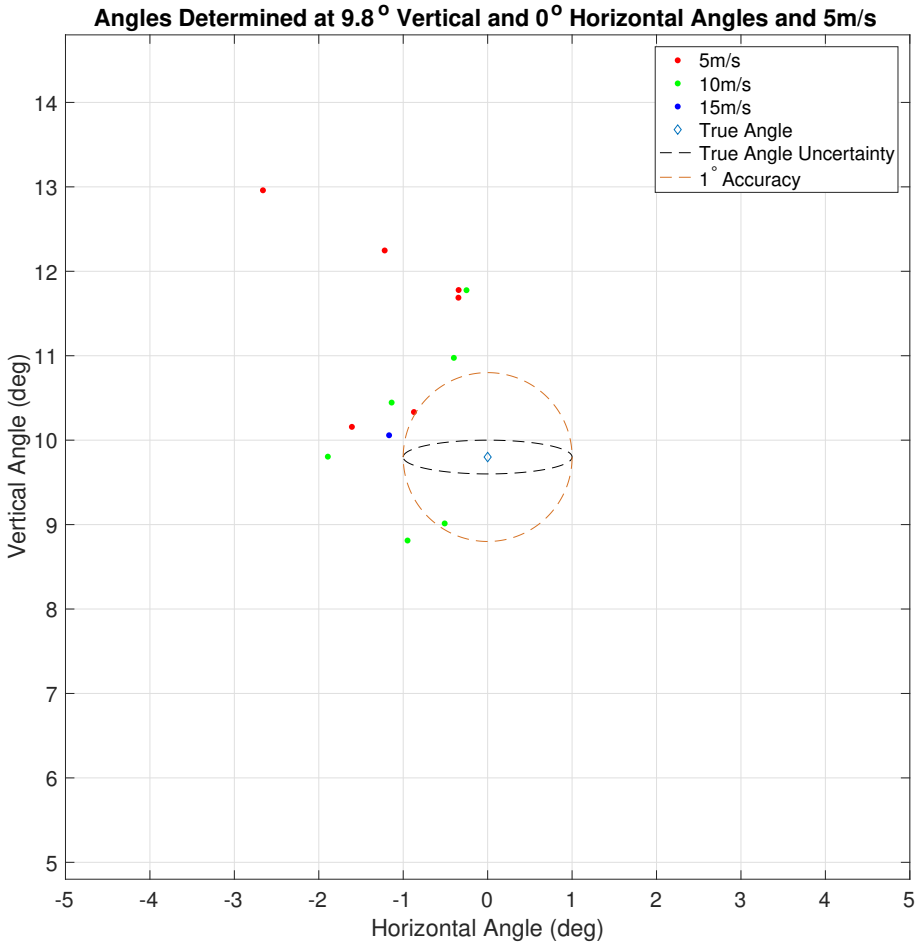


Figure 5.19: Flow angle estimates in 0° side slip and 9.8° angle of attack wind tunnel tests, detailed view.

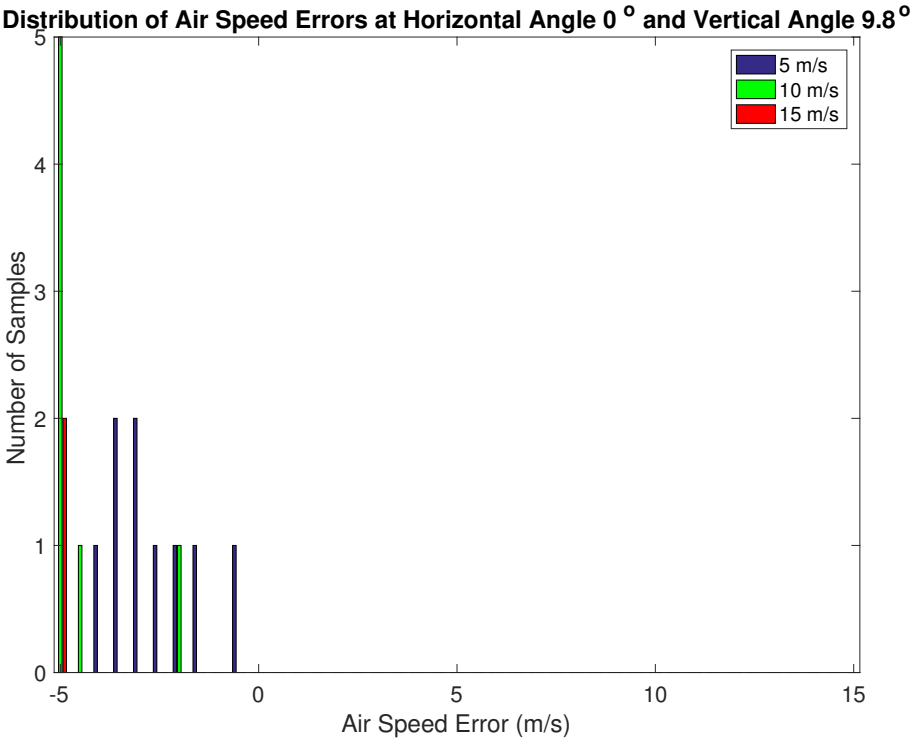


Figure 5.20: Distribution of air speed errors in 0° side slip and 9.8° angle of attack wind tunnel tests.

likely contributed. The image shown in figure 5.17 was one of the higher quality images captured in this orientation and shows poorly defined streak returns.

5.6 0° Side Slip and -5.9° Angle of Attack Flow Results

Figure 5.21 shows an example image of particle streaks taken with the camera pointed at a -5.9° angle of attack and 0° side slip angle. Figure 5.22 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.23 shows a detailed view of the points near the true angles. Points in gray are those measured at other positions. Figure 5.24, shows the distribution of air speed errors from the estimates made in this position.

Generally better streak image quality at this position led to improved angle error, especially seen at the 5m/s speed. While there is an apparent bias of close to 1° the horizontal component is consistent with biases seen in other positions and may result from the inherent inaccuracy of the horizontal angle measurement. The vertical component may be driven by mapping error. Most estimates are clustered within 1° which is close to the prediction from simulation. A larger distribution of estimates is seen in the vertical angle which would be expected fora radial distortion as shown in chapter 4. With smaller angular separation of the flow from the image center, mapping errors would be expected to have a smaller impact. Some bias is seen in the vertical angle which could again result from mapping errors and/or small additional downward velocity due to gravity overcoming fluid-dynamic forces on larger particles.

Air speed performance remains relatively poor as in most of the other cases tested. Once again the large number of overly low estimates is a contrast with the simulation prediction, with estimates often too low by 5m/s or more. Again a mix of method errors, incorrectly identified streak ends and distortion effects likely contributed.

Figure 5.21: Example image of particle streaks in the 0° side slip and −5.9° angle of attack flow case.

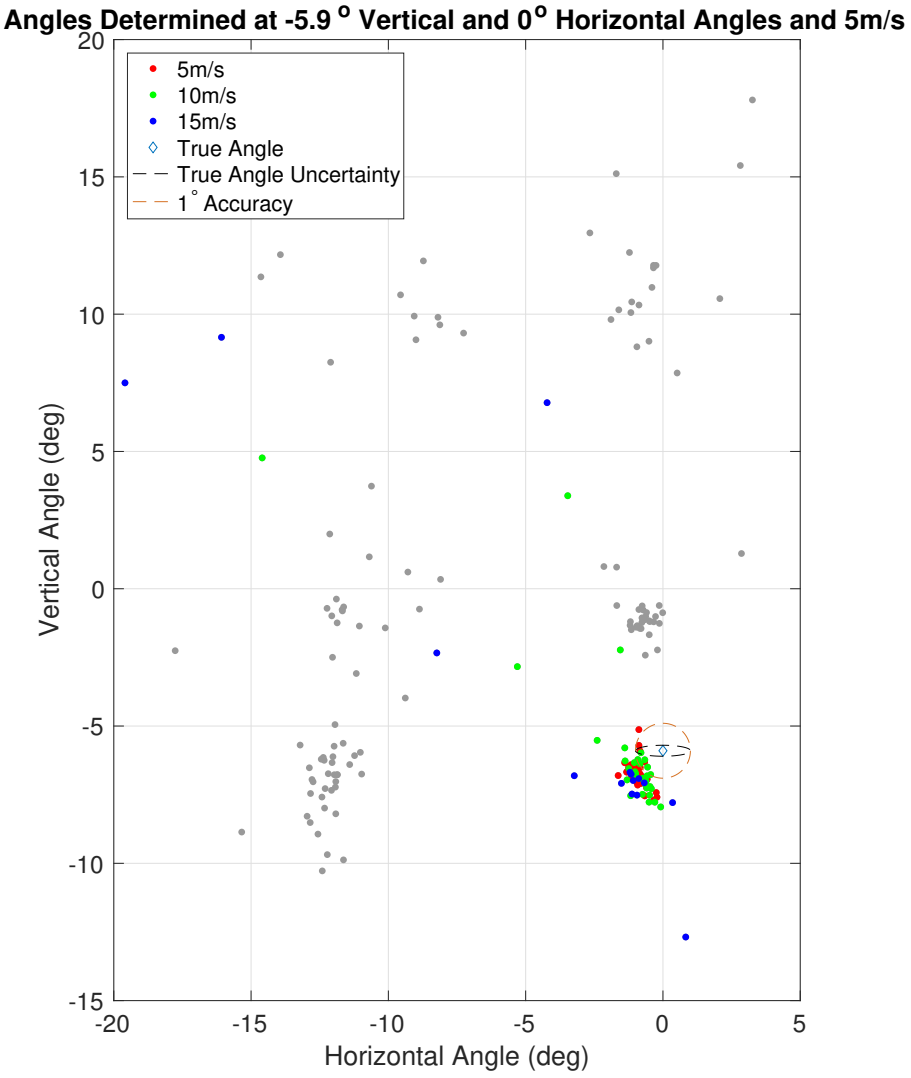


Figure 5.22: Flow angle estimates in 0° side slip and −5.9° angle of attack wind tunnel tests.

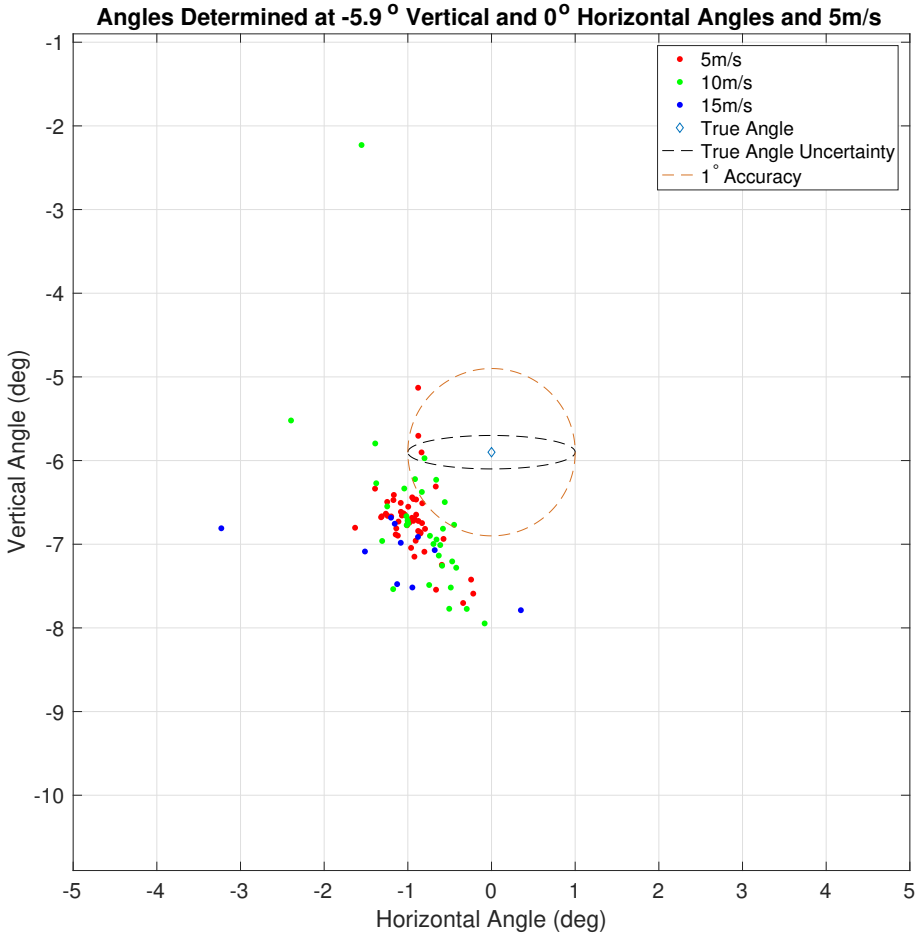


Figure 5.23: Flow angle estimates in 0° side slip and −5.9° angle of attack wind tunnel tests, detailed view.

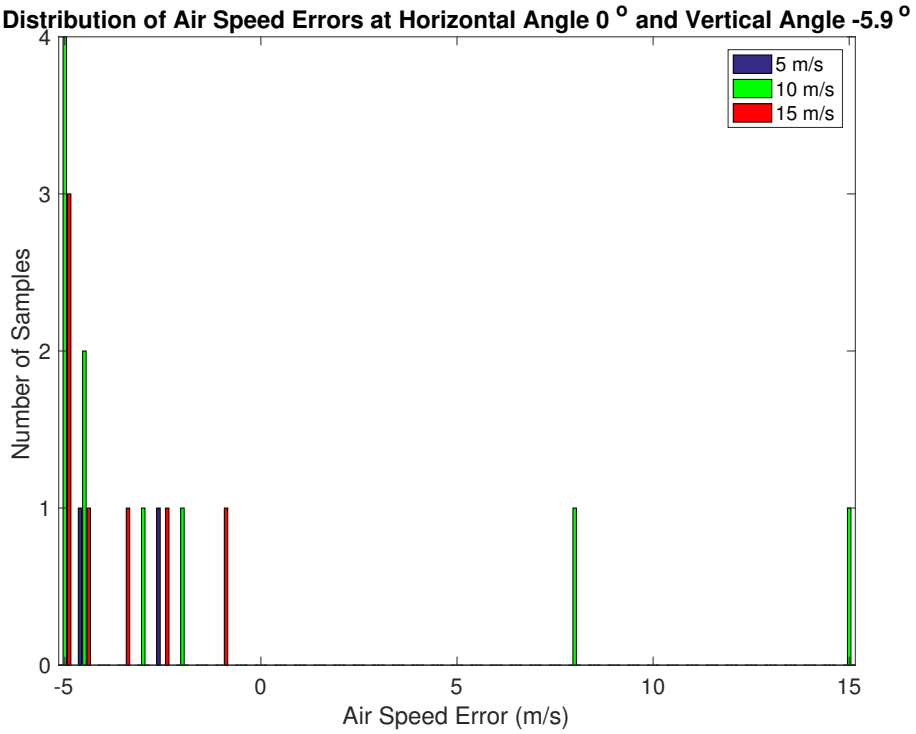


Figure 5.24: Distribution of air speed errors in 0° side slip and -5.9° angle of attack position wind tunnel tests.

5.7 10° Side Slip and −5.9° Angle of Attack Flow Results

Figure 5.25 shows an example image of particle streaks taken with the camera pointed at a −5.9° angle of attack and 10° side slip angle. Figure 5.26 shows the distribution of angle estimates with the camera aligned at this orientation at the three air speeds tested. Figure 5.27 shows a detailed view of the points near the true angles. Points in gray are those measured at other positions. Figure 5.28, shows the distribution of air speed errors from the estimates made in this position.

Angle accuracy is poorer in this case than the −5.9° angle of attack and 0° side slip angle case. This is again indicative of the mapping not matching the pinhole model. Interestingly a bias of around 2° is seen in the horizontal angle but most of the data cluster within±1°, while in the vertical angle there is smaller apparent bias, perhaps 1°, but a wider random distribution. The distribution in vertical angles is larger than predicted from simulation which could be some result of the details of the mapping error, or could result from gravitational effects adding vertical velocity components to some particles, and depending on the particle set captured, this could have a smaller or larger effect.

Air speed errors remain very large, an issue seen in most of the cases measured in these tests. Again the errors are largely negative bias which is contradicts the simulation results that indicated positive bias. Again very large errors of several m/s are seen. The 10m/s case is especially poor, with most of the results too low by more than 5m/s. A possible reason for the generally low air speeds seen throughout testing is that the streaks are being detected as too short because the way the ends of the streaks appear are different than the way they are seen in simulation and the method used does not capture the full streak length as a result.

5.8 Validation Testing Result Summary

The number of samples gathered in this test effort was relatively low due to issues with the test setup used. The limited samples taken showed relatively good angular estimation

Figure 5.25: Example image of particle streaks in the 10° side slip and −5.9° angle of attack flow case.

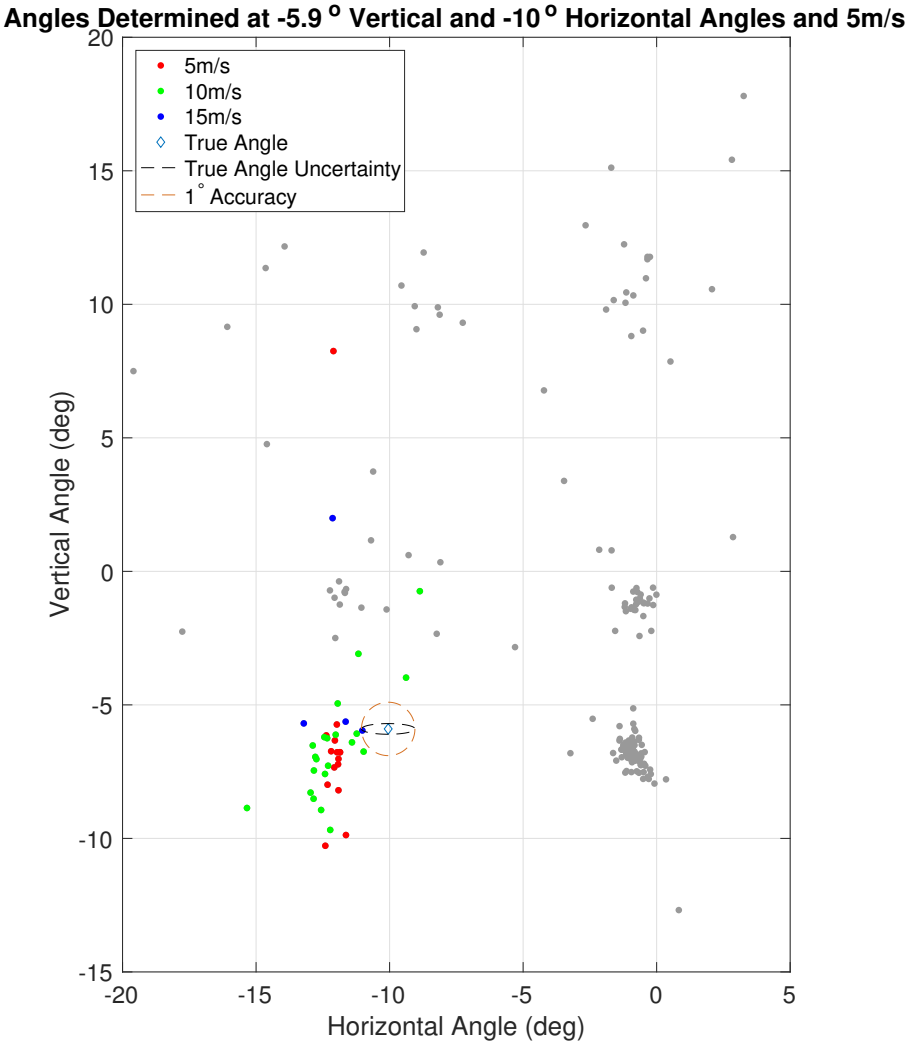


Figure 5.26: Flow angle errors in 10° side slip and −5.9° angle of attack wind tunnel tests.

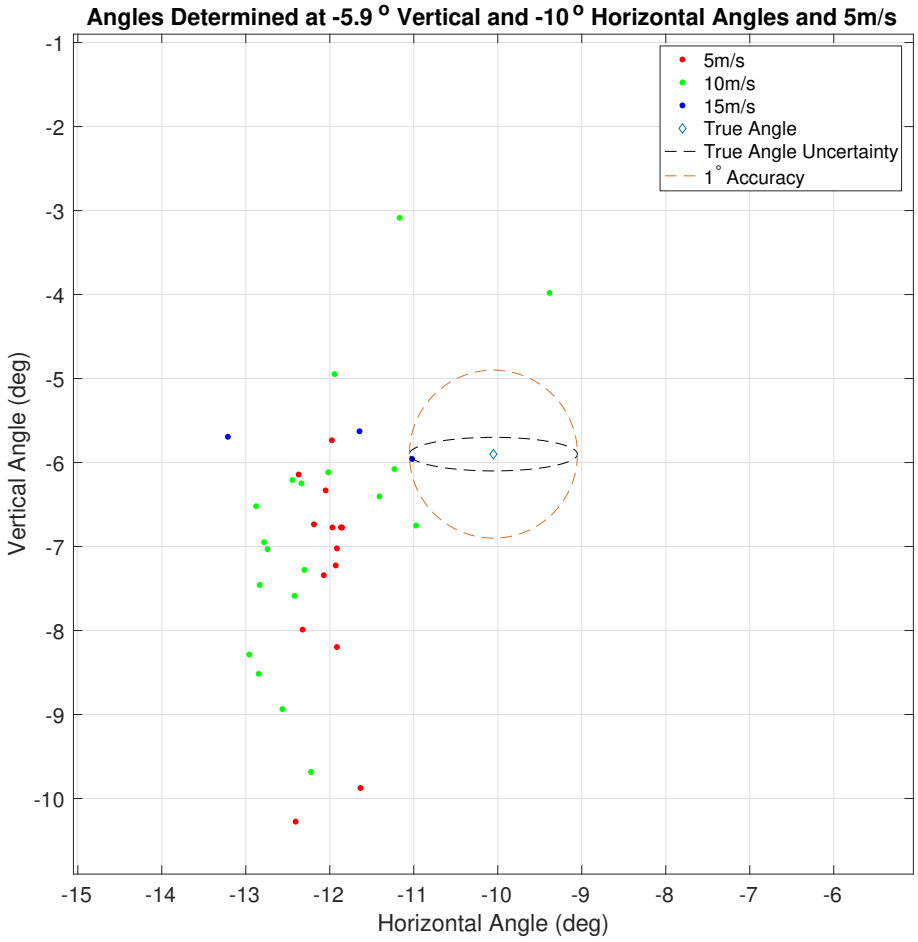


Figure 5.27: Flow angle errors in 10° side slip and −5.9° angle of attack wind tunnel tests, detailed view.

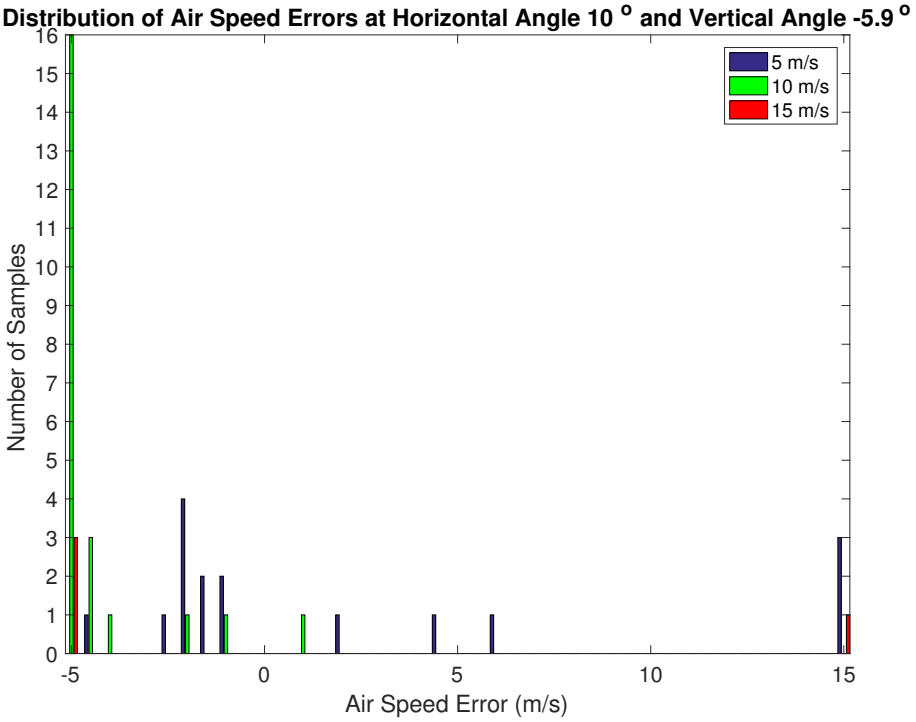


Figure 5.28: Distribution of air speed errors in 0° side slip and −5.9° angle of attack position wind tunnel tests.

performance in the centered flow case, where radial distortion mapping errors have a minimal impact (since the errors are symmetric about the image center). Performance was similar to predictions from simulation with small numbers of streaks detected. Sub-degree flow angle accuracy seen in the limited samples collected is competitive with, or superior to, existing systems. Off-center cases showed wider errors, these were generally similar or poorer than existing systems. These were unsurprising given that the true lens mapping was not known and the method relies on a pinhole model. Extending the method to improve performance in these cases is discussed in chapter 6. Test conditions with bright features or weak particle returns (possibly due to scattering conditions), generally led to poorer performance. Conclusive test results would require an improved test approach with better seeding equipment to allow for larger numbers of samples to be captured. Mapping errors must also be addressed by using a more accurate camera mapping model in future tests.

Air speed errors were large but in cases with good streak returns (a limited subset of tested cases due to the relatively crude test arrangement), air speed errors were no worse, and in some cases better, than the predicted performance in air speed. In many cases however, where streak returns were weaker, air speed estimation was extremely poor even when angle error performance was relatively good. While air speed performance was not the primary focus of this work, improvements to the approach need to be a focus for future work. Some potential changes are discussed in chapter 6.

Chapter 6

Conclusions and Future Work

6.1 Conclusions

The PSA method has been shown capable of estimating three-dimensional local flow velocity from particle streak images with limited information in imaged volumes of arbitrary depth in difficult imaging conditions. The method provides relatively high directional accuracy compared to existing systems, in some cases exceeding their abilities. This is critical for determining vertical and lateral flow components from fixed-wing-sUAS. Sub-degree accuracy capability has been shown in simulation and preliminary test results indicate the potential for similar performance in practice. Simulation data indicates that for large numbers of detected streaks, accuracy below 0.1° in flow angles may be possible. In practice though, detecting such large numbers of streaks may be very difficult. The method could be used to develop a sensor for determining relative flow velocity for fixed-wing aircraft operating in suitable particle environments, such as those potentially encountered by sUAS operating in the lower troposphere.

This work showed the potential for air speed measurements for illumination-limited imaging geometries that could achieve competitive flow accuracy to existing systems, with errors in the $1-2$m/s range; however, in focus-limited cases, airspeed accuracy was relatively poor in simulation, with errors of 3m/s or more, due to the uncertain limits of the image volume. This compares poorly to existing systems. In physical testing air speed errors were also generally large, with only the centered 5m/s case providing estimates that could roughly

match existing systems. Further work is needed to improve air speed sensing accuracy.

Simulated results were backed by preliminary wind tunnel results further indicating the potential for high angular accuracy, though due to focus limiting of imaging, air speed accuracy was again poor. The method shows potential for use in simple flow sensing applications but several extensions and improvements are possible. These are discussed in the next section.

6.2 Future Work

The work presented here demonstrated the potential capabilities and limitations of the proposed method. However a large number of extensions are possible. Significant further work is also needed to create feasible systems for use as aircraft air data systems on sUAS (a driving motivation of this work). In this chapter a number of areas for potential future developments are identified.

6.2.1 Computational Performance

A full assessment of computational performance of the method is needed. Some preliminary work was done to keep computational loads reasonable. For example edge detection was done using integer operations, and the separate flow direction and air speed determination stages allowed the direction to be determined with an unconstrained linear approach and the airspeed estimation to be maintained as a single degree of freedom problem. However, more work is needed to assess the areas of greatest load and estimate the total loads for typical estimates. Such information could then be used to help optimize computation to allow for higher performance.

A true assessment of computational complexity is complicated by the use of RANSAC, which results in a variable number of solutions being calculated, and because the method was implemented in MATLAB with many built-in functions which vary the methods they use on the fly, resulting in differing loads depending on the circumstance. In testing though,

edge detection was generally the most expensive step and is fortunately an area where vast improvement is possible as built in Canny edge detection in MATLAB is faster than the modified method used here and the modified method is simpler in principle, meaning the modified method should be as fast or faster if efficiently implemented.

6.2.2 Performance Improvements

Performance limitations were observed in validation testing and simulation results. Angular accuracy was good in centered cases but degraded at larger angles in wind tunnel testing. This is thought to be largely a result of the true lens mapping not matching the pinhole model used in the method as described in chapter 3. The direction vectors in equation are derived from this pinhole model, but determining them with a more accurate mapping calibrated to the camera system in use does not change the ability to solve the direction problem.

6.2.2.1 Air Speed Estimation

While the primary goal of this work was accurate flow direction sensing, an air speed solution was attempted. However, in practice this method showed generally mediocre to poor performance relative to existing sUAS flow sensing systems. A number of alternative approaches could likely be investigated. A superior approach could potentially be derived by using an assumed range scale based off the known subject distance and the widths of detected streaks (by scaling the narrowest, in focus, parts of streaks to be at the subject distance).

Under this approach the air speed estimation step would not be necessary as the scale factor for $\vec{V}_s$ would be chosen by assuming the range to the first endpoint, probably as the range to the center of the imaged volume. The air speed error under this approach would be determined by the x distance from the center of the imaged volume to the true position of the first endpoint. Thus, the accuracy of the method will improve as the x thickness of

the imaged volume decreases relative to the x distance to the center of the imaged volume.

A similar method would be to assume the ranges to the starting points of all streaks, for example that they lie near the more distant side of the imaged volume from the candle. Then the assumed final endpoints are all determined by $\vec{V}\Delta t$. This would reduce the number of unknown quantities to 3, the elements of $\vec{V}$, which would decrease computational load. There would be no need to compute the scaled $\vec{V}_s$. The accuracy of such an approach however would necessarily rely on same ratio of the x thickness of the imaged volume to the x distance to the center of the imaged volume noted in the previous proposed method.

6.2.3 Extension To Arbitrary Volumes

The method as described here relies on imaging depth limits that are constant laterally across the image. However this may not be optimal for certain set ups and vehicle integration strategies. While not necessary when using focus-limiting, if using illumination-limiting, it will be necessary to allow for depth limits that vary across the image. Thus one potential area for future work is to extend the solution method to deal with arbitrary illumination geometries.

In order to achieve this, it is necessary to allow for a direction dependent depth in the air speed solution. This would allow for placing each line end point in the illumination range based on the limits of that range along the line that passes from the camera through that point. To do this, the illumination limits will need to be functionally defined in terms of the direction vectors to the end points. Determining an efficient method for accomplishing this and ascertaining the effects on system performance are suitable areas for future research.

6.2.4 Optimization of Imaging Systems

This work focused on using captured particle streak images to determine a local flow. The details of the optical system were modeled and accounted for, but no effort was made to optimize the system (in large part because the available optical systems for testing were lim-

ited). As noted in section 3.3, there are a large number of design trade-offs in imaging system parameter selection. A promising area for further study is to determine the optimal set of camera features taking into account size, weight, power, cost, and computing requirements associated with changing the various parameters.

Alternative camera systems could also be designed for capturing the smaller, far more numerous particles present in the atmosphere. This would allow for the proposed method to be used for sensing air data in a wider variety of environments.

One element of improving the imaging system for use in the outdoor environment is developing closely matched illumination and imaging systems. In the outdoor environment, the sun provides powerful broadband illumination, creating overpowering background signal. To mitigate this, a narrow band illumination approach could be used. By band limiting the illumination and filtering the camera signals the amount of background signal can be cut while still providing a large amount of particle illumination. This method could potentially be further improved upon by using a polarized filter to limit light to the expected polarization angle of light scattered from the particles.

One possible method for maximizing background suppression with this approach is to match the illumination and imaging wavelengths to atmospheric absorption bands. These are light wavelengths where the sun's intensity is reduced by absorption by atmospheric gases [30]. An illustration of these bands is shown in figure 6.2. However, due to re-radiation from the Earth's surface, selecting a wavelength for optimal background suppression is likely to be more complicated.

Further complexity in selecting imaging wavelength results from the properties of the imaging sensors themselves. The sensitivity of digital image sensors (CCD and CMOS), the quantum efficiency, varies with wavelength. The quantum efficiency curves for several example sensors are shown in figure 6.1. As such optimizing the imaging system is a complex problem requiring a road study of imaging systems and extensive testing. Ultimately the results of such work could be used to design a purpose-built camera and illumination system

for capturing the images needed for this work.

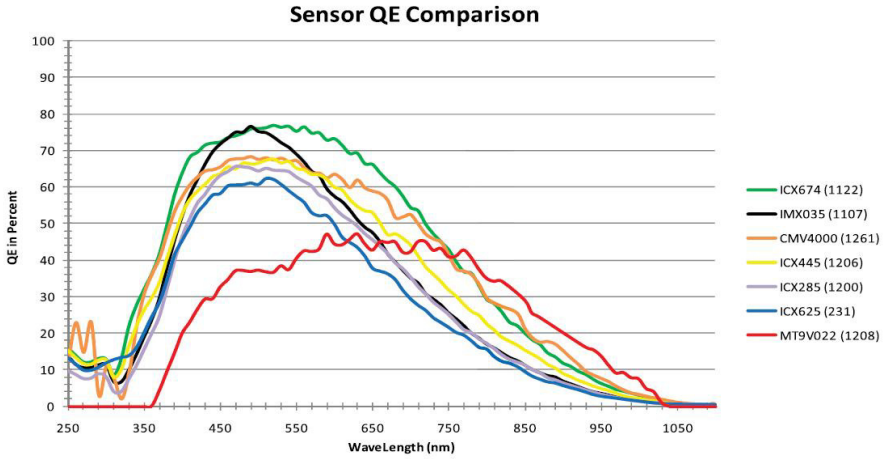


Figure 6.1: Quantum efficiencies of several digital camera sensors. *Image credit: Point Grey Research, Inc.*

6.2.5 Resolving Directional Ambiguity

While the targeted application of this method is initially fixed wing UAS an obvious benefit exists if such a method could be used on a rotary wing aircraft. The standoff capability of the measurement allows a potential for measuring air-relative velocity from multi-rotor UAS without interference from the strong local flow fields. However, in the direction solution for this method there is a sign ambiguity. Assuming a forward velocity is a feasible solution for fixed wing vehicles, but not for rotary wing vehicles which can hover. Determining reliable methods for resolving this directional ambiguity is another area of potential future work.

To resolve this ambiguity a number of approaches could be investigated. One would be to use a separate sensor or vehicle state information (for example vehicle tilt) to provide a rough flow direction and use this to resolve the ambiguity and allow for a solution. Another option would be to use two or more images in rapid succession. Streaks could then be tracked from one image to the next to determine which end of a streak is the beginning and which is

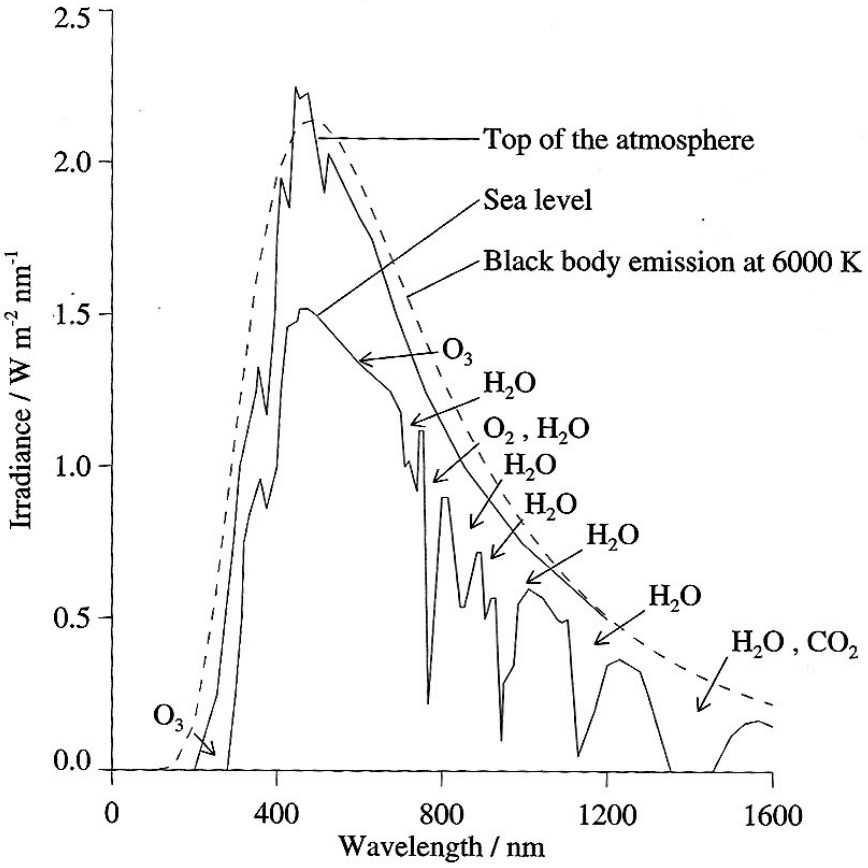


Figure 6.2: Spectrum of sunlight intensity on Earth's surface [30].

the end. This would also allow for a method to determine what streaks remain in the volume for the entire image period and which enter or leave. This could be done by taking a three image sequence and then using only the streaks from the middle image that can be linked to streaks in the first and last images. Yet another approach could be to use a frequency coded light source to give streak start and end points different colors, though this would be difficult to work into a narrow band illumination with matched image filtering strategy.

Bibliography

[1] Adobe Support. Correct image distortion and noise in Photoshop. https://helpx.adobe.com/photoshop/using/correcting-image-distortion-noise.html.

[2] R J Adrian. Particle-image techniques for experimental fluid mechanics. Annual Review of Fluid Mechanics, 23:261–304, 1991.

[3] G. B. Airy. On the Diffraction of an Object-glass with Circular Aperture. Transactions of the Cambridge Philosophical Society, 5:283, 1835.

[4] Brian Argrow and Adam Houston. UAS for in situ sensing of an atmospheric airmass boundary. AIAA InfoTech at Aerospace Conference, 1(May):453–460, 2007.

[5] E. Asmi, V. Kondratyev, D. Brus, T. Laurila, H. Lihavainen, J. Backman, V. Vakkari, M. Aurela, J. Hatakka, Y. Viisanen, T. Uttal, V. Ivakhov, and A. Makshtas. Aerosol size distribution seasonal characteristics measured in Tiksi, Russian Arctic. Atmospheric Chemistry and Physics, 16(3):1271–1287, 2016.

[6] DC Brown. Decentering Distortion of Lenses. Photometric Engineering, 32(3):444–462, 1966.

[7] John Canny. A computational approach to edge detection. IEEE Transactions on pattern analysis and machine intelligence, (6):679–698, 1986.

[8] Anjan Chakrabarty and Jack W. Langelaan. Energy-Based Long-Range Path Planning for Soaring-Capable Unmanned Aerial Vehicles. Journal of Guidance, Control, and Dynamics, 34(4):1002–1015, 2011.

[9] F Charrière, E Cuche, P Marquet, and C Depeursinge. Biological cell (pollen grain) refractive index tomography with digital holographic microscopy - art. no. 609008. Three-Dimensional and Multidimensional Microscopy: Image Acquisition and Processing Xiii, 6090:9008, 2006.

[10] W Y Chen. Energy Dissipation Rates of Free Atmospheric Turbulence. Journal of Atmospheric Sciences, 31:2222–2224, 1974.

[11] A. D. Clarke, Y. Shinozuka, V. N. Kapustin, S. Howell, B. Huebert, S. Doherty, T. Anderson, D. Covert, J. Anderson, X. Hua, K. G. Moore, C. McNaughton, G. Carmichael, and R. Weber. Size distributions and mixtures of dust and black carbon aerosol in Asian outflow: Physiochemistry and optical properties. Journal of Geophysical Research D: Atmospheres, 109(15):1–20, 2004.

[12] Jeff Conrad. An introduction to depth of field. Large Format Photography, pages 1–9, 2006.

[13] Aeroprobe Corporation. Aeroprobe Micro Air Data System User Manual. 1(Revision D):1–38, 2016.

[14] Liselle Douyon and David E Schteingart. Lorenz-Mie Scattering. Matrix, 31(1):1–19, 2002.

[15] Dantec Dynamics. Flexpiv made challenging piv measurement successful, 2017.

[16] Dantec Dynamics. Measurement principles of lda, 2017.

[17] Jerry Eaves and Edward Reedy. Principles of modern radar. Springer Science & Business Media, 2012.

[18] Jack Elston, Brian Argrow, Eric Frew, Adam Houston, and Jerry Straka. Evaluation of unmanned aircraft systems for severe storm sampling using hardware-in-the-loop simulations. Journal of Aerospace Computing, Information, and Communication, 8(9):269–294, 2011.

[19] Jack Elston, Brian Argrow, Maciej Stachura, Doug Weibel, Dale Lawrence, and David Pope. Overview of Small Fixed-Wing Unmanned Aircraft for Meteorological Sampling. Journal of Atmospheric and Oceanic Technology, 32(1):97–115, 2015.

[20] Jack S. Elston, Jason Roadman, and Maciej Stachura. The Tempest Unmanned Aircraft System for In Situ Observations of Tornadic Supercells: Design and VORTEX2 Flight Results. Journal of Field Robotics, 28(4):461–483, 2011.

[21] Sijie Fu, Pascal Henry Biwole, and Christian Mathis. Particle Tracking Velocimetry for indoor airflow field: A review. Building and Environment, 87:34–44, 2015.

[22] C. J. Grund, R. M. Banta, J. L. George, J. N. Howell, M. J. Post, R. A. Richer, and A. M. Weickmann. High-resolution doppler lidar for boundary layer and cloud research. Journal of Atmospheric and Oceanic Technology, 18(3):376–393, 2001.

[23] M Hardesty, S Tucker, and A Brewer. Lecture notes in computer assisted diagnosis, April 2011.

[24] Derek Hoiem. Projective geometry , camera models and calibration, 2011.

[25] Adam L. Houston, Roger J. Laurence, Tevis W. Nichols, Sean Waugh, Brian Argrow, and Conrad L. Ziegler. Intercomparison of unmanned aircraftborne and mobile mesonet atmospheric sensors. Journal of Atmospheric and Oceanic Technology, 33(8):1569–1582, 2016.

[26] Colorado Correctional Industries. Merv Rating Chart. pages 0–1.

[27] M Ivey, J Verlinde, R Petty, R Ellingson, and D Desilets. Polar Research with Unmanned Aircraft and Tethered Balloons. (July), 2013.

[28] Bernd Jahne. Digital image processing, volume 4. Springer, 2005.

[29] S Kameyama, T Ando, K Asaka, Y Hirano, and S Wadaka. Compact all-fiber pulsed coherent Doppler lidar system for wind sensing. Applied optics, 46(11):1953–1962, 2007.

[30] Laboratory for Atmospheric and Space Physics. Class 5- Spectroscopy of Atmospheres.

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

[32] Prasun Mahanti. Micro-particle Streak Velocimetry - Theory, Simulation Methods and Applications. PhD thesis, Arizona State University, 2011.

[33] S. Martin, J. Bange, and F. Beyrich. Meteorological profiling of the lower troposphere using the research UAV "m2AV Carolo". Atmospheric Measurement Techniques, 4(4):705–716, 2011.

[34] MathWorks. bwconncomp. `https://www.mathworks.com/help/images/ref/bwconncomp.html`.

[35] Christian Mätzler. MATLAB Functions for Mie Scattering and Absorption. IAP Research Report. (No. 2002-11), 2002.

[36] Stephanie Mayer, Anne Sandvik, Marius O. Jonassen, and Joachim Reuder. Atmospheric profiling with the UAS SUMO: A new perspective for the evaluation of fine-scale atmospheric models. Meteorology and Atmospheric Physics, 116(1-2):15–26, 2012.

[37] NCAR UCAR Earth Observing Laboratory. Calibration Laboratory Resources.

[38] Tevis W. Nichols, Brian Argrow, and Derek B. Kingston. Error Sensitivity Analysis of Small UAS Wind-Sensing Systems. AIAA Information Systems-AIAA Infotech @ Aerospace, (January):1–15, 2017.

[39] David Nistér. Preemptive RANSAC for live structure and motion estimation. Machine Vision and Applications, 16(5):321–329, 2005.

[40] MIT OpenCourseWare. Isotropic homogeneous 3D turbulence. page 14, 2007.

[41] N T Ouellette, H T Xu, and E Bodenschatz. A quantitative study of three-dimensional Lagrangian particle tracking algorithms. Experiments in Fluids, 40(2):301–313, 2006.

[42] M K Owen, D S Ensor, and L E Sparks. Airborne particle sizes amd sources found in indoor air. Atmospheric Environment, 26A(12):2149–2162, 1992.

[43] C Perrino. Atmospheric Particulate Matter. Biophysics and Bioengineering Letters, (March):35–43, 2010.

[44] CM Peterson, HJ Paulus, and GH Foley. The number-size distribution of atmospheric particles during temperature inversions. Journal of the Air Pollution Control Association, 19(10):795–801, 1969.

[45] David Pope, Dale Lawrence, and Brian Argrow. Optical Flow Techniques for Wind-Velocity Sensing on a Small Unmanned Aircraft System. (January):1–13, 2015.

[46] a. K. Prasad. Stereoscopic particle image velocimetry. Experiments in Fluids, 29(2):103–116, 2000.

[47] Joachim Reuder, Markus Ablinger, Hálfdán Ágústsson, Pascal Brisset, Sveinn Brynjólfsson, Markus Garhammer, Tómas Jóhannesson, Marius O. Jonassen, Rafael Kühnel, Stephan Lämmlein, Tor de Lange, Christian Lindenberg, Sylvie Malardel, Stephanie Mayer, Martin Müller, Haraldur Ólafsson, Ólafur Rögnvaldsson, Wolfgang Schäper, Thomas Spengler, Günther Zängl, and Joseph Egger. FLOHOF 2007: An overview of the mesoscale meteorological field campaign at Hofsjökull, Central Iceland. Meteorology and Atmospheric Physics, 116(1-2):1–13, 2012.

[48] Joachim Reuder, Pascal Brisset, Marius Jonassen, Martin Müller, and Stephanie Mayer. The Small Unmanned Meteorological Observer SUMO: A new tool for atmospheric boundary layer research. Meteorologische Zeitschrift, 18(2):141–147, 2009.

[49] Joachim Reuder, Marius O. Jonassen, and Haraldur Ólafsson. The Small Unmanned Meteorological Observer SUMO: Recent developments and applications of a micro-UAS for atmospheric boundary layer research. Acta Geophysica, 60(5):1454–1473, 2012.

[50] Duda Ro and Hart Pe. Pattern classification and scene analysis. 1973.

[51] Shapiro and Stockman. Binary Image Analysis. Computer Vision, pages 1–10, 2000.

[52] Wenceslao E. Shaw-Cortez and Eric Frew. Efficient Trajectory Development for Small Unmanned Aircraft Dynamic Soaring Applications. Journal of Guidance, Control, and Dynamics, 38(3):519–523, 2015.

[53] Irwin Sobel. An isotropic 3 3 image gradient operator. 02 2014.

[54] Optical Air Data Systems. Wind Sceptor. Technical report.

[55] Cameron Tropea and Alexander L Yarin. Springer handbook of experimental fluid mechanics, volume 1. Springer Science & Business Media, 2007.

[56] Aline van den Kroonenberg, Tim Martin, Marco Buschmann, Jens Bange, and Peter Vörsmann. Measuring the wind vector using the autonomous mini aerial vehicle M2 AV. Journal of Atmospheric and Oceanic Technology, 25(11):1969–1982, 2008.

[57] Luke Wallace, Arko Lucieer, Christopher Watson, and Darren Turner. Development of a UAV-LiDAR system with application to forest inventory. Remote Sensing, 4(6):1519–1543, 2012.

[58] C.P. Wang. Laser Doppler Velocimetry. J. Quant. Spectrsc. Radint. Transfer, 40(3):309–319, 1988.

[59] Svante Wold, Kim Esbensen, and Paul Geladi. Principal Component Analysis. Chemometrics and Intelligent Laboratory Systems, 2(1-3):37–52, 1987.

[60] Kevin C. Zhou, Brendan K. Huang, Ute A. Gamm, Vineet Bhandari, Mustafa K. Khokha, and Michael A. Choma. Particle streak velocimetry-optical coherence tomography: a novel method for multidimensional imaging of microscale fluid flows. Biomedical Optics Express, 7(4):1590, 2016.