DEVELOPMENT OF HYBRID PARTICLE TRACKING ALGORITHMS AND THEIR APPLICATIONS IN AIRFLOW MEASUREMENT WITHIN AN AIRCRAFT CABIN MOCK-UP

BY

WEI YAN

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Agricultural Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2010

Urbana, Illinois

Doctoral Committee:

Professor Yuanhui Zhang, Chair
Professor Richard S. Gates
Associate Professor Xinlei Wang
Professor Anthony M. Jacobi
Professor John G. Georgiadis

ABSTRACT

Obtaining reliable experimental airflow data within an indoor environment is a challenging task and critical in studying and solving indoor air quality problems. The Hybrid Particle Tracking Velocimetry (HPTV) system is aimed at fulfilling this need. It was developed based on existing Particle Tracking Velocimety (PTV) and Volumetric Particle Tracking Velocimetry (VPTV) techniques. The HPTV system requires three charge-coupled device (CCD) cameras to view the illuminated flow field and capture the trajectories of the seeded particles. By adopting the hybrid spatial matching and object tracking algorithms, this system can acquire the 3-Dimensional velocity components within a large volume with relatively high spatial and temporal resolution.

Synthetic images were employed to validate the performance of three components of the system: image processing, camera calibration and 3D velocity reconstruction. These three components are also the main error sources. The accuracy of the whole algorithm was analyzed and discussed through a back projection approach. The results showed that the algorithms performed effectively and accurately. The reconstructed 3D trajectories and streaks agreed well with the simulated streamline of the particles.

As an overall testing and application of the system, HPTV was applied to measure the airflow pattern within a full-scale, five-row section of a Boeing 767-300 aircraft cabin mockup. A complete experimental procedure was developed and strictly followed throughout the experiment. Both global flow field at the whole cabin scale and the local flow field at the breathing zone of one passenger were studied. Each test case was also simulated numerically using a commercial computational fluid dynamic (CFD) package. Through comparison between the results from the numerical simulation and the experimental measurement, the potential model validation capability of the system was demonstrated. Possible reasons explaining the difference between experimental measurement and model prediction were discussed.

To my parents, wife and daughter.

ACKNOWLEGEMENTS

I wish to thank my advisor, Dr. Yuanhui Zhang, for his intellectual support, constant inspiration, understanding, and friendship during my Ph.D. study. Without him, this research and dissertation would not be possible.

I would like to thank Dr. Richard S. Gates, Dr. Xinlei Wang, Dr. Anthony M. Jacobi, and Dr. John G. Georgiadis for serving on my committee. Their comments and advice are invaluable and have significantly improved the quality of the research.

I would like to thank everyone in the Bioenvironmental Engineering Division for their assistance and stimulating discussions, especially Dr. Yigang Sun, Mr. Steve Ford, and Ms. Jingwei Su for their help with the experimental system setup; and Dr. Pascal H. Biwole, Dr. Dongning Li and Mr. Doug Barker for their help in conducting the experiment and discussing the algorithms. Many thanks also go to Ms. Mary Beth Munhall for her help in editing this dissertation.

I am grateful to Dr. K.C. Ting, Head of the Department of Agricultural and Biological Engineering, for his leadership and encouragement. It was exciting and an honor to be enrolled in a top Agricultural and Biological Engineering program in the nation.

I also wish to thank all the people who provided me with help and friendship during my time at the University of Illinois at Urbana-Champaign.

Finally, a special thank you goes to my family. I owe my thanks to my wife, Qin Wang, for her constant support, patience and company; and thanks to our parents for taking care of my daughter so that I could focus on my thesis work. Without them, I would never have achieved what I have now.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

1.1 Background and significance

There are more than 1 billion people traveling by air throughout the world and more than 600 million passengers traveling in the U.S. every year (Owe, 1997; Thomas et al., 1999). In large questionnaire studies, complaints about cabin environments were found to be very common. The extremely high people density, the relatively low pressure, and the low humidity make the environment in an aircraft very unique. In commercial aircraft, the typical available airspace per person ranges from 1000 to 2000 liters/person, compared with the 10000-30000 liters/person range in an office building. Smaller enclosed air space per person requires larger ventilation rates to flush out airborne contaminates (Hocking, 1998). However, current ventilation design does not completely prohibit disease transmission. For example, an Alaskan passenger jet was delayed on the ground for three hours, 72% of the passengers and 40% of the crew contracted influenza (Moser et al., 1979). Another reported influenza outbreak occurred on a naval transport flight from Puerto Rico to Key West, Florida (Klontz et al., 1989). A total of 23 out of 77 passengers were infected by a sick passenger. Most recently, Olsen et al. (2003) reported the SARS outbreak on one flight from Hong Kong to Beijing. Sixteen passengers were laboratory-confirmed infected by one symptomatic patient after the three hour travel.

Limited by measurement technology until today, it is difficult to fully understand the disease transport mechanism in an aircraft cabin. Some researchers argue that the risk of exposure is related to the flight time and the distance between source and receptor. They proposed that the exposure risk is high when traveling time is longer than 8 h and the distance is within two rows of the sick passenger (Hocking, 1998). However, this conclusion was not confirmed by the accident report (Mangili and Gendreau, 2005). Leder and Newman (2005) separated two possible respiratory infection transmission routes: (1) by droplet spread and (2) by the airborne route. When a sick passenger is coughing, sneezing or talking, droplets are

generated by the high air velocity passing though the upper respiratory tract. Large droplets (>5 micrometer) containing organisms may impose directly on a susceptible host's conjunctiva or mucosa, or may settle down quickly. This behavior usually results in a so called "direct infection" which happens at very close vicinity (< 1m). Very small droplets or the residual nuclei of droplets will follow the airflow and affect passengers throughout the entire cabin. This process usually results in "indirect infection."

Current ventilation systems mainly focus on the "indirect infection" control. The most straight forward strategy is to increase the ventilation rate to dilute the pollutant concentration. For example, one cabin air exchanges occur in every 4-6 minutes versus 5-12 minutes in offices or homes (Wang, 2006). However, this strategy was criticized due to its high energy cost. To save the energy cost of heating/cooling air, half of the discharged air is re-circulated. Using the re-circulated air is risky from the point of view of disease transmission. Although all airline ventilation systems have installed high efficiency particulate air filters (HEPA) to clean the re-circulated air, their efficiency is not sufficient. The efficiency of HEPA is normally higher than 99.9% for particles larger than 0.3 micron and some are extended to 0.01 micron (Hocking, 1998). It does work for bacteria which has diameters ranging from 0.5-15 micron. However, it is not sufficient to remove viruses. Viruses have a diameter typically between 0.003 and 0.05 micron. That is why ASHRAE concluded that no known filtering technique was effective to control an airborne viruses' infection. The alternative infection control approach is managing the pollutant transport through airflow pattern optimization. For example, in a Boeing 767-300 aircraft cabin, the supply ducts are along the top of the cabin over the aisle way and the exhausts are at floor level of the side walls. Two vortexes are generated at each side of the cabin, while the longitudinal airflow is minimized in such an arrangement. The purpose of this design is to control the disease from spreading in the longitudinal direction.

Recently, some local environment control strategies have been proposed to reduce the "direct infection." A novel seat for the aircraft cabin was developed. It supplies clean outside air directly to the breathing zone of the passengers (Jacobs and De Gids, 2006). Based on the tracer gas experiment they reported that the tracer gas concentration could be significantly reduced if

the personal airflow achieved $3m^3/s$ per seat. At the same time, the passenger comfort would not be significantly affected by adding this ventilation system. Another group (Gao and Niu, 2008) proposed to use an adjustable fresh air supply nozzle, which could be integrated with the seat, to help control the local environment close to the sick passenger. They argued that the nozzle could shield up to 60% of the pollutants in the user's inhalation. However, without reliable experimental data to validate them, their conclusion is doubtable.

To directly observe and quantitatively evaluate the pollutant transport within a cabin is extremely challenging. As far as we know, there is no available technique for this purpose. Currently, the tracer gas approach is one of the most widely used methods to study the contaminant dispersion within an indoor environment (Rydock and Hermansen, 2002). The experiment is usually conducted in a full-scale chamber. Tracer gas is released to mimic the pollutant. The concentration at some fixed locations are sampled and detected. A global picture of the pollutant transport could be recovered through data interpolation (Wang, 2006; Yan et al., 2009; Zhang and Chen, 2007). However, the tracer gas method is an intrusive measurement technique. High dense sampling points will affect the airflow pattern and prohibit the accuracy of the experiment. To overcome this limitation, non-intrusive optical measurement techniques have been developed. These techniques have been used in measuring the airflow pattern within an aircraft cabin. Some successful applications include particle image velocimetry (PIV) (Zhu et al., 2006), stereoscopic particle image velocimetry (SPIV) (Sun et al., 2005), and volumetric particle tracking velocimetry (VPTV) (Zhang et al., 2005). The problem with these systems is that most of them only have 2D tracking capability or their target objects are limited within a plane or a very thin volume. It is urgently necessary to extend these systems to true 3D volume measurements.

Compared with experimental methods, the CFD can provide more detailed information with less expense. Dozens of simulations have been conducted for airflow patterns within an aircraft cabin (Aboosaidi et al., 1991; Lin et al., 2005; Singh et al., 2002; Yan et al., 2009; Zhang

and Chen, 2007). However, these simulated results are either validated using a limited sampling point data, or not validated at all. In addition, previous works only focused on the global field of the cabin. Little analysis was conducted to study the local environment close to the releasing source. Although some literature reported simulation results of an inhalation region in simplified ventilation cases (Gao and Niu, 2004; Murakami, 2004; Zhu et al., 2005), none of them performed simulations within full-scale aircraft cabins.

With the current experimental and simulation status, it would be valuable to develop a non-intrusive 3D measurement system which could detect the airflow field in large scale volume. The experiment would be conducted in both the global and local environment of the cabin. The experimental data would benefit both public health and computational fluid dynamics research communities.

1.2 Objectives

The goal of this study is to develop a hybrid particle tracking algorithm with the capability to measure 3D airflow patterns in large scale volume and then apply this system to measure the global airflow field of a full-scale aircraft cabin mock-up and the local environment close to an individual passenger after his coughing/sneezing. In addition to the experimental study, numerical simulations will also be conducted. Experimental and simulated results will be compared. Specific objectives are to:

(1) Develop a novel hybrid particle tracking velocimetry (HPTV) algorithm based on two existing techniques in our group: volumetric particle tracking velocimetry (VPTV) (Sun et al., 2004) and particle tracking velocimetry (PTV) (Li, 2008). The uniqueness of the HPTV is that it can track the highly dense particles in large volume with a large velocity range at a high sampling rate (about 30Hz).

(2) Establish a measurement procedure for the HPTV system. It will consist of illumination, trace particle seeding, calibration, image acquiring, error analysis, particle tracking, and post-processing for flow field reconstruction.

(3) Measure the airflow pattern and the transport of airborne contaminants in a full-scale aircraft cabin mock-up using the HPTV system. Both the global airflow field produced by the ventilation system and the local airflow field closely related to passenger behavior will be determined.

(4) Conduct numerical simulations with the same configuration as the experimental set-up using commercial CFD software. The experimental and simulated results will be compared.

CHAPTER 2 LITERATURE REVIEW

2.1 Introduction

The air motion in rooms has a relative broad velocity range (Sandberg, 2007). Air velocity of supply devices ranges from 0.5m/s (displacement ventilation) to 10m/s (diffusers for mixing ventilation). In the occupied zone, the typical air velocity is 0.05-0.5m/s. The turbulence intensity in the occupied space is about 25%, and the velocity fluctuation time scale is about 0.5s. The Kolmogrov scale in an occupied zone is about 1 mm.

An ideal instrument for indoor application should have properties such as high sensitivity, high accuracy, large measurement range, whole volume based, high spatial resolution, and non-intrusive to the airflow. The techniques commonly used at present can be classified into two categories: (1) point-measuring techniques and; (2) whole-field measuring techniques. Point-measuring techniques use a probe to record the history of flow over time at one point in space. It is also possible to simultaneously access the parameters at several positions using multi-sensors. By doing this, the discrete whole-field parameters could be obtained. The experimental result is usually represented in an Eulerian frame. The measured parameters are represented as a function of time and space. An alternative technique, whole-field measurement, simultaneously measures the parameters at different locations relying on flow visualization techniques, which were developed during the past two decades. The strategy of this technique is making the fluid movement visible by seeding proper “fluid tracers.” The movement of the tracers is recorded by cameras. The whole velocity field could be recovered by analyzing the physical positions of the tracers at different image frames. The ways to represent the result of whole-field measurement techniques depends on the tracking approach used. If individual tracers are tracked, the result will be in a Lagrangian frame, such as particle tracking velocimetry. If groups of tracers are tracked, such as particle imaging velocimetry, the result will be in an Eulerian frame.

2.2 Review of velocity measurement techniques

2.2.1 Point-measuring techniques

The most traditional point-wise velocity measurement technique is the Piot tube. It is a thin hollow tube, which has a side hole, and is aligned with flow velocity. Pressure can be measured as the moving fluid is brought to rest. This pressure is known as the total pressure. According to Bernoulli's equation, the fluid velocity is related to these total and static pressure differences. The Piot tube is an inexpensive method of metering. The flow meter itself is cheap and it is cheap to install. However, it is difficult to apply it to an indoor air quality study. The first problem is the relatively low spatial and temporal resolution (Kreith and Goswami, 2005; Tavoularis, 2005). The response time of the Pitot tube is usually longer than the turbulent fluctuation time; therefore, only an averaged velocity over the response time could be obtained. In addition, because it is an intrusive point-wise measuring equipment, it is impossible to obtain high spatial resolution without disturbing the flow field. The third problem of applying the Pitot tube in an indoor air flow field is the velocity range. As described in the previous section, the typical air velocity in the occupied zone is 0.05-0.5m/s, which is too small for the Pitot tube to detect. Besides, it is impossible to align the Pitot tube with the air flow direction, because this parameter is usually unavailable before conducting the experiment. In some cases, even the flow itself is unstable, without proper alignment the obtained result is not reliable.

Thermal anemometers is another widely used point-wise measuring equipment. When air passes the electrically heated thermal sensor, for example hot-wire/film, it has a cooling effect on the sensor. The resistance of the sensor usually depends on its temperature, which is related to the heat convection and the air velocity rate around the sensor. Therefore, a relationship can be established between the resistance and the flow velocity. Compared with the Pitot tube, the thermal anemometer device has a much quicker response time. Its continuous output is well suited to digital sampling. It provides time series that can evaluate the statistical parameters for turbulent character studies (Goldstein, 1996). For example, Flynn et al. (Flynn et al., 1995) used

hot-film anemometry to obtain the velocity data for a turbulent push-pull ventilation system and validated their numerical models. The researchers at Athens University (Athens, Greece) (Papadopoulos et al., 1996) developed a triple hot-wire system to study the three components of velocity above a threshold of 0.1m/s for indoor air flow. Currently, the 3D thermal anemometers have been commercialized, for example TSI Model 1299A (TSI, Shoreview, MN.). However, thermal anemometers still face the same challenging problem as the Pitot tube. The spatial resolution is low since it is still an intrusive point-wise measurement device. When applied to measure the low velocity of indoor airflow, their accuracy is usually not satisfying because the free convection generated by the heat sensor is comparable with forced convection in the occupied zone. In other words, the measurement of low-speed flows with thermal anemometers could produce an error of 100%, which is completely unacceptable.

Laser Doppler Anemometry (LDA) or Laser Doppler Velocimetry (LDV) is another point-measuring technique, which was first reported by Yeh and Cummins (1964). The basic principle of the LDV is as follows: The fluid is seeded by small particles. Two beams, which are usually produced by splitting a single beam, are made to intersect to generate a set of straight fringes with known fringe spacing. As particles pass through the fringes, they reflect light into a photo-detector. The velocity could be obtained by multiplying the received signal frequency and the fringe spacing. The accuracy of LDV/LDA could achieve $\pm 0.1\%$ in its measuring range. Its temporal resolution is very high and it is a non-intrusive technique (Menon, 1998). These advantages make LDV/LDA an impressive point-wise measuring equipment, which could fully recover the high frequency turbulent fluctuation. Recently, applying LDV/LDA to an indoor air flow study has been widely accepted. For example, Jiang et al. (2003) conducted the velocity measurement using LDA on a grid plane at the center section of a building model. The probe is positioned using a computer controlled traversing arm which provided a spatial resolution of 0.5mm vertically and 1mm horizontally. Posner et al. (2003) conducted their LDA experiment in a one-tenth sub-scale model room to access the whole airflow structure produced by the ventilation system. A full-scale test room LDA experiment with dimensions of 3.4x2.7x2.4m was

done recently by Richmond et al. (2006) to access the velocity field in a complex furnished room. LDA could also be extended to whole-field measuring, which is called Doppler Global Velocimetry (DGV). Instead of detecting a single particle, DGV detects clouds of particles. Then the Doppler shifted frequency of these particles are transformed to intensity variations in the image plane, which is further processed by computer to obtain the velocity map of the flow (Meyer and Komine, 1991). However, DGV is considered to be more suitable for the measurement of high speed flows (estimated to be above 30m/s) due to the resolution limitation which is approximately equal to 1m/s (Smith, 1998). This is obviously beyond the velocity range of indoor airflow.

An ultrasonic anemometer is another point-wise measuring equipment. It measures the time taken for an ultrasonic pulse travelling from one transducer to the opposite one and also the time of the opposite direction. Differences are measured between pairs of transducers allowing calculations of both air velocity and direction. The spatial resolution is given by the path length between transducers, which is typically 10 to 20cm. Ultrasonic anemometers can take measurements with very fine temporal resolution, 20Hz or better, which make them well suited for turbulence measurements (Knowles and Athelstan, 1953). The application of ultrasonic anemometers for indoor airflow studies has also been reported. Kim et al. (2001) conducted their experiment to measure the 3D velocity distribution of an under-floor air-conditioning room. The flow characteristics are obtained by a 3D ultrasonic anemometer. Recent research has been conducted in an aircraft cabin the size of 4.9x4.32x2.1m (Zhang et al., 2009). Air velocity was measured by two sets of Kaijo Ultrasonic Anemometers (DA-650) (Kaijo Corporation, Japan) with TR-92T probes. According to their report, the path length between transducers is 3cm and the accuracy could achieve 0.005m/s with $\pm$ 1% error.

Table 2-1. Compare point-wise measurement techniques.

Techniques	Advantages	Disadvantages	Recommend for IAQ?
Pitot tube	(1) Inexpensive and easy to install	(1) Slow response (2) Low accuracy for low air velocity (3) Hard to align with airflow (4) Low spatial resolution (5) Intrusive	No
Thermal anemometer	(1) Cheap to buy and install (2) Quick response time	(1) Low accuracy for low air velocity (2) Detect velocity direction will be expensive (3) Low spatial resolution (4) Intrusive method	No
LDA/LDV	(1) Quick response time (2) High accuracy (3) Non-intrusive method	(1) Expensive to buy and install (2) Low spatial resolution	Yes
Ultrasonic anemometer	(1) Quick response time (2) High accuracy	(1) Expensive to buy and install (2) Low spatial resolution (3) Intrusive method	Yes

2.2.2 Whole-field measurement techniques

There is increasing demand to get a large quantity of experimental data at mesh locations distributed in the airflow field. It would also be interesting if the transport history of individual particles could be recorded in a Lagrangian frame. One of the most interesting and possible techniques to achieve this purpose is the airflow visualization technique, which has been widely used to study the velocity field. In this technique, tracers are seeded in the fluid. Therefore, the

spread of fluid could be observed directly by the naked eyes or recorded by cameras through tracking seeds which include smoke, bubbles and fluorescent particles. By quantitatively examining the particle position on the images of corresponding cameras, it is possible to reconstruct the 2D/3D velocity field. The most popular systems belonging to this category include PIV, SPIV, HPIV, and PTV, etc.

PIV was proposed by Pickering et al. (1985) and Adrian et al. (1984). It relies on the loss of phase information of a single image to recover the flow velocity. However, in these works only one image was canalized; therefore, the direction of velocity is ambiguous although the angle and magnitude could be recovered. The modern PIV was developed based on the pioneering work of Adrian. He proposed to reconstruct the velocity field by utilizing the auto-correlation for a double-exposure image, which results in the PIV experimental design rules recommended by Keane and Adrian (1991b). The complete PIV strategy was generalized and described by Keane and Adrian (Adrian, 1991; Keane and Adrian, 1991a; Keane and Adrian, 1993). The basic principle is: A pair of camera frames produced by two separate exposures is recorded. The frames are next split in a large number of interrogation areas, often called tiles. The displacement vector for each tile is calculated with the help of signal processing. It is converted to a velocity using the time between image exposures. The spatial resolution of PIV is determined by the size of the tiles. PIV has recently been used to study the airflow in a generic simplified airplane cabin (Bosbach et al., 2006). In their experiment, both large scale PIV and detail PIV measurements were conducted. The purpose of the large scale PIV is to capture the vortices in the occupied zone of the cabin. The size of interrogation windows is 48x48 pixels, which has the corresponding spatial resolution of 70×70mm. The spatial resolution of large scale PIV has to be large due to the limitation of seeding particles and illumination. Helium filled soap bubbles with diameters between 1 and 3mm are used to trace the airflow in their large scale experiment. In order to study the Coanda effect and the turbulent diffusion of the wall jet, they also conducted a detailed scale PIV. The interrogation window size amounted to 24x24 which corresponds to a spatial resolution of 4x4mm. The seeding particles are small droplets of

Di-Eehyl-Hexyl-Sebacat (DEHS), synthetic oil. A similar large scale experiment was reported by Sun et al. (2007). Their experiment was conducted in a 5.5x3.7x2.4m full-scale ventilation room. The spatial resolution was 100x100mm. PIV applications in a microclimate or in a sub-scale room model were also reported by (Mortensen et al., 2008; Posner et al., 2003). The advantages of applying PIV to an indoor air quality study include: (1) non-intrusive, (2) high temporal resolution, and, (3) whole-field accessing. However, the limitations are also obvious. PIV only uses one camera to reconstruct the velocity field. Therefore, only 2D velocity components are recovered from the images. The laser sheet for PIV is usually very thin to avoid overlapping of particles in the depth direction. The PIV experimental result is represented in an Euler frame, which means it is not easy to track the movement of individual particles.

To reconstruct the 3D velocity components, the stereo-PIV (SPIV) was further developed (Arroyo and Greated, 1991; Grant et al., 1998; Palero et al., 2000), and based on the same principle as the human eyesight. Instead of using one camera, an additional camera was applied in SPIV to view the same object from a second direction. Ideally, the most accurate determination of the out-of-plane displacement (i.e., velocity) could be obtained when there is 90° between the two cameras. However, because of the distortion of the image field, the angle has to be small enough to achieve an acceptable defocusing. Although it is possible to obtain a 3D velocity field using the SPIV system, the depth of field-of-view is still very limited. In recent years, holographic PIV systems (HPIV) have been proposed. Holographic PIV is based on the holographic image of a particle field to reconstruct the 3D displacement of particles over two exposure times for a volumetric domain. Compared with traditional PIV, the complexity and cost of HPIV is extremely high. The capacity of current charge-coupled device (CCD) cameras is still unaffordable for such a demand (Barnhart et al., 2004).

Instead of tracking particle groups, it is also possible to track individual particles or particle streaks at different exposure times (frames). The velocity of airflow could be recovered by dividing particle displacement over time (which can be obtained by frame rate). This approach is called particle tracking velocimetry (PTV) or particle streak velocimetry (PSV). To

our best knowledge, the development of these techniques occurred in the 1980s. A 2D PSV approach was proposed by Adamczyk and Rimai (1988b). Although they named the approach PTV, they actually detected the line-segment length of particles during exposure time. A total of four sequential images were recorded. The duration time of the first exposure was $100\,\mu s$, and the remaining three were $250\,\mu s$. The streak patterns of dot-dash-dash-dash were detected as the trajectory of particles. This approach was further extended to 3D PSV by using two orthogonal cameras to reconstruct the 3D velocity components of particles (Adamczyk and Rimai, 1988a). The two end points of a streak in camera 1 are projected into camera 2 to find the corresponding streak. The two streaks are defined as being matched if the error was smaller than a tolerance. Then the real-world coordinate was constructed based on the camera model. Adrian (1991) has given a comprehensive review on these methods. He defined the PSV as a single-frame technique and the PTV as a multi-frame technique. He concluded that the particle streak could not be truncated or overlapped with each other in order to be detected. Because the light source for PSV is usually not strong enough, the seeding particles are normally larger than PIV; for example, some experiments use helium filled bubbles which have diameters of several hundred micrometers. However, since the particle images of PTV are isolated points, the confusion due to particle overlapping is largely eliminated. The particle could be tracked over a long period. PTV requires a high speed camera. If the maximum acceptable displacement for a tracking algorithm was 1mm and the sampling rate of the camera is 30Hz, then the maximum velocity we can measure is only 30mm/s. A high speed camera could compensate for this problem; however, it will produce large amounts of images within a short time, which will increase the computational load. Applying the PTV technique to study the air flow character has been very common in recent years (Cenedese and Querzoli, 1997; Cowen and Monismith, 1997; Doh et al., 2004; Doh et al., 2000). However, all the work is limited to small scale. To the author's knowledge, large volume measurement, for example, in a full-scale room chamber, has not yet been reported.

2.3 Research status in BEE group

The first generation of the algorithm in the Bioenvironmental Engineering (BEE) group was developed by Zhao (2000), who conducted an experiment in a full-scale testing room (5.5x3.7x2.4m) within a room ventilation simulator. A two-dimensional airflow velocity vector map was obtained at the cross-section in the middle of the room using the PIV technique.

The second generation algorithm is volumetric particle streak-tracking velocimetry (VPSTV) (Sun et al., 2007; Sun and Zhang, 2003; Wang, 2006). It is capable to acquire the 3D fluid velocity in a volume domain. VPSTV places two cameras at different angles to view the illuminated slice. The movements of particles are recorded as straight streaks if the exposure time was properly set. The velocity magnitudes are calculated from a pair of matched streaks in two synchronous images taken by two cameras. The principle of this algorithm is similar to the PSV approach described in a previous section. Sun and Zhang (2003) applied this system to obtain the 3D airflow velocity field within the same ventilation room as Zhao (2000). They reported that the average absolute error was about 0.05m/s, while the largest relative error was below 30%.

The most recent algorithm in the BEE group was developed by Li (2008), which is named volumetric particle tracking velocimetry (VPTV) system. The basic principle is similar to the PTV approach. The experiment was conducted in a relatively small volume. The observation region was the water flow in a section of cylindrical glass tube which has an inner diameter of 23mm and outer diameter of 25.4mm. Three synchronized cameras were placed to capture the images of seeded particles viewed from different angles. Instead of only taking a snap shot, which has been done in PSV, the image sequences of multi-frames were recorded. The core VPTV algorithm consists of two steps: (1) temporal-tracking and (2) spatial-tracking. The purpose of the temporal tracking algorithm is to link the particles in different frames. Each image gives a snap shot of particles. It is crucial to establish the corresponding relationship between frames. VPTV rejected fake trajectories based on several criteria: The displacement of individual particles between two adjacent frames should be less than a maximum value, the acceleration of

a particle should be bounded by some value, and the final trajectories should be smooth (has a lower cost function value). Based on these rules, a temporal-tracking algorithm is proposed. After these 2D trajectories were detected, a stereo triangulation camera model is applied to do the spatial matching and 3D reconstruction. Details of this stereo triangulation model will be introduced in the following chapter.

However, directly applying VPSTV and VPTV algorithms to study the airflow pattern within a full-scale cabin is challenging. The first problem is the poor accuracy of the velocity compound in an out-of-plane direction. The principle of VPSTV is comparable to human eyes, which are normally sensitive to moving objects parallel to the face plane (in-plane direction). It is difficult to localize the objects moving toward the face (out-of-plane direction). The error in the out-of-plane direction could be up to ten times of the error in the in-plane coordinates (Sandberg, 2007). The VPTV system used three cameras to reconstruct the real world coordinate. Better accuracy could be obtained by taking advantage of the additional information from the third camera. However, it suffered a great deal from the illumination system. VPTV relied on the central coordinate of individual particles to do the spatial matching. A particle is regarded to be matched only if it could be observed by all three cameras. The brightness value of pixels is compared with a pre-defined threshold to judge if it belongs to a particle image or not. An even illumination within the target volume is preferred in order to obtain a clear image, which means there will be obvious borderlines between the background and particles. To ensure the image quality, the VPTV system experiment is usually conducted in small volume and uses an argon-ion laser as the light source. Unfortunately, this could not be satisfied in the large volume chamber set-up. The illumination system in the occupied full-scale cabin is very bad. There is a large amount of particles being lost during image processing. When directly applying the VPTV algorithm, less than ten trajectories were detected, which is completely unacceptable.

Applying VPSTV and VPTV to study local environments is also difficult. The velocity range at this region is quite broad. The initial velocity at the mouth/nose could achieve 10m/s, and it decayed very quickly along spatial and temporal domains. In the occupied zone, the

typical air velocity decreases to 0.05-0.5m/s. The velocity range will constrain the choice of exposure time (Δt_e) and frame rate (Δt_f) of VPSTV and VPTV systems. For the VPSTV approach, if we choose a Δt_e to make the maximum acceptable streak length be 50 pixels, which are A particle and streak image at local environment corresponding to velocity of 10m/s, the streak length of particles in the occupied zone will be less than 2.5 pixels (0.5 m/s). The particle images will become dots instead of streaks. For the VPTV approach, if the Δt_f was selected to ensure a detectable particle displacement with 2 pixels (0.5m/s), the particle displacement with velocity of 10m/s will achieve 40 pixels, which is by far beyond the maximum acceptable value (normally 15 pixels).

Motivated by these facts, we realized that developing a hybrid particle tracking algorithm has more advantages. The hybrid idea consists of two components: (1) hybrid object tracking and (2) hybrid spatial matching. The first component means we track both dots and streaks at the same image. When high speed particles move relatively long distances during exposure time, their images are short streaks, otherwise they are dot shapes. Both dots and streaks could be produced on the same image Figure 2-1.

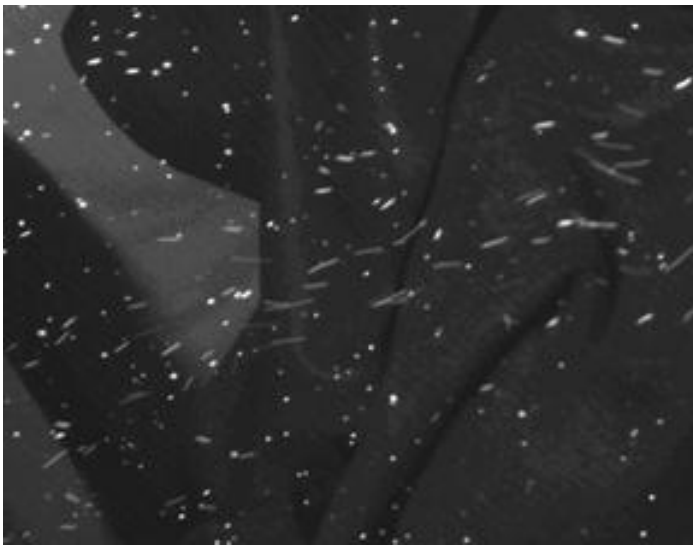


Figure 2-1. A particle and streak image at local environment.

The hybrid particle tracking approach will significantly increase the detectable velocity range. The second hybrid component means that we will reconstruct the real-world coordinates (x, y and z) of a particle, even if it was only observed by two cameras. For individual particles, it is necessary to use at least three cameras to obtain the matched relation in both spatial and temporal domain, which is defined as a three-way matching algorithm. However, by taking advantage of temporal tracking, two cameras are enough to achieve the matching purpose, which is defined as a two-way matching algorithm. Hybrid spatial matching means we will apply both of them. If a particle was observed by three cameras, the three-way matching algorithm will be applied; otherwise, the two-way matching will be used. Details of the matching mechanism will be discussed in the following chapter.

CHAPTER 3 ALGORITHM DEVELOPMENT

The HVPT system is a combination and further development of the previous VPTV and VPSTV techniques. The method can be summarized as follows: Three CCD cameras are installed as perpendicular with each other as possible and three particle image sequences are obtained simultaneously by each camera. Then, an image processing algorithm is applied to detect the location of dot particles and streak particles. At the same time, the length and orientation of the streaks are extracted. The improved algorithms developed based on VPTV and VPSTV are adopted subsequently to process 2D trajectories and 3D reconstruction. Finally, the detected 3D particle trajectories are combined together to obtain the velocity field and do the post-processing. The diagram of the whole procedure is shown in Figure 3-1. The specifics of these steps are presented in the following sections.

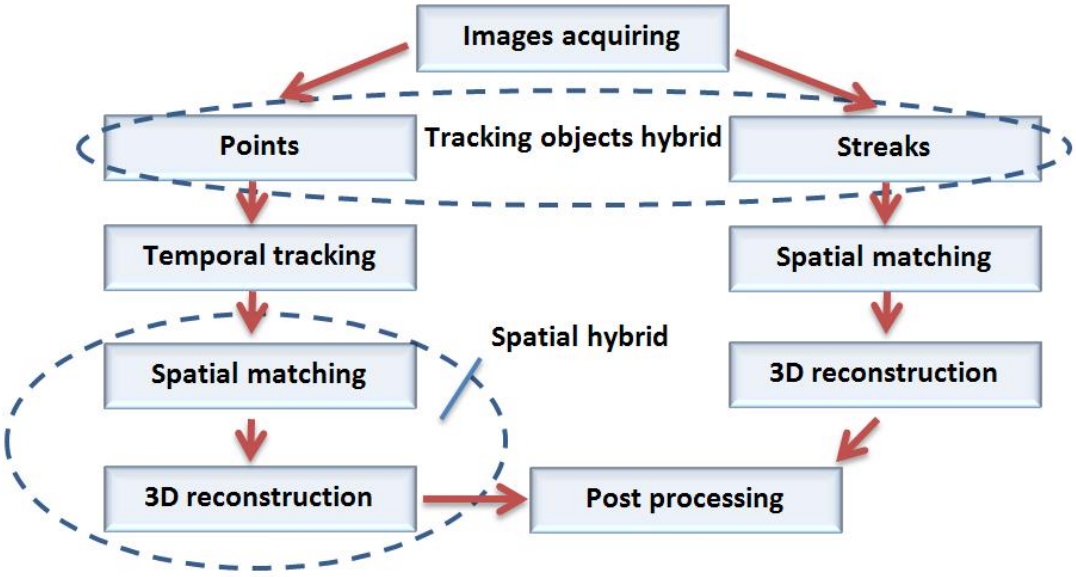


Figure 3-1. Flow diagram of HVPT system.

3.1 Image processing

The purpose of image processing is: (1) to remove the noise produced by electronic devices, lens and system set-up; (2) to detect the location of particles (centers of dots and streaks); and (3) to determine the orientation and length of detected streaks. The camera speed in the current experiment is 30 frames per second, which means 90 images will be captured within one second. It is impossible to count particles manually. A practical algorithm should have the capability to access the image information effectively and automatically. The image processing approach is proposed in the following chart (Figure 3-2):

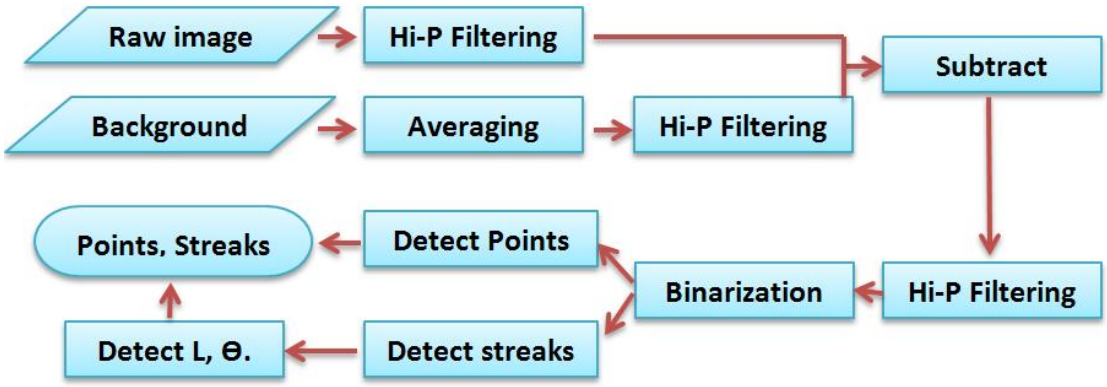


Figure 3-2. Flow diagram of image processing procedure to detect points and streaks.

Before processing the particle images, ten frames of background images are read first. These images are averaged to obtain the background without noise. A small trick here is that the data type of gray image matrix is an unsigned integer, which means the pixel values are from 0 to 255. It is not proper to simply add up all pixel values and divide it by the number of total frames. For example, adding up 210 and 100 will obtain 255 instead of 310. To avoid this problem, the following averaging equation should be used:

$$P_i = \sum_{j=1}^{n} \left(P_{i,j} / n \right) \tag{3.1}$$

Where P_i is the averaged value of the i[th] pixel, $P_{i,j}$ is the value of the i[th] pixel at the j[th] frame, and n is the number of total frames.

Ideally, the particle images should be bright dots or streaks. However, these images could be blurred or weak due to poor illumination. The high pass filter, which is often called the sharpening filter, is used to enhance the particles. It is an empirical process to choose the proper filter for the experiment. After several trials, the following high pass Gaussian filter was chosen:

$$F = \begin{vmatrix} -1 & 0 & -1 & -3 & -4 & -3 & -1 & 0 & -1 \\ 0 & -2 & -7 & -8 & -7 & -8 & -7 & -2 & 0 \\ -1 & -7 & -5 & 3 & 7 & 3 & -5 & -7 & -1 \\ -3 & -8 & 3 & 19 & 27 & 19 & 3 & -8 & -3 \\ -4 & -7 & 7 & 27 & 42 & 27 & 7 & -7 & -4 \\ -3 & -8 & 3 & 19 & 27 & 19 & 3 & -8 & -3 \\ -1 & -7 & -5 & 3 & 7 & 3 & -5 & -7 & -1 \\ 0 & -2 & -7 & -8 & -7 & -8 & -7 & -2 & 0 \\ -1 & 0 & -1 & -3 & -4 & -3 & -1 & 0 & -1 \end{vmatrix} \quad (3.2)$$

The high pass filter is applied to both particle images and the averaged background image. After that, the filtered background is removed from each particle image by simply subtracting. The resulting image is further enhanced by the high pass filter. As we know, high pass filtering also increases the intensity of noise. However, the noise is usually an isolated single pixel after binarilization, which could be removed easily by region property.

The enhanced particle images are binarized through a threshold operation. Pixel values are set to be 1 if they are above the threshold value, otherwise set at 0. Figure 3-3 compares the images before and after the whole image processing: filtering, subtraction and bineralization, etc. It is observed that large amounts of weak point particles, which are not obvious in the raw image, are recovered by the procedure.

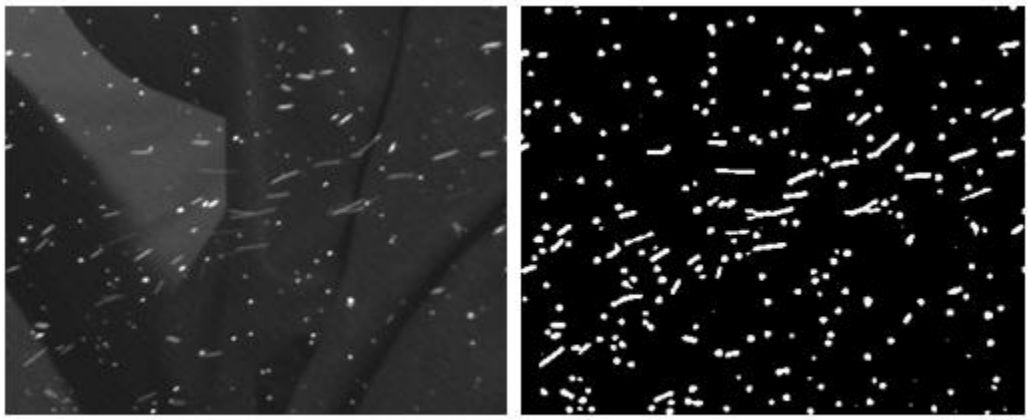


Figure 3-3. Compared images before and after the hi-pass filtering and binarization.

The binary images only contain "0" or "1." The points and streaks are detected by analyzing the region property of the pixels' cluster. As illustrated in Figure 3-4(a), noise usually only contains fewer pixels. An area threshold is pre-defined to remove the noise. If the pixels cluster is smaller than this value, it will be removed. A label operation is conducted next to detect the center of mass of the region which is defined as the location of particle. The streaks are marked if the region area is larger than a streak threshold. We also need to access their orientation and length. As shown in Figure 3-4(b), an ellipse which best fit the pixel cluster is found. By examining the eccentricity (the ratio of the distance between the foci of the ellipse and its major axis length), major axis length, and orientation (the angle between the x-axis and the major axis of the ellipse that has the same second-moments as the region), we could distinguish dots and streaks. The algorithm is proposed as follows:

Procedure Detect particle points and streaks

Loop each region in image
Access: Area, Center, Eccentricity, L (major axis length), orientation;
If Area <= Area.threshold
Break;
Else
If Eccentricity <= E.threshold
Save Point: Center;
Else
Save Streak: Center, L, orientation;
End
End
End Loop

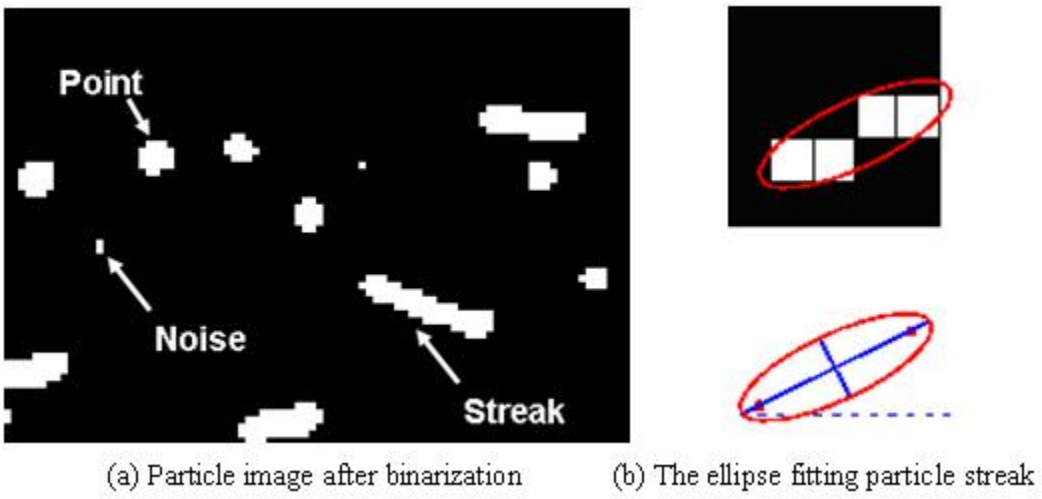


Figure 3-4. Particle detection on binary image.

In the current algorithm, it is assumed that all streaks are represented by straight lines. Streaks are projected trajectories of seeding particles during the exposure time. Due to the high velocity and the turbulence effect, the streaks could be long curves instead of straight lines. Reducing the exposure time will shorten the streaks and straighten the curves. But it will result in a poor illumination condition at the same time. For airflow within an indoor environment, as discussed in Chapter 2, the velocity is usually below 10m/s and the target volume is relatively

large. According to the experiment, a straight line is usually a good approximation. Figure 3-5 shows the result of the image processing procedure described above. The size of the image is 1392x1040. Only a small portion is shown here to give a clear view of the recognized points and streaks. The algorithm successfully detected the particle locations and the length and orientation of the streaks. The information of dots and streaks are recorded separately and waiting for further tracking, matching and reconstruction process. The data file saved the center coordinates (x, y) of the particles, the length (L), and the orientation (θ) of the streaks.

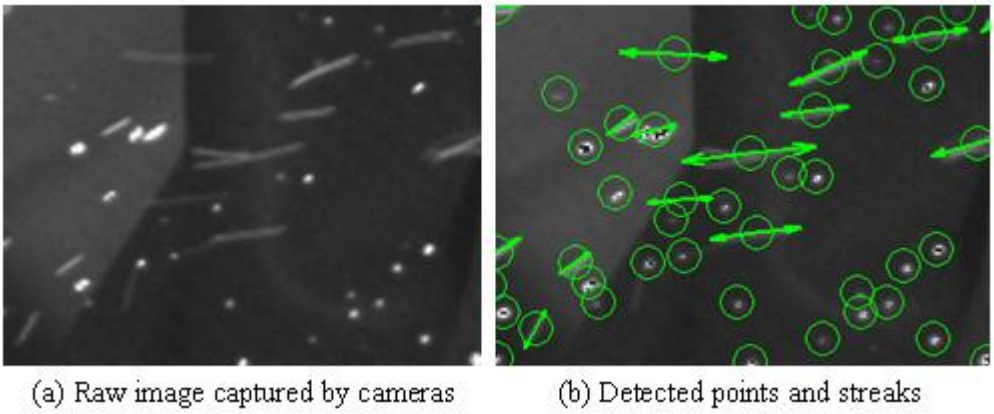


Figure 3-5. Comparison of the raw image and the image with detected particles.

3.2 Dot particle tracking algorithm

As described in Figure 3-1, the detected dots and streaks will be processed separately. A developed procedure based on the VPTV algorithm will process the dots part. This procedure includes three steps: (1) 2D temporal tracking, (2) hybrid spatial matching, and (3) 3D reconstruction.

3.2.1. Temporal tracking algorithm

The first essential part of the algorithm is to link the identical particles from one image frame to another, which is defined as temporal tracking. The traditional temporal tracking algorithm is based on the Lagrangian tracking of individual particles over a long period. It could be performed in 3D space (real world coordinate) or 2D space (image coordinate). Malik et al.

(1993) proposed a prototype of a 3D particle tracking algorithm as shown in Figure 3-6. The scheme operates on a sequence of time steps with at least four frames. Label the current frame as f. The next frame is $f+1$, and so on. Let the particle position be noted as $\mathbf{x}_i^f$, which indicates the physical coordinate of the i[th] particle in frame f. Three searching neighborhoods (r_0, r_1, r_2) are pre-estimated based on the expected velocity fluctuation. Searching neighborhoods are defined as the radius for each frame, centered at the estimated particle locations. Assume that f frame has been tracked. The first step is to estimate the expected location $\mathbf{e}_i^{f+1}$ of the current particle $\mathbf{x}_i^f$ in frame $f+1$. A linear approximation without acceleration is applied:

$$\mathbf{e}_i^{f+1} = \mathbf{x}_i^f + u_i^f \Delta t \tag{3.3}$$

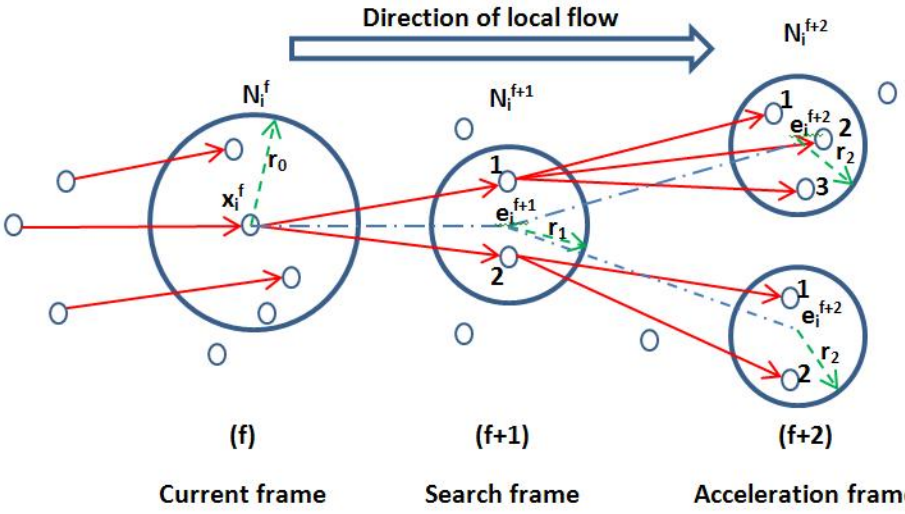


Figure 3-6. Schematic of 3D temporal particle tracking (Malik et al., 1993).

u_i^f is the location velocity estimated from the particles within neighborhood (r_0). Suppose n_i^f particles within (r_0) are linked to a particle in frame $f-1$, then u_i^f is estimated as follows:

$$u_i^f \approx \frac{1}{n_i^f} \sum_{j=1}^{n_i^f} u_j^f \tag{3.4}$$

The next step is searching all the particles within neighborhood (r_1), and constituting the set of possible links from (f,i) to $f+1$. Repeat the previous steps. A tree shape of trajectory

candidates is obtained. The fake links are rejected based on the "sorting principle," which assumed that the time step Δt is small enough that the particles undergo uniformly accelerated motion. Based on this criterion, they obtained the real trajectories.

We further developed this technique and proposed a novel temporal tracking procedure (Li et al, 2008). The uniqueness of our algorithm is that it performs well in cases where particle position error, false detection and missing particles are significant. The whole tracking process is performed in 2D image frame. We use a high order instead of linear extrapolation method to predict the future particle position. Therefore, the involved frames in regression are increased. Up to five previous particle positions are used to perform the polynomial regression in contrast with only two or three positions in the traditional method. A minimum trajectory length threshold is pre-defined to reject "false link detection." The trajectories with a frame number less than this threshold will be discarded. The missing particles are treated by a technique called "irregular tracking." The concept behind the "irregular particle tracking" can be explained from a sampling time interval point-of-view. Regular tracking algorithms track particle motion at consecutive time-frames. A traceable trajectory must appear on each time-frame (as a particle-image) from the beginning to the end of the trajectory. If no matched particle-image/particle is found in one frame, the tracking procedure will stop. However, the "irregular tracking" algorithm will continue to move on to the next frame and try to interpolate at one point. Because of this strategy, the irregular tracking algorithm can obtain longer trajectories than a traditional method. A series of criteria is validated after the tracking proceeds. A detailed description and discussion of this algorithm has recently been published (Li et al., 2008). The result of the temporal tracking algorithm is 2D trajectory image and data files. A typical example is shown in Figure 3-7 and Table 3-1. In Table 3-1, all numbers are in the image coordinate system; (x,y) is the pixel coordinate of the detected particles. For each experimental case, three trajectory tables are recorded. They correspond to the three cameras.

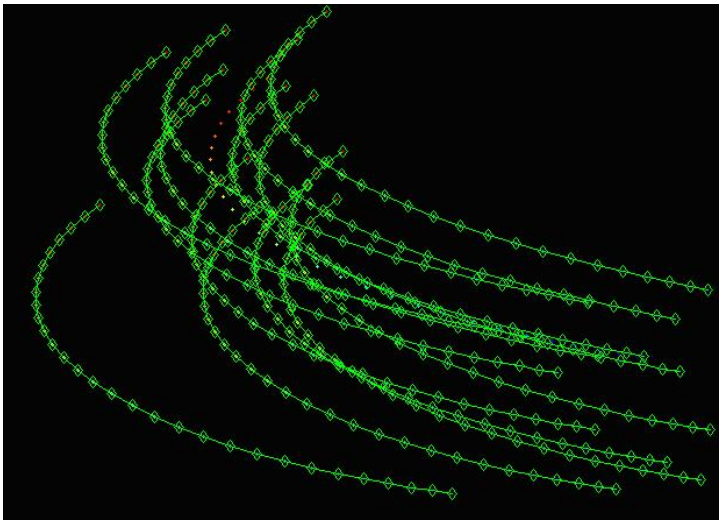


Figure 3-7. Detected 2D trajectories images (it has been trimmed to fit the window).

Table 3-1. Detected 2D trajectories table.

	Frame 1		Frame 2		Frame j	
	x	*y*	*x*	*y*	*x*	*y*
Traj 1	754	435	737	432	……	……
Traj 2	829	302	812	299	……	……
Traj 3	845	331	829	329	……	……
Traj i	871	271	855	268	……	……
……	……	……	……	……	……	……

3.2.2. Hybrid dot spatial matching

The purpose of spatial matching is to find the identical 2D particle trajectories taken by various cameras based on the camera models. In other words, given a trajectory in one camera, we want to find its corresponding trajectories in the other two cameras. It is worth mentioning that these relationships are not necessary to be one-to-one. It is highly possible they would be one-to-many, many-to-one or many-to-many due to broken trajectories.

3.2.2.1 Camera model

Camera model is the mathematical relationship between image coordinate and physical world coordinate. Figure 3-8 illustrates the principle of camera projection. Camera model

describes the projection as two steps: (1) transform from world coordinate to camera coordinate and (2) transform from camera coordinate to image coordinate. These steps are usually controlled by two groups of parameters: extrinsic and intrinsic. The extrinsic parameters are the parameters that define the location and orientation of the camera reference frame with respect to the world reference frame. The intrinsic parameters are the parameters to link the pixel coordinate of an image point with the corresponding coordinates in the camera reference frame (Trucco and Verri, 1998).

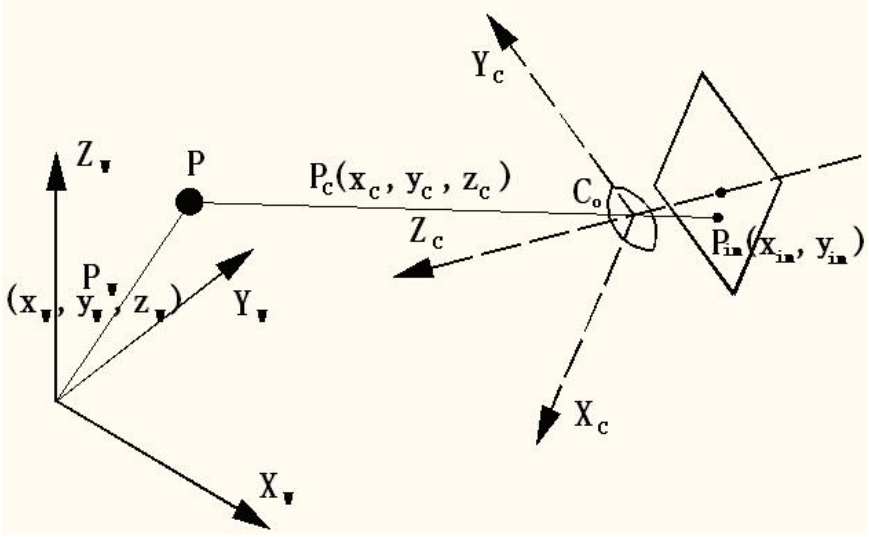


Figure 3-8. Geometrical principle of camera projection.

Extrinsic parameters:

As shown in Figure 3-8, extrinsic parameters uniquely identify the transformation between world coordinate (X_W, Y_W, Z_W) and camera coordinate (X_c, Y_c, Z_c). A typical choice for these parameters are a 3D translation vector **T** and a 3D rotation matrix **R**. For example, the coordinate of particle **P** in world and camera coordinate systems are $\mathbf{P_W}$ and $\mathbf{P_C}$, respectively. The relation between them could be written as:

$$\mathbf{P}_C = \mathbf{R}\mathbf{P}_W + \mathbf{T} \tag{3.5}$$

Or in matrix form, we have:

$$\begin{aligned} x_C &= r_{11}x_W + r_{12}y_W + r_{13}z_W + T_x \\ y_C &= r_{21}x_W + r_{22}y_W + r_{23}z_W + T_y \\ z_C &= r_{31}x_W + r_{32}y_W + r_{33}z_W + T_z \end{aligned} \tag{3.6}$$

Intrinsic parameters:

Intrinsic parameters defined the relationship between the camera coordinate (X_c, Y_c, Z_c) and the image coordinate (x_{im}, y_{im}). Note that the unit of (x_{im}, y_{im}) is pixel. The CCD camera array is made up of a rectangular grid of photosensitive elements. Assume that (s_x, s_y) (in millimeters) is the effective size of the pixel in horizontal and vertical direction, and (o_x, o_y) is the image center in pixel coordinate. Then the coordinate in millimeters and in pixels are related as follows:

$$\begin{aligned} x &= -(x_{im} - o_x)s_x \\ y &= -(y_{im} - o_y)s_y \end{aligned} \tag{3.7}$$

Where (x , y) is the image coordinate in millimeters. As indicated in Figure 3-8, it is a scaled value from camera coordinate by lens. Assuming that the focal length of the lens is f, we have the following scaling relation:

$$\frac{f}{z_c} = \frac{x_d}{x_c} = \frac{y_d}{y_c} \tag{3.8}$$

Without considering the lens distortion, (x , y) is equivalent to (x_d, y_d). Substitute Equation 3.8 into Equation 3.7, we obtain:

$$\begin{aligned} x_{im} &= -\frac{f}{s_x}\frac{x_c}{z_c} + o_x \\ y_{im} &= -\frac{f}{s_y}\frac{y_c}{z_c} + o_y \end{aligned} \tag{3.9}$$

There are five parameters in Equation 3.8 and Equation 3.9: f , s_x , s_y , o_x and o_y .Only four of them are independent variables. In practice, a set of four parameters are defined as intrinsic parameters (without considering the distortion):

$$f_x = \frac{f}{s_x}, \quad \alpha = \frac{s_x}{s_y}, \quad o_x \text{ and } o_y \tag{3.10}$$

In some cases, for example, when fish eye lens are used or we want to study the object at the periphery of the image, it is necessary to consider the distortion introduced by the lens. This distortion is modeled by a radial distortion correction and tangential distortion equation (Brown, 1971):

$$\begin{aligned} x &= x_d \underbrace{(1 + k_1 r^2 + k_2 r^4)}_{\text{radial distortion}} + \underbrace{[2k_3 x_d y_d + k_4 (r^2 + 2x_d^2)]}_{\text{tangential distortion}} \\ y &= y_d \underbrace{(1 + k_1 r^2 + k_2 r^4)}_{\text{radial distortion}} + \underbrace{[k_3 (r^2 + 2y^2) + 2k_4 x_d y_d]}_{\text{tangential distortion}} \end{aligned} \tag{3.11}$$

Where (x_d, y_d) is the image coordinates in millimeter without considering the distortion, and $r^2 = x_d^2 + y_d^2$. This equation introduced four additional intrinsic parameters k_1, k_2, k_3 and k_4. This distortion could be discarded if the peripheral pixels were not important, or if low distortion lens were used. Sometimes k_2 could be ignored if it is much smaller than k_1.

Projection model:

Assuming we have obtained both the extrinsic and intrinsic parameters, a pipe line from physical coordinate to image coordinate could be established. The image projection procedure is described in the following steps:

$$\begin{array}{ccccccccc} \mathbf{P}_W & \xrightarrow{eq3.6} & \mathbf{P}_C & \xrightarrow{eq3.8} & \mathbf{P}_d & \xrightarrow{eq3.11} & \mathbf{P} & \xrightarrow{eq3.7} & \mathbf{P}_{im} \\ (x_w, y_w, z_w) & & (x_c, y_c, z_c) & & (x_d, y_d) & & (x, y) & & (x_{im}, y_{im}) \end{array} \tag{3.12}$$

This procedure is a non-linear transformation because of Equation 3.11. If we neglect the distortion, a direct relationship between $\mathbf{P}_W$ and $\mathbf{P}_{im}$ can be established as:

$$\begin{aligned} x_{im} &= -f_x \frac{r_{11} x_w + r_{12} y_w + r_{13} z_w + T_x}{r_{31} x_w + r_{32} y_w + r_{33} z_w + T_z} + o_x \\ y_{im} &= -\frac{f_x}{\alpha} \frac{r_{21} x_w + r_{22} y_w + r_{23} z_w + T_y}{r_{31} x_w + r_{32} y_w + r_{33} z_w + T_z} + o_x \end{aligned} \tag{3.13}$$

Inverse projection problem:

As shown in Equation 3.13, image projection is pretty straight forward and uniquely determined. However, the reverse problem is not well posed. $\mathbf{P}_W$ has three degrees of freedom, but $\mathbf{P}_{im}$ only has two degrees of freedom. It is impossible to recover $\mathbf{P}_C$ or $\mathbf{P}_W$ solely by $\mathbf{P}_{im}$. The basic principle is similar to the human eyes. It is not possible for humans to identify the 3D location with only one eye. In the same way, at least two cameras are needed to identify the 3D location of $\mathbf{P}_W$. Combining Equation 3.6 and Equation 3.8, we define the following equations:

$$\begin{aligned} x_{norm} &= \frac{x_d}{f} = \frac{x_c}{z_c} = \frac{r_{11}x_w + r_{12}y_w + r_{13}z_w + T_x}{r_{31}x_w + r_{32}y_w + r_{33}z_w + T_z} \\ y_{norm} &= \frac{y_d}{f} = \frac{y_c}{z_c} = \frac{r_{21}x_w + r_{22}y_w + r_{23}z_w + T_y}{r_{31}x_w + r_{32}y_w + r_{33}z_w + T_z} \end{aligned} \tag{3.14}$$

Where x_{norm} and y_{norm} are defined as the normalized camera coordinate of the objects. Reversing Equation 3.14, we obtained the following linear system:

$$\begin{aligned} (x_{norm}r_{31} - r_{11})x_w + (x_{norm}r_{32} - r_{12})y_w + (x_{norm}r_{33} - r_{13})z_w &= T_x - x_{norm}T_z \\ (y_{norm}r_{31} - r_{21})x_w + (y_{norm}r_{32} - r_{22})y_w + (y_{norm}r_{33} - r_{23})z_w &= T_y - y_{norm}T_z \end{aligned} \tag{3.15}$$

If the same model is applied to multi-cameras, we will have at least four equations with three unknowns. Such a system can be solved by the least square method.

3.2.2.2 Epipolar geometry

In order to establish Equation 3.15, we still face some problems. Assume two or more dot objects are captured by two cameras at the same time. We haven't figured out the mapping relationship between dots in one camera and those in the other camera. This problem is called spatial matching.

Figure 3-9 shows the basic principle of epipolar geometry. Assume that the left and right cameras are observing the object P at the same time. O_l and O_r are the lens centers. Image 1 and image 2 are the film planes. Vector $\mathbf{P}_l$ and $\mathbf{P}_r$ are positions of P referred to the left and right

camera frames respectively. Vector $\mathbf{p}_l$ and $\mathbf{p}_r$ refer to the projection of P in the film planes. $\mathbf{P}_l$ and $\mathbf{P}_r$ are related by a rotation matrix **R** and a translation vector $\mathbf{T} = (O_r - O_l)$.

$$\mathbf{P}_r = \mathbf{R}(\mathbf{P}_l - \mathbf{T}) \tag{3.16}$$

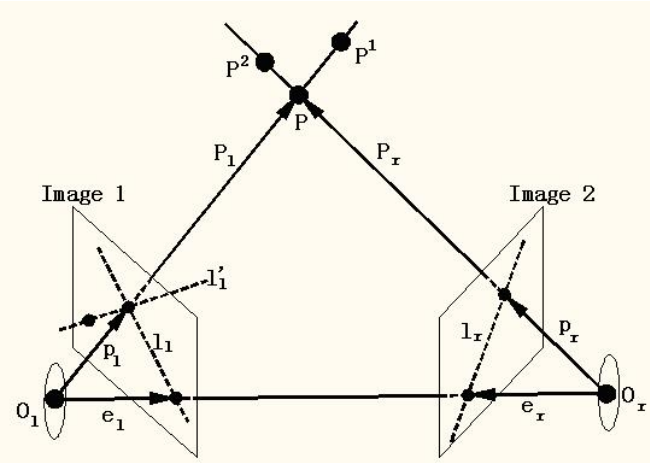


Figure 3-9. Principle of epipolar geometry.

The line O_lO_r is intersected with image 1 and image 2 at two points which are called epipoles. They are also the image of the projection centers. We denote the vectors of epipoles as $\mathbf{e}_l$ and $\mathbf{e}_r$, respectively. As shown in Figure 3-9, all objects on plane $O_rP\,O_l$ will be projected on line l_l and l_r. The plane identified by P, O_l and O_r is called the epipolar plane. From the coplanar requirement, we have:

$$(\mathbf{P}_l - \mathbf{T})^T \mathbf{T} \times \mathbf{P}_l = 0 \tag{3.17}$$

Substitute Equation 3.16 into Equation 3.17 and we obtain:

$$(\mathbf{R}^T \mathbf{P}_r)^T \mathbf{T} \times \mathbf{P}_l = (\mathbf{P}_r{}^T \mathbf{R}) \mathbf{T} \times \mathbf{P}_l = 0 \tag{3.18}$$

Define

$$\mathbf{T} \times \mathbf{P}_l = \begin{bmatrix} 0 & -T_z & T_y \\ T_z & 0 & -T_x \\ -T_y & T_x & 0 \end{bmatrix} \mathbf{P}_l \equiv \mathbf{S}\mathbf{P}_l \tag{3.19}$$

Substituting Equation 3.19 into Equation 3.18, we have the following requirement for coplanar:

$$(\mathbf{P}_r^T\mathbf{R})\mathbf{S}\mathbf{P}_l = \mathbf{P}_r^T(\mathbf{R}\mathbf{S})\mathbf{P}_l \Rightarrow \mathbf{P}_r^T\mathbf{E}\mathbf{P}_l = 0 \quad (3.20)$$

E is called the essential matrix. In a practical problem, we usually use $\mathbf{p}_l$ and $\mathbf{p}_r$ instead of $\mathbf{P}_l$ and $\mathbf{P}_r$. Notice that $\mathbf{p}_l$ and $\mathbf{p}_r$ has the same definitions as $\mathbf{P}_d$ in the previous section. Therefore, they also satisfy Equation 3.20.

$$\mathbf{p}_r^T\mathbf{E}\mathbf{p}_l = 0 \quad (3.21)$$

Equation 3.21 seems to be a very simple criterion for spatial matching validation. However, it is worth noting that it is only a necessary condition for matching. Even two objects satisfy this equation, it is not safe to regard them as the same object. For example, in Figure 3-9, all objects on line O_{rP} has the same image on camera 2. It is not possible to distinguish them with only two cameras.

3.2.2.3 Hybrid dot spatial matching

The spatial matching algorithm is to create the corresponding relationship between 2D trajectories from different camera views. The input of variables are the coordinates of 2D trajectories in image frames, as described in section 3.2.1, and the calibrated camera parameters. The output variables are the 3D coordinates of the trajectories in world frame.

Before developing the algorithm, we needed to clearly define the matching criteria. As described before, the temporal tracking procedure could be processed in both 2D and 3D. For 2D tracking, let us denote $\mathbf{P}_{i,j,t}$ as the image coordinates of a particle on the jth trajectory in camera i at the tth frame. Define $\mathbf{P}_{1,j,t}$ and $\mathbf{P}_{2,j',t}$ to be matched if they satisfy Equation 3.21. If all the particles on the jth trajectory $\mathbf{P}_{1,j,1} \sim \mathbf{P}_{1,j',t}$, in camera 1 are matched with the particles $\mathbf{P}_{2,j',1} \sim \mathbf{P}_{2,j',t}$ on the j'th trajectory in camera 2, we say these two trajectories are matched. Since it has taken the advantage of temporal tracking information, the matching ambiguity is largely

reduced (Guezennec et al., 1994). On the contrary, for 3D tracking, the spatial matching algorithm is processed before the temporal tracking. It fully relies on the information from the images. Lacking the temporal information, the third camera becomes necessary to reduce the ambiguity. The principle is shown in Figure 3-9; l_l is an epipolar line in image 1. If we have a third camera denoted as the up camera, it will produce the second epipolar line l_l'. These two lines will uniquely determine the position of $\mathbf{p}_l$. We say that $\mathbf{P}_{1,j,t}$, $\mathbf{P}_{2,j',t}$ and $\mathbf{P}_{3,j'',t}$ are matched if either two of them satisfy Equation 3.21. This approach has been widely used in recent research.

In practical application, the spatial tracking algorithm is usually vulnerable due to missing particles or noise raised by image processing. It is very hard to continuously detect trajectories from the first frame to the last frame without any missing points. The detected 2D trajectories tend to be broken into shorter segments. The VPTV algorithm is trying to obtain trajectories as long as possible by taking additional information from the third camera.

Assume the particle trajectories are recorded from frame 1 to 40. The gray cells in Table 3-2 are detected 2D broken trajectories. The first index of T indicates the camera number, and the second index of T denotes the trajectory number. For example, $T_{1,1}$ is the first trajectory in camera 1; $T_{3,2}$ is the second trajectory in camera 3. The columns stand for the frame number as indicated in the first row. We also assume that $T_{1,1}$, $T_{1,2}$ and $T_{1,3}$ all belong to the same long trajectory which is broken into pieces due to a missing particle. $T_{2,i}$ (i=1~2) and $T_{3,j}$ (j=1~3) are the corresponding trajectories in camera 2 and camera 3 respectively. The 2D PTV tracking algorithm uses two cameras to do spatial matching and uses Equation 3.15 to reconstruct the 3D trajectories. For example, if camera one and camera two are used, the detected trajectory would be $T1_1$ and $T1_2$. Only particles observed by two cameras at the same time could be detected and recovered. The 3D tracking PTV algorithm has an even stricter requirement. Because no temporal tracking information is used in spatial matching, it requires the particles to be simultaneously observed by at least three cameras. The detected trajectories, for example $T2_{i,}$ are even shorter than $T1_i$. The VPTV approach is developed by trying to overcome this limitation. To

obtain longer trajectories, it adopted the following approach: (1) the trajectories are regarded as matched if all available particles satisfy Equation 3.21. For example, $T_{1,1}$, $T_{2,1}$, and $T_{3,1}$ are considered matched because either two particles from the 6^{th} frame to the 9^{th} frame satisfy Equation 3.21. (2) Once the trajectories are matched, the missing particles in one camera can be recovered from the information of the other two cameras based on Equation 3.15. In this way, a longer trajectory $T3_1$ is obtained.

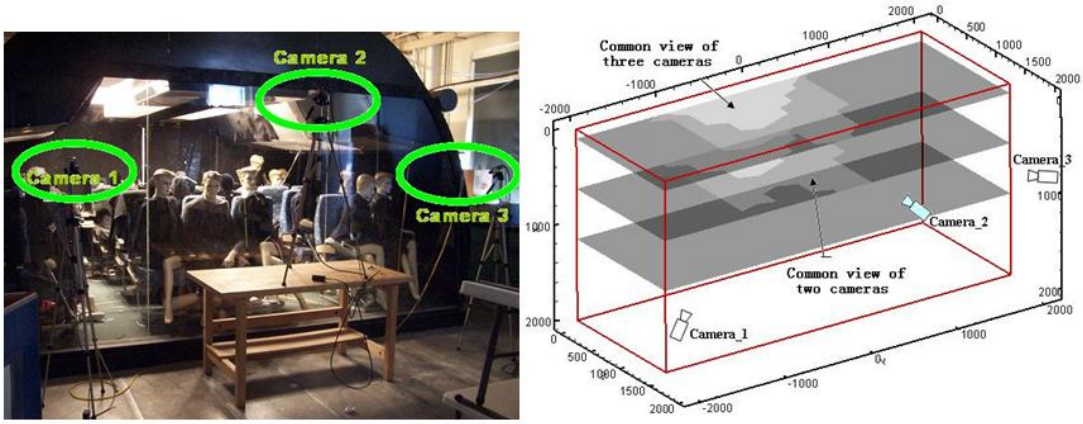
However, this approach is still limited by the strict requirement of spatial matching: at least three cameras are needed to recover the 3D trajectories. In our experiment, long trajectories are demanded. But at the same time, we want to detect as many trajectories as possible. Due to the poor illumination condition in the full-scale aircraft cabin, it is extremely challenging to ensure a particle is observed by all cameras. According to the experiment result, less than ten trajectories are detected by applying the VPTV algorithm. The detectable volume of VPTV is also limited. The common view of three cameras is usually much smaller than that of two cameras. This will significantly reduce the detectable volume in the experiment.

Table 3-2. The detected 2D particle trajectories on three cameras.

Frame	1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40
Camera 1	$T_{1,1}$				$T_{1,2}$			$T_{1,3}$
Camera 2	$T_{2,1}$							$T_{2,2}$
Camera 3		$T_{3,1}$			$T_{3,2}$			$T_{3,3}$

Table 3-3. The detected 3D particle trajectories using different approaches.

Frame	1-5	6-10	11-15	16-20	21-25	26-30	31-35	36-40
2D tracking PTV	$T1_1$							$T1_2$
3D tracking PTV		$T2_1$						$T2_2$
VPTV	$T3_1$							$T3_2$
HPTV	$T4_1$				$T4_2$			$T4_3$



(a) The experimental setup of cameras (b) The common view of cameras

Figure 3-10. Compared common view of two and three cameras.

Figure 3-10 (a) shows a typical arrangement of the experiment. Camera one and camera three are observing the target volume from the left and right sides respectively, while camera two is observing from the up direction. The experiment is conducted within cuboids with a dimension of 4mx1.5mx2m. In order to visualize the common view of the cameras, we divided the volume with cubic cells at a size of 0.01mX0.01mX0.01m. Each cell is projected to the image coordinate of the three cameras. If a cell is beyond the image scope (1392 pixels x 1040 pixels), it is discarded. A cubic cell is marked as the common view of two cameras if it was within the scope of either two image spaces. Similarly, we also matched the common view of the three cameras. Figure 3-10 (b) compares the detectable volume of the three and two camera systems. Obviously, the common view of two cameras is larger than that of three cameras.

To improve the detecting rate and increase the detectable volume, we propose the hybrid spatial matching strategy. The basic idea is matching and constructing the 3D trajectories using both three cameras and two cameras. If a trajectory is observed by all cameras, an algorithm similar to VPTV will be applied. If it is only visible to either two cameras, a 2D PTV algorithm will be used. Details of the algorithm are described as follows:

(1) Define particle number threshold N. All 2D trajectories with less than N particles will be discarded. In the current algorithm, N is set to be 6.

(2) With extrinsic camera parameters and Equation 3.20, derive the essential matrix between either two cameras E_{12}, E_{13}, and E_{23}.

(3) The three camera spatial matching algorithm is processed first. As shown in Figure 3-11, $\mathbf{P}_{i,j,k}$ is a 2D coordinate of particles obtained after temporal tracking. i is the camera number; j is the trajectory number; and k is the frame number. Each column represents one 2D trajectory. Notice that in practice, the frame number k could be less than the total frames due to a missing particle, broken trajectory or false detection. It is also possible that a trajectory does not start from the first frame. All these missing points are filled by NULL in the trajectory data file so that the length of all trajectories is identical. Consider the situation when we select the j^{th} trajectory in camera 1. Our purpose is to find its corresponding trajectories in camera 2 and camera 3. At the first frame, the first coordinate of the trajectory in camera one is $\mathbf{P}_{1,j,1}$. Scan all particles at the first frame in camera 2, $\mathbf{P}_{2,j',1}$. If it satisfies

$$\mathbf{P}_{1,j,1}\mathbf{E}_{1,2}\mathbf{P}_{2,j',1} \le match\_threshold\,, \quad j' = 1,...J' \tag{3.22}$$

where match threshold is a predefined threshold for spatial matching; J' is the total trajectories in camera 2; it is considered a "matched" particle. Let $\Omega 2_1^j$ denote the set of particles of the first frame in camera 2 satisfying Equation 3.22. In this notation, the "2" indicates camera two. The "1" indicates the first frame. And the label "j" refers to the j^{th} target trajectory in camera one. The same approach is repeated to particles in camera 3 as well. Let $\Omega 3_1^j$ denote the set of matched particles in camera three. Then we match $\Omega 2_1^j$ and $\Omega 3_1^j$ by examining the essential matrix equation between camera two and camera three:

$$\Omega_1^j = \left\{ (j, j', j'') : \left| \mathbf{P}_{2,j',1}\mathbf{E}_{2,3}\mathbf{P}_{3,j'',1} \le match\_threshold; j' \in \Omega 2_1^j; j'' \in \Omega 3_1^j \right. \right\} \tag{3.23}$$

Where j" is the trajectory index in camera three.

Ω_1^j is the set of matched particles by three cameras.

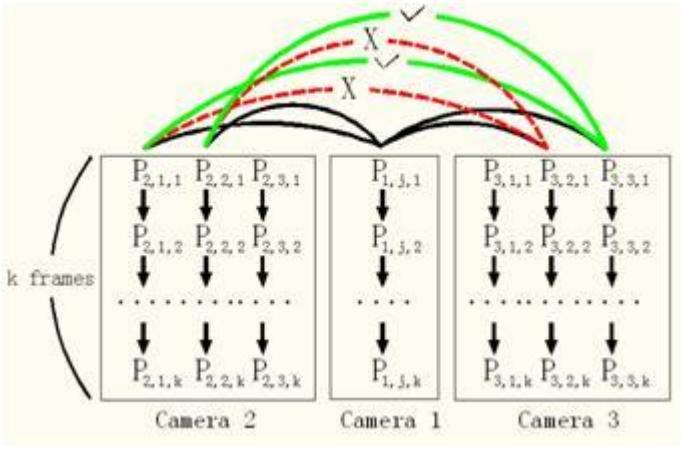


Figure 3-11. Spatial matching approach.

The whole procedure is described in Figure 3-11. Particle $\mathbf{P}_{1,j,1}$ is matched with $\mathbf{P}_{2,2,1}$ and $\mathbf{P}_{3,3,1}$ respectively. And $\mathbf{P}_{2,2,1}$ and $\mathbf{P}_{3,3,1}$ are matched with each other. Therefore, we define $\mathbf{P}_{1,j,1}$, $\mathbf{P}_{2,2,1}$, and $\mathbf{P}_{3,3,1}$ are three-way matched particles. Denote (j, j', j'') as a three-way matched triple. Their corresponding trajectories in three cameras are matched as the trajectory candidates. The same procedure is processed onto the next frames and so on. If particles up to six frames are matched, the trajectories will be recorded as three-way matched trajectories. For example in Table 3-4, since the particles between the 11th and 20th frames are three-way matched, (1,1,1) will be the matched triple, and $T_{1,1}$, $T_{2,1}$ and $T_{3,1}$ are the matched trajectories.

Table 3-4. The three-way matching example.

Frame	1-10	11-20	21-30	31-40	41-50	51-60
Camera 1	$T_{1,1}$					
Camera 2	$T_{2,1}$					
Camera 3		$T_{3,1}$			$T_{3,2}$	

(4) The two-way spatial matching process follows right after the three-way matching approach. To avoid overlapping with two-way matching trajectories, all three-way matched particles are set to be NULL in the data file. For example, if Table 3-4 is the original trajectory data file,

after three-way matching it will become Table 3-5. The data between the 11th and 20th frame are filled with NULL, but the other data is still kept. The new data file has the same data structure as Figure 3-11 except that there are no three-way matched trajectories in it. Two-way matching is still processed based on the epipolar geometry requirement: Equation 3.21. Similar to step 3, two trajectories are considered matched if up to six frames satisfies the epipolar geometry requirement. All matched particles in the data file are set to be NULL immediately to avoid the duplicated detecting. Camera one and camera two are matched first, then camera one and camera three, and finally camera two and camera three. This order is arbitrarily selected. Any other order would work as long as all these three pairs are processed.

Table 3-5. The trajectory data file after three-way matching.

Frame	1-10	11-20	21-30	31-40	41-50	51-60
Camera 1	$T_{1,1}$		$T_{1,1}$			
Camera 2	$T_{2,1}$					
Camera 3			$T_{3,1}$		$T_{3,2}$	

In step 1, we have set a minimum length for the trajectories. This criteria is proposed based on two facts: In the temporal tracking process, we need at least four frames long in order to calculate the cost function of the trajectories (Li, 2008). In the spatial matching process, long trajectories are usually preferred to validate the corresponded matching relationship. The more particles on the trajectories, the less strict threshold we could use in Equation 3.22. Equation 3.22 is usually not zero in reality even if two particles were exactly matched. The error could come from the image processing, camera calibration or synchronization. Using a large threshold is helpful to reduce the possibility of rejecting the true trajectories.

The current algorithm used an essential matrix instead of a fundamental matrix to do spatial matching between cameras. In practice, a fundamental matrix is straight forward to use because it directly substitutes the pixel coordinates into Equation 3.22. However, it hasn't considered the distortion of the lens. This will definitely prohibit its accuracy. On the other hand, essential matrix uses the normalized camera coordinates. The distortion has been modeled during the data conversion.

Since both three-way and two-way spatial matching are adopted in the algorithm, all matched particles are discarded immediately after step 3 and step 4 to avoid the duplication of detected trajectories. This strategy has significantly improved the efficiency and accuracy of the algorithm.

3.2.2.4 3D dot trajectories reconstruction

After the spatial matching, we obtained the three-way or two-way matched trajectories. Picking up the image coordinates of particles on one trajectory and substituting them into Equation 3.9, Equation 3.14 and Equation 3.15. For two-way matched particles, there are four equations with three unknowns. For three-way matched particles, there are six equations with three unknowns. These equations are independent with each other as long as the lens is not coplanar or collinear. The 3D physical positions of the trajectories could be obtained by solving these least square problems. The reconstructed 3D trajectories are reconstructed as the following format:

Table 3-6. The detected 3D trajectory table.

	Frame 1			Frame 2			Frame j	
	x	y	z	x	y	z	x	y
Traj 1	754	435	135	737	432	352	……	……
Traj 2	829	302	146	812	299	368	……	……
Traj 3	845	331	149	829	329	374	……	……
Traj i	871	271	169	855	268	386	……	……
……	……	……	……	……	……	……	……	……

3.3 Streaks tracking algorithm

Reconstruction of the instantaneous three-dimensional velocity field from multi-views of particle streak images is based on the stereo vision technology (Barnard and Fischler, 1982; Medioni and Nevatia, 1985). To the author's best knowledge, the earliest automated 3D particle

streaks tracking velocimeter was developed in the late 1980s to study the gas flow through a transparent engine throttle-body (Adamczyk and Rimai, 1988a). In this study, two orthogonal cameras are set up to simplify the reconstruction procedure and improve the accuracy over small-angle stereoscopy. Their matching and reconstruction procedure is shown in Figure 3-12. Pick up streak vector A in camera 2. Assume the extremes of these vectors are (x_{orig2}, y_{orig2}) and (x_{end2},y_{end2}). From the knowledge in section 3.2.2.2, two epipolar lines are projected onto the video plane of camera 1. The computer then picks the camera 1 vector closest to these projection lines minimizing the sum of the squares of the distance between a vector's origin and the origin projection from the other view. If this parameter is less than a predefined threshold, this vector pair will be accepted as a matched pair candidate. Then repeat the similar procedure in the reversed direction, starting from camera 1 and projecting onto camera 2. If two streaks are matched in both directions, they are accepted as the real matched pair and substituted into Equations 3.9, 3.14 and 3.15 to reconstruct the 3D streak.

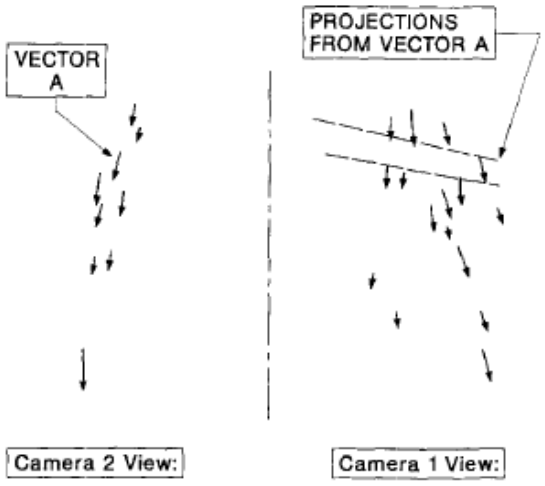


Figure 3-12. Reconstruction procedure from two orthogonal cameras (Adamczyk and Rimai, 1988).

VPSTV (Sun, 2007) proposed a similar approach to detect the particle streak in a larger volume. Two synchronized cameras with different exposure times are adopted. Assuming the

exposure time of camera 1 is t_1, choose $0.5t_1$ as the exposure time for camera 2. The matching procedure is shown in Figure 3-13. First, pick up the endpoint of one vector in camera 1, for example point A. Search the matching points in camera 2 by substituting the endpoints of all streaks into a test equation equivalent to Equation 3.21. If the parameter is less than a threshold, these two streaks are candidate pairs. Then substitute the coordinate of point C, which is the middle point of the selected vector, and the endpoints of the corresponding streaks in camera 2 into the test equation. If the parameter is also less than the threshold, they are accepted as matched pairs. Similar to section 3.2.2.4, Equations 3.9, 3.14 and 3.15 are applied to reconstruct the 3D streaks.

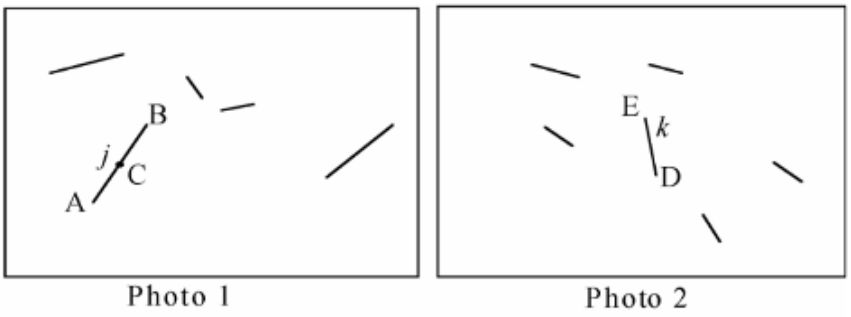


Figure 3-13. Reconstruction procedure from two cameras with different exposure time (Sun, 2007).

The efficiency and uniqueness of the matching process depends on the threshold value being selected. A compromise between the matching accuracy and the number of detected streaks has to be made. According to a previous report (Adamczyk and Rimai, 1988a), if two cameras are used, a value threshold equal to 2.5 results in obtaining 3-16 streaks from projection files containing 20-25 streak pairs. The ambiguity of matching is due to a lot of factors, for example, the illumination, the pixel resolution, the set-up of the cameras, optical distortions, particle density, and so on. Decreasing the particle density will definitely reduce the ambiguity of the matching procedure. However, a trade-off between reduced spatial resolution and ambiguity has to be considered.

Asynchronization is another frequent problem that prohibits the matching procedure. Assume that the three camera shutters open at t_1 t_2 and t_3 respectively ($t_2>t_1>t_3$). Define the synchronization error between camera 1 and camera 2 as $t_{e12}=t_2-t_1$. The frame rate of our current CCD cameras is 30 fps. The exposure speed we used is 1/120s-1/60s, which corresponds to 8.3ms-16.6ms. Assume the synchronization error between two cameras te_{12} is 3ms, which is about 36%-18% of the exposure time. The endpoints of the streaks will be shifted to18% to 36% from the exact point since the streaks are generated during the exposure time. Therefore, it will be extremely difficult to apply Equation 3.21 directly if the asynchronization exists. Fortunately, it is still possible to compensate this asynchronization if a tricky procedure is applied.

In the following section, a unique 3D streaks tracking algorithm, which can resist the asynchronization, will be introduced. Since we have three cameras in the current system, it is straight forward to perform the spatial matching. As described in 3.2.2.2, two intersected epipolar lines are corresponding to one endpoint. This greatly reduced the ambiguity of the matching procedure. Much denser particle streaks could be detected compared with traditional methods.

The detail of the algorithm is illustrated in Figure 3-14.

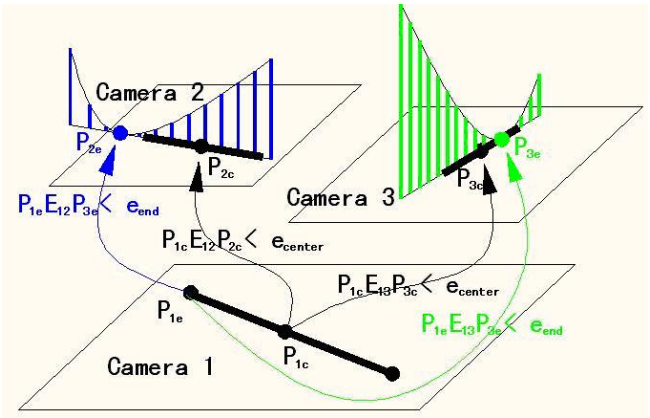


Figure 3-14. Streak matching procedure for three asynchronized cameras.

(1) Select one streak in camera 1, whose exposure time is in the middle of the three cameras ($t_2>t_1>t_3$). Pick up the center point of this steak. Assume its coordinate is $\mathbf{P}_{1c}$.

(2) Define a threshold e_{center}. Substitute $\mathbf{P}_{1c}$ and the center points of all streaks in camera 2 into Equation 3.21. If a parameter is less than e_{center}, this streaks pair will be marked as a matched pair candidate. Denote all matched streaks in camera 2 as $\mathbf{S}_2$. The value of e_{center} must be determined empirically. A proper value is obtained by trial and error, considering the accepted rate and the accuracy of the matching procedure. In general, e_{center} should not be very strict because asynchronization does exist in the experiment. It is impossible to obtain a test parameter of exactly zero.

(3) In a similar way, substitute $\mathbf{P}_{1c}$ and the center points of all streaks in camera 3 into Equation 3.21. The streaks are marked as matched candidates between camera 1 and camera 3 if their parameter is less than e_{center}. Denote all matched streaks in camera 3 as $\mathbf{S}_3$.

(4) The two resultant streak sets $\mathbf{S}_1$ and $\mathbf{S}_2$ are further validated. Pick up two points, for example $\mathbf{P}_{2c}$ and $\mathbf{P}_{3c}$ from $\mathbf{S}_1$ and $\mathbf{S}_2$ respectively, and substitute them into Equation 3.21. If the parameter is also less than e_{center}, these three streaks ($\mathbf{P}_{1c}$,$\mathbf{P}_{2c}$,$\mathbf{P}_{3c}$) are matched pairs. In the above algorithm, the projecting procedure is performed in three directions: camera 1 to camera 2; camera 1 to camera 3 and camera 2 to camera 3. In image space, this procedure is equivalent to searching the intersection point of two epipolar lines, which is much more efficient than searching along a single line.

(5) Next, the computer will synchronize the three cameras by shifting the position of streaks in cameras 2 and 3. The exposure time of camera 1 is selected as the time reference because the shifting distance will be minimized. Consider the two matched streaks of camera 1 and camera 2 graphically described in Figure 3-14. Assume that the synchronization error is smaller than half of the exposure time. Then the maximum shifting distance will be half of the streak. If the exposure time is short enough, the streak is approximately a straight line. The corresponding point of $\mathbf{P}_{1e}$ in camera 2 will be chosen along the extended line through $\mathbf{P}_{2c}$. Extend the streaks in camera 2 for half of the original length at both ends. Substitute $\mathbf{P}_{1e}$ and all points on the corresponding streak into Equation 3.21. The point minimizing the test

parameter is the candidate of the synchronized endpoint. If it is also less than a predefined threshold e_{end}, it will be accepted as the new endpoint. In a similar way, the other synchronized endpoint could also be located. Repeat the whole process for other matched streak pairs and for camera 3. At the end of this step, the synchronized new streaks pairs ($\mathbf{P}_{1c}$,$\mathbf{P}_{2c}$',$\mathbf{P}_{3c}$') are marked.

(6) Substitute the endpoints of synchronized streaks into Equations 3.9, 3.14 and 3.15 to reconstruct the 3D streaks.

3.4 Summary

In this chapter, we described the procedure of a hybrid volumetric particle tracking system. The first section is an image processing algorithm. Its purpose is to isolate the particles and streaks on the raw images. At the same time, the location, length, and orientations of these particles are detected and recorded in the data files. These data are then processed following two separated paths. The dot particles are processed using the 2D temporal tracking and 3D spatial matching algorithm illustrated in section 2. The streak particles are processed using the streaks tracking algorithm in section 3. Finally, both dots and streaks data are delivered to a post-processing unit to perform the display and rendering.

The uniqueness of this system is that it introduced two hybrid concepts: the spatial matching and reconstruction hybrid and the tracking object hybrid. By employing these two approaches, the detectable velocity range and measurement volume of the system has been increased. Another novel idea developed in the algorithm is the artificial synchronization technique. It will significantly improve the accuracy of the streak tracking algorithm.

CHAPTER 4 DISCUSSION AND ANALYSIS

Like other particle tracking algorithms, the proposed HVPT algorithm should be validated before application. Such validation is challenging because we do not know the "exact answer" corresponding to the experiment data. As discussed in Chapter 3, the trajectory construction procedure consists of several steps. Each step will introduce certain errors. In this chapter, we will discuss these algorithms and analyze the errors associated with them. The synthetic images with known particle trajectories will be used to validate the performance of the HVPT system.

4.1 Algorithm discussion

4.1.1 Particle center detecting algorithm

The first task of the HVPT system is to process the images and extract the image coordinate of the particles. Errors in this procedure could arise from multiple sources, including random thermal noise, finite spatial resolution of the camera sensor, the detection of scattered light, high particle seeding density, diffraction of the aperture, and imperfect lens focusing (Feng et al., 2007; Ouellette et al., 2006). In traditional PIV or small scale PTV experimental set-ups, seeding particles are usually illuminated by a laser sheet with band-pass optical filter. Very sharp images are generated. Several particle detecting algorithms had been proposed for these applications. The typical algorithms include weight averaging, function fitting, template matching, image cross correlation, etc.

The weight averaging method is one of the most efficient and simple approaches among these algorithms. It is also referred to as the "moment method." At the first step, the recorded images are binaralized. A gray threshold I_{th} is chosen. All pixels brighter than I_{th} are set to be white; otherwise, it is black. The binary images are then segmented into groups of contiguous

pixels representing individual particles. In the segmentation process, some algorithms only select the contiguous white pixels as the particle regions. Other algorithms use a minimized polygon or rectangle that encloses all the contiguous white pixels as the particle region (Feng et al., 2007). The center (x_c,y_c) of the particle is then determined by the gray-value weighted average position of the pixels within the particle region. Let I (x,y) represent the gray-value of pixel at image coordinate (x,y), the equation to calculate particle position is:

$$\mathbf{X}_{calc} = \frac{\sum_k \mathbf{X}_k I_k(x,y)}{\sum_k I_k(x,y)} \tag{4.1}$$

Where $\mathbf{X}_k$ is the image coordinate vector of the contiguous pixels and $\mathbf{X}_{calc}$ is the calculated particle position vector.

Another approach for particle detecting is the Gaussian fitting algorithm (Cowen and Monismith, 1997). A Gaussian function is proposed to approximate the intensity profile of a particle image:

$$I(x,y) = \frac{I_0}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\left(\frac{x-x_c}{\sigma_x}\right)^2 + \left(\frac{y-y_c}{\sigma_y}\right)^2\right]\right\} \tag{4.2}$$

Given an image with N local maximums, N Gaussian functions are used to fit these regions. The center of the Gaussian function is regarded as the particle centers. The advantage of this approach is that it can handle the overlapping particles, and it is highly accurate (Ouellette et al., 2006). However, it is computationally time consuming. And the biggest problem is that it needs to determine five parameters (I_0, x_c, y_c, σ_x, σ_y) for each particle, which means at least five contiguous pixels are required to identify the particle center. If the particle image is smaller than five pixels, this algorithm will fail. To overcome this problem, the one-dimension Gaussian fitting approach was proposed (Cowen and Monismith, 1997). Overall two 1-D Gaussian functions are needed. One function is responsible for vertical detecting; the other one is for horizontal detecting. In each direction, at least three contiguous pixels are needed for the regression.

A neural network approach is a recently developed particle-detecting algorithm (Carosone et al., 1995; Ouellette et al., 2006). In this algorithm, a 9x9 pixel window centered on the local intensity maximum is used as the input layer of 81 neurons. The output layer is two neurons corresponding to the horizontal and vertical position of the particle center. After a one-time training, the authors reported a successful performance using this algorithm. The neural network approach is promising because it could easily handle the problem of overlap and robust to noisy images.

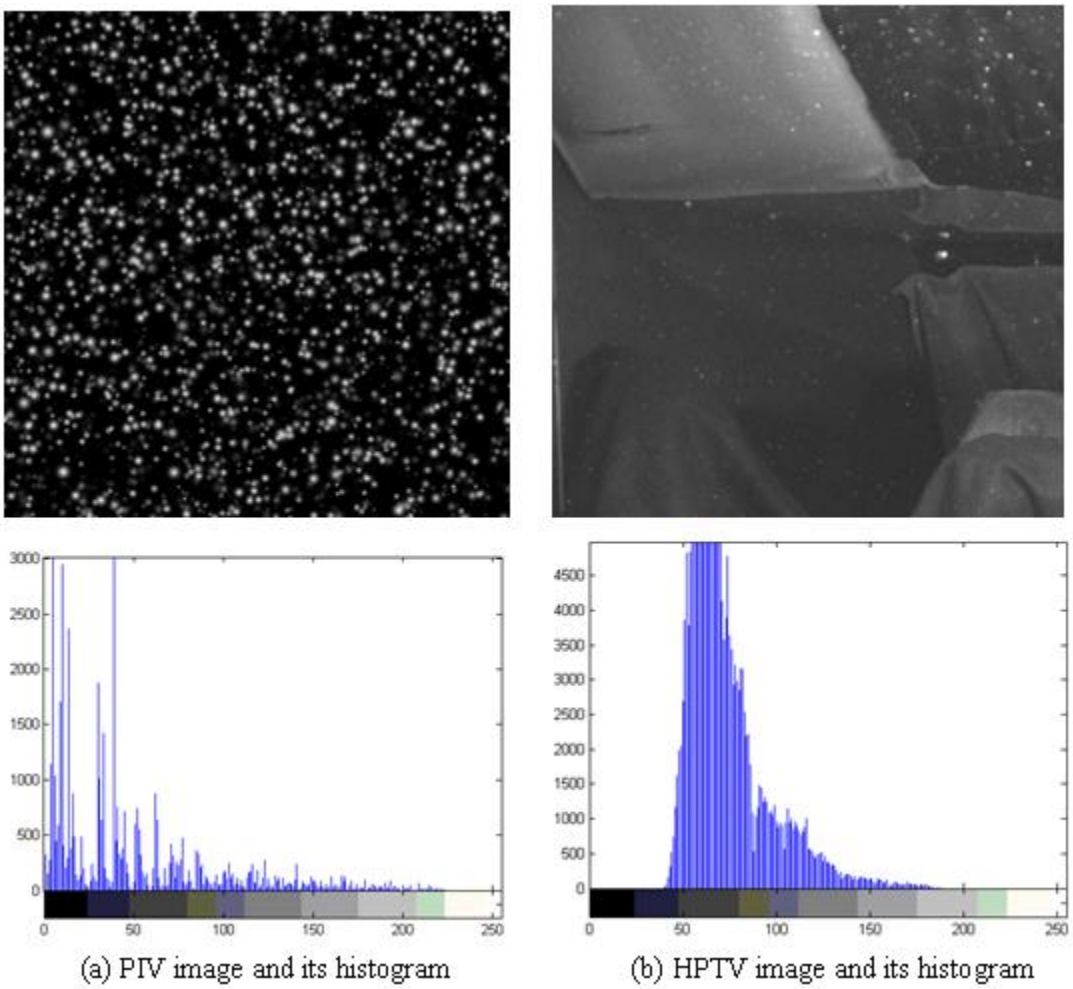


Figure 4-1. Compared raw and histogram images of PIV and HPTV system.

Directly applying these algorithms to our experiment is difficult. We used normal light as the illumination light source instead of a laser sheet; therefore, the edges of particle images are

not as sharp as those obtained in the PIV or small scale PTV experiment. Figure 4-1 compared the images obtained by the HPTV and PIV systems. The left column shows the PIV image and the pixel intensity histogram. The right column shows the HPTV image and its histogram. The horizontal axis is the pixel intensity while the vertical axis is the pixel number. As shown in the histogram figures, the pixel intensity spectrum of PIV is widely distributed. It means that the background (low intensity pixels) and particle image (high intensity pixels) are separated well. It will be easy to isolate the particles. But in the image of HPTV, the spectrum is shrinking to the narrow middle intensity and behaving as a bell shape. It means that the intensity of the background is high and comparable with those of particle images. It will be challenging to do the image processing under such situation.

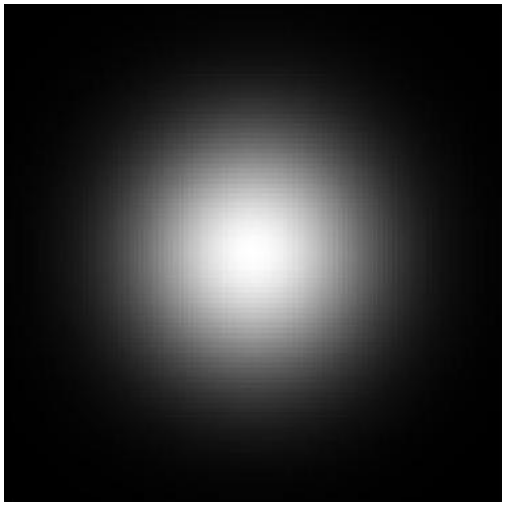


Figure 4-2. Simulated particle with 100x100 pixels (the maximum intensity is 255).

In previous research (Cowen and Monismith, 1997; Feng et al., 2007; Ishikawa et al., 2000; Ouellette et al., 2006; Pereira et al., 2006; Ruhnau et al., 2005), pure synthetic images are usually used to validate the performance of the algorithms. Initially, a black image is used as the background. Then the particles in either Gaussian intensity profile or Bessel function profile are added to the background image. The particle density, size and intensity are controlled and varied so as to test the detecting rate and accuracy of the particle detecting algorithms. The response to

the noise level could also be investigated by adding proper Gaussian profile noise. However, a black background is not an appropriate choice for the current research. Because ordinary light is used as the light source, the furniture and manikins are clearly observed in the images. We found that "white noise" is not the dominate error source. Therefore, we used the semi-synthetic images to validate the algorithm.

The semi-synthetic did not use the black image as the background; instead, it used a real background image of the cabin without releasing bubbles. It is more representative because the image noise is not artificially simulated but "real." The furniture and manikins are also "real." The only artificial part is the particles with accurately known positions in the image.

The particle pattern design is straight forward. It has the basic 2D Gaussian bell shape. Given the particle diameter D_p, the standard deviation is set to be $D_p/6$. Assume that the maximum particle intensity is I_{max}, then the intensity profile of the simulated particle image becomes:

$$I_p(x,y) = \frac{18 I_{\max}}{\pi D_p^{\,2}} \exp\left\{-18\left[\left(\frac{x-x_c}{D_p}\right)^2 + \left(\frac{y-y_c}{D_p}\right)^2\right]\right\} \tag{4.3}$$

Figure 4-2 shows one example of the simulated particle images with particle size equaling to 100x100 and peak intensity equaling to 255. Thousands of these particles will be generated and randomly plotted on the background image as shown in Figure 4-3. The pixel values at these particle locations are determined by the following equation:

$$I(x,y) = \max\left\{I_b(x,y), I_p(x,y)\right\} \tag{4.4}$$

Where I_b is the intensity value of the background image.

With these artificial simulated images, the particle-detecting algorithm is applied to search the center coordinate of these particles. In this study, we define two particles are matched if the distance between the detected and exact locations are less than 2 pixels. For each case (given certain I_p and D_p), twenty replications are simulated. Figure 4-3 shows four typical semi-synthetic simulations we conducted.

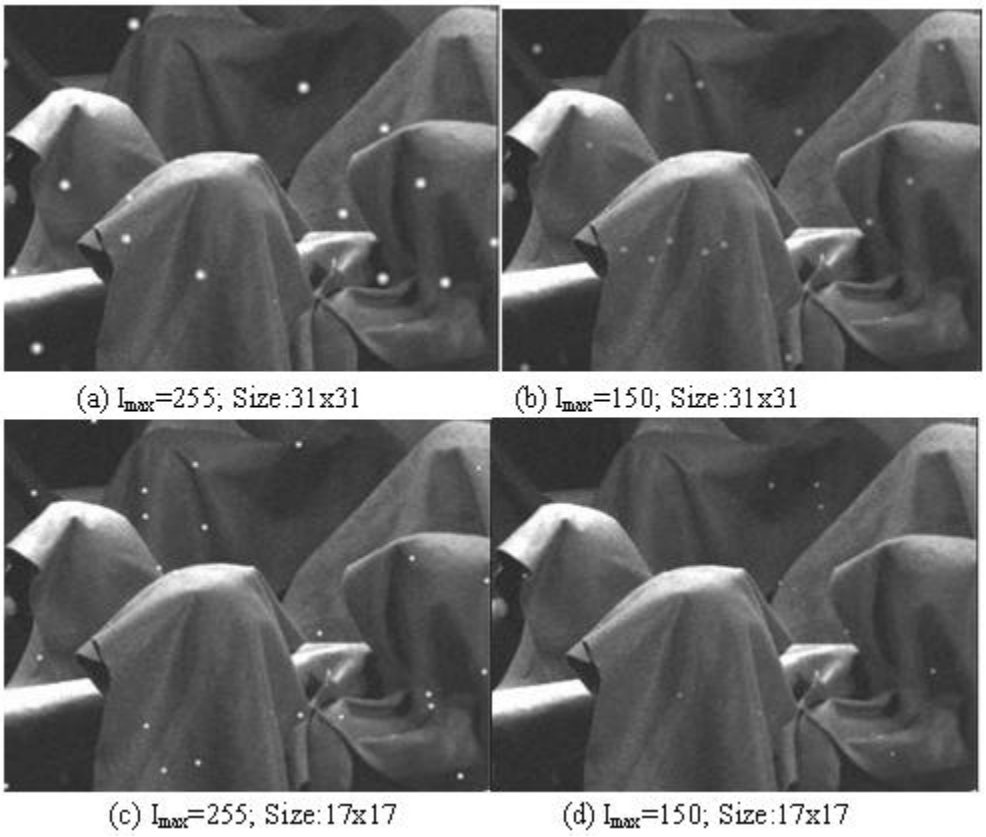


Figure 4-3. Typical examples of the semi-synthetic images taken in a aircraft cabin mock up.

In order to quantitatively evaluate the performance of the tracking algorithm, we define a detecting rate as:

$$r_{\det ected} = \frac{N_{\det ected}}{N_{exact}} \tag{4.5}$$

Where $N_{detected}$ is the number of particles correctly detected by the algorithm. N_{exact} is the number of total released particles. The incorrect rate is defined as:

$$r_{incorrect} = \frac{N_{fail}}{N_{total}} \tag{4.6}$$

Where N_{fail} is the number of the falsely detected particles.

These fake particles are artificially generated by image processing algorithms or from device noise, therefore, should be excluded. N_{total} is the total number of detected particles and

equals to the sum of N_{fail} and $N_{detected}$. Figure 4-4 and Figure 4-5 show the detecting rate and incorrect rate obtained by analysis of the semi-synthetic images we generated. Each dot represents one simulation. From these figures, we observed that a larger particle size will benefit the particle detecting algorithm. In Figures 4-4 and 4-5, a higher detecting rate is obtained as we increase the particle image diameter. At the same time, the incorrect rate decreased as we expected.

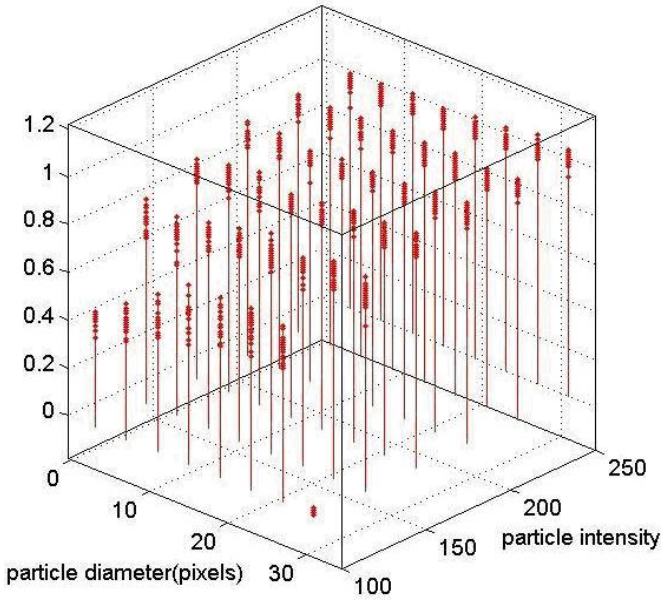


Figure 4-4. Detecting rate varying with particle size and intensity (brightness).

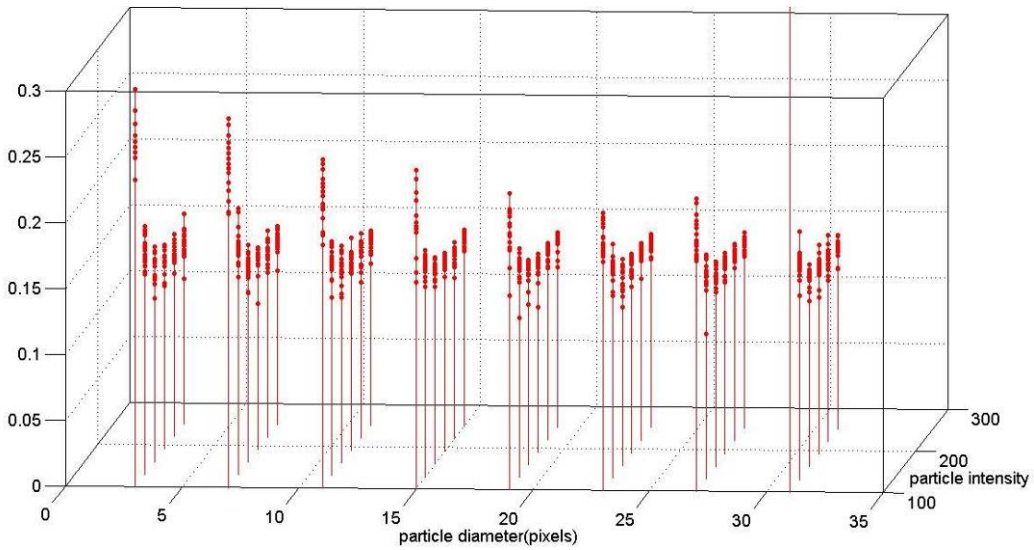


Figure 4-5. Incorrect rate changing with particle size and intensity (brightness).

The particle intensity is another key factor for particle detecting. As shown in Figure 4-4, the algorithm detected almost all of the particles when I_{max} is close to 250, even though the particle diameter is small. Figure 4-6 related the detecting rate and incorrect rate to the particle-intensity according to the semi-synthetic image testing we conducted.

Detecting error is another algorithm performance character of concern. It is defined as the distance between the exact particle location and the predicted location by the algorithm.

$$\varepsilon_{detecting} = \sqrt{\left(x_p - x_c\right)^2 + \left(y_p - y_c\right)^2} \tag{4.7}$$

In the semi-synthetic image testing, 100 particles are released for each simulation case. Overall, 1.344×10^5 particles are released and 8.1112×10^4 of them are detected by the algorithm. We analyzed the error distribution shown in Figure 4-7.

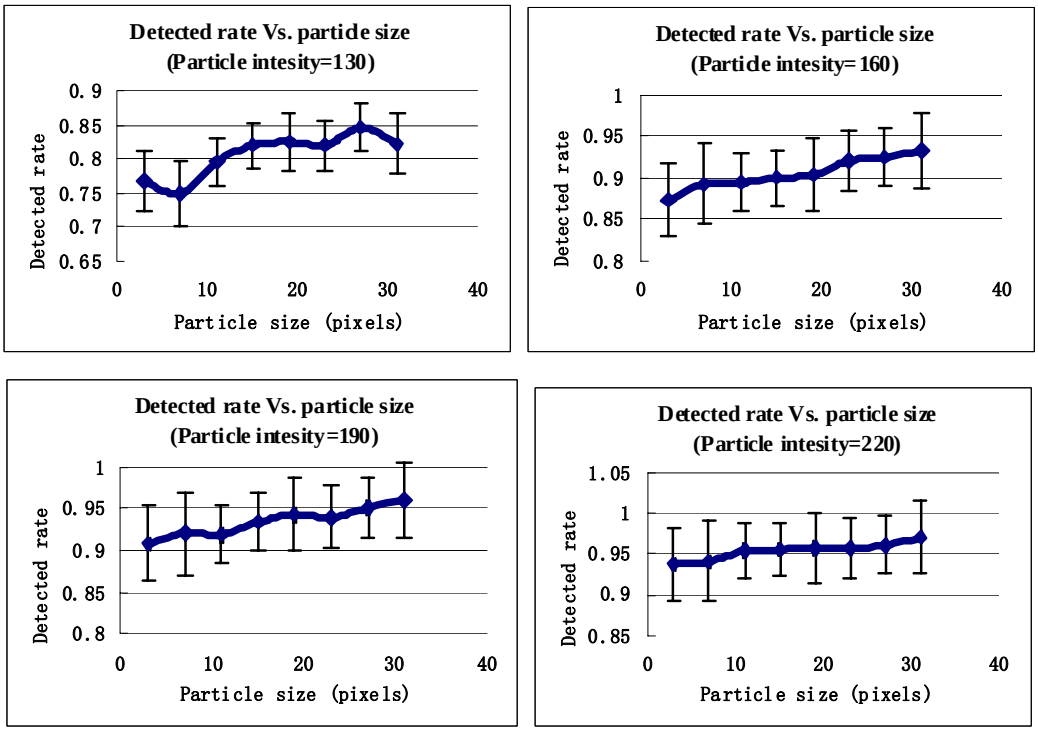


Figure 4-6. The detecting and incorrect rates vary with particle size and intensity.

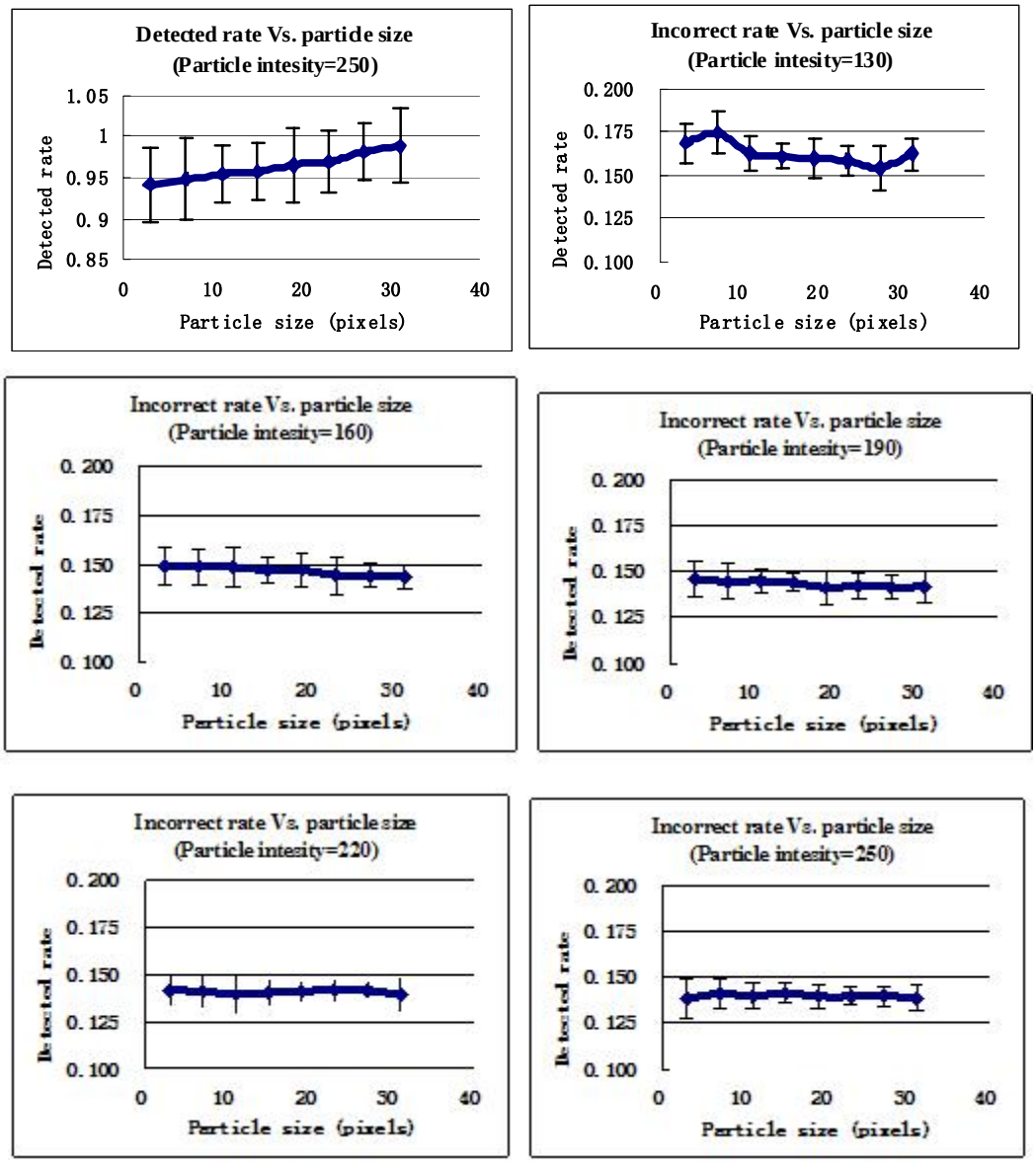


Figure 4-6. (cont.).

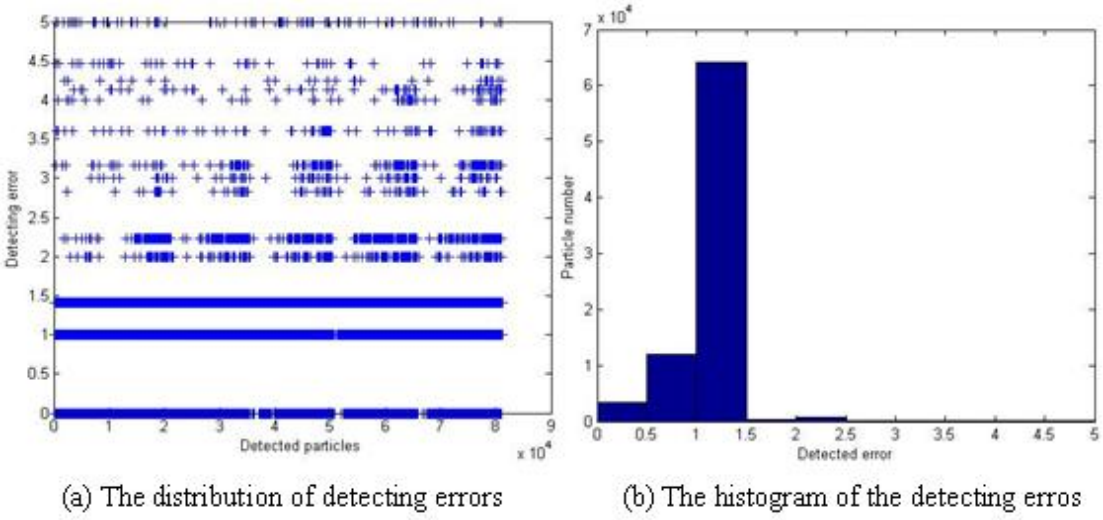


Figure 4-7. The distribution and histogram of the detecting error.

The detecting error is discretely distributed in the figure because the pixel coordinates are integer numbers. The algorithm works perfectly if the detecting error equals to 0. If the detecting error is 1, it means that either the x or y coordinate has one pixel error. Similarly, if the detecting error is 1.44, it means that both the x and y coordinate has one pixel detecting error, and so on. From Figure 4-7, it is observed that most detected particles have less than one pixel error in both x or y directions because very few errors beyond 1.44 pixels. There is a very interesting periodic pattern in Figure 4-7(a), especially at the detecting error ranged from 2 to 3.5. In fact this pattern is caused by our simulation order. We fixed particle intensity at a certain value, and increased particle size from 3 to 31 pixels periodically. This pattern indicated that increasing particle size tends to introduce some errors and affects the accuracy of the algorithm.

4.1.2 Camera calibration algorithm

Camera calibration toolbox developed by the computer vision research group at the California Institute of Technology is used in current research. Their work is mainly based on two papers (Heikkila and Silven, 1997; Zhang, 1999). The calibration process they recommend consists of six steps:

(1) Print a calibration pattern and attach it to a planar surface.

(2) Take a few images of the model plane under different orientations.

(3) Detect the feature points in the images.

(4) Estimate the analytical solution of the camera parameters.

(5) Estimate the distortion parameters by solving a least-square problem.

(6) Refine all parameters by maximum likelihood estimation.

The advantage of this algorithm is that it does not require the exact motion of the calibration board. In our experiment, the calibration board is a chessboard pattern printed by the laser printer. The calibration procedure is carrying the board and moving it within the measurement volume. All of the intrinsic and extrinsic camera parameters could be recovered by analyzing the recorded images. It is quite flexible and robust.

However, it is difficult to give universal error estimation for the camera calibration algorithm. These errors are case-by-case problems, which are related to many factors including noise level, number of planes, orientation of the model plane, camera location, target volume, measurement depth, and so on.

A previous researcher (Zhang, 1999) used simulated images to estimate the performance of the algorithm. Define the exact coordinates of the feature point in the image as u_0 and v_0. Assume that its 3D real world coordinates are $\mathbf{X}$. Once we obtained the calibrated parameters, $\mathbf{X}$ could be easily projected into the image space. Assume that the projected feature point has the coordinates of u and v. Then the calibration error can be evaluated as:

$$\begin{aligned} e_x &= |u - u_0| \\ e_y &= |v - v_0| \end{aligned} \tag{4.8}$$

Zhang (1999) reported that the errors increase linearly with the noise level. He also examined the influence of the orientation of the model plane with respect to the image plane. Since it will be challenging to detect the location of the feature points if the model plane was nearly parallel to the image plane of the camera, the calibration error decreased as the angle between them increased. They reported that the best performance angle was about 45^o. A

particular important result they reported was the relationship between errors and the number model plane images. As Figure 4-8 shows, the performance of the algorithm is getting better if more images are used. However, it is not necessary to use more than ten images because their contribution decreased quickly at this range.

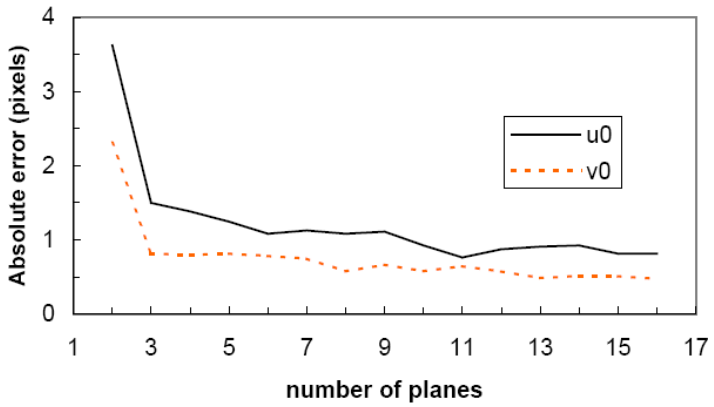


Figure 4-8. Calibration error varying with the number of model planes (Zhang, 1999).

We evaluated the calibration error for our particular application. Figure 4-9 shows a typical example of camera calibration conducted within an aircraft cabin mock-up. Details of the experimental set-up will be described in the next chapter. Figure 4-9 (a) shows one typical position of the calibration board. A total of twelve positions were adopted. Their locations and orientations were randomly selected within the target volume as shown in Figure 4-9 (b). The exact coordinate of the feature points on the model planes were obtained by a corner extracting algorithm. After all camera parameters were recovered, the feature points were back projected on the original image using these parameters and compared with the image processing result. The result is shown in Figure 4-10. The maximum error is about three pixels.

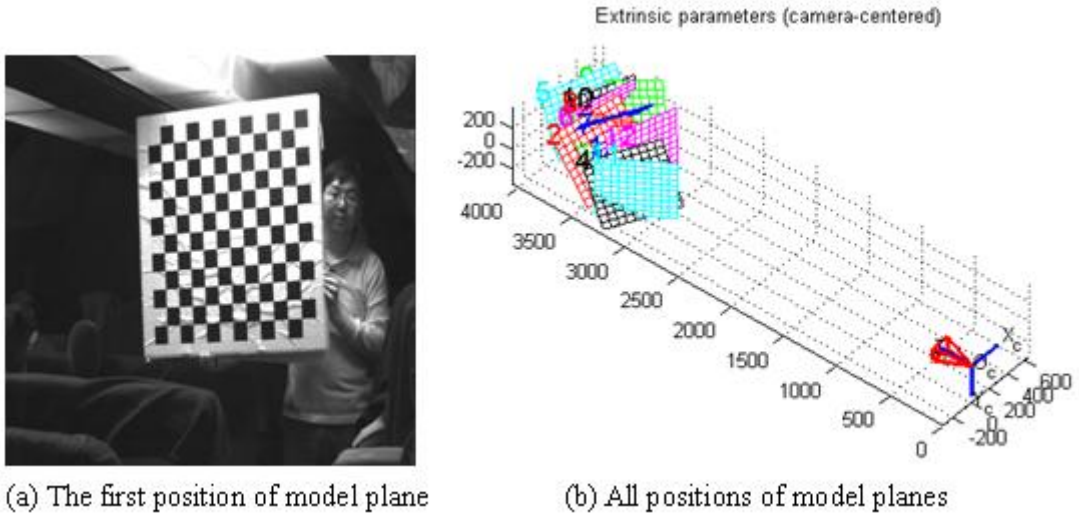


Figure 4-9. A typical example of camera calibration.

4.1.3 3D reconstruction algorithm

As discussed in Chapter 3, the real world coordinates of particles are obtained by solving an inverse projection problem. The governing equation of the linear system is Equation 3.15.

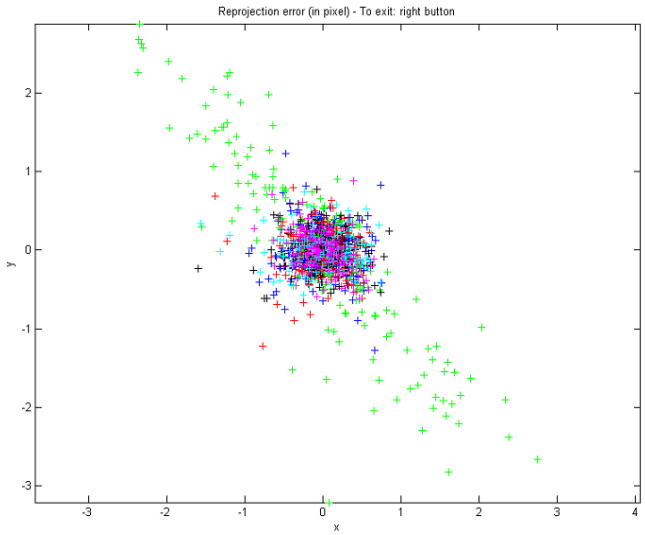


Figure 4-10. Camera calibration error analysis.

From Equations 3.8, 3.9 and 3.14, it is straightforward to establish the relationship between a normalized camera coordinate and the pixel coordinate:

$$x_{im} = -f_x x_{norm} + o_x$$
$$y_{im} = -\frac{f_x}{\alpha} y_{norm} + o_y \tag{4.9}$$

In the HVPT algorithm, the particle detection error Δx_{im}, Δy_{im} and the calibration errors of intrinsic camera parameters are inherited by x_{norm} and y_{norm}, which is shown in Equation 4.9. Rewrite Equation 3.15 in the following form:

$$\mathbf{A}\mathbf{X}_w = \mathbf{b} \tag{4.10}$$

where $\mathbf{A} = \begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \mathbf{A}_3 \end{bmatrix}$, and $\mathbf{b} = \begin{bmatrix} \mathbf{b}_1 \\ \mathbf{b}_2 \\ \mathbf{b}_3 \end{bmatrix}$.

$\mathbf{A}_i$ and $\mathbf{b}_i$ are the matrix of the i^{th} camera. They are determined by x_{norm}, y_{norm} and the extrinsic camera parameters. The extrinsic calibration error goes to both **A** and **b**.

$$\mathbf{A}_i = \begin{bmatrix} x_{norm}r_{31} - r_{11} & x_{norm}r_{32} - r_{12} & x_{norm}r_{33} - r_{13} \\ y_{norm}r_{31} - r_{21} & y_{norm}r_{32} - r_{22} & y_{norm}r_{33} - r_{23} \end{bmatrix}; \quad \mathbf{b}_i = \begin{bmatrix} T_x - x_{norm}T_z \\ T_y - y_{norm}T_z \end{bmatrix} \tag{4.11}$$

The problem is how these errors will affect the final result $\mathbf{X}_w$. Equation 4.10 is a typical linear least square problem. Let $\hat{\mathbf{X}}_w$ be the solution to minimize $\left\|\mathbf{A}\mathbf{X}_w - \mathbf{b}\right\|_2$. Assume that **A** is over determined and full rank, which is usually the case. If there is a small error in A and b noted by $\Delta\mathbf{A}$ and $\Delta\mathbf{b}$, define the following variables:

$$\varepsilon = \max\left[\frac{\left\|\Delta\mathbf{A}\right\|_2}{\left\|\mathbf{A}\right\|_2}, \frac{\left\|\Delta\mathbf{b}\right\|_2}{\left\|\mathbf{b}\right\|_2}\right]$$
$$\rho = \left\|\mathbf{A}\hat{\mathbf{X}}_w - \mathbf{b}\right\|_2 \tag{4.12}$$
$$\sin(\theta) = \frac{\rho}{\left\|b\right\|_2}$$

Then the error $\left\|\mathbf{X}_w - \hat{\mathbf{X}}_w\right\|_2$ is bounded by:

$$\frac{\left\|\mathbf{X}_w - \mathbf{X}_w\right\|}{\left\|\mathbf{X}_w\right\|} \le \left\{\tan(\theta)\left[con(\mathbf{A})\right]^2 + \frac{2con(\mathbf{A})}{\cos(\theta)}\right\}\varepsilon \tag{4.13}$$

Where con(A) is the condition number of **A**. It is defined as:

$$con(\mathbf{A}) = \frac{\sigma_{\max}(\mathbf{A})}{\sigma_{\min}(\mathbf{A})} \tag{4.14}$$

$\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ are the maximum and minimum Eigen-values of matrix **A** respectively. From Equation 4.13, it is observed that reconstruction error is theoretically related to the second order power of condition number of **A**. This condition number is determined by particle locations, intrinsic camera parameters and extrinsic camera parameters. To achieve high measurement accuracy, it is important to avoid the ill condition of matrix **A**. However, the character of matrix A cannot be arbitrarily set up before the experiment. The intrinsic parameters are pre-defined variables related to the CCD camera sensor. The only variables we can control during the experiment to optimize matrix **A** are the extrinsic parameters. The ideal camera set-up is to image the test section from three orthogonal directions (Sinha, 1988), because the orthogonal set-up produces the least correlation between cameras. In such configuration, the condition number of **A** will be close to 1.

In practice, it is usually inconvenient to study the accuracy with Equation 4.13. An alternative more straightforward error analysis approach is through back projection.

4.2 Algorithm validation based on synthetic images

The simulated images allow one to estimate the performance of the algorithm because the exact locations of the particles are precisely known before running the algorithm. Therefore, synthetic images are widely used as a validate approach in particle tracking algorithm development (Ishikawa et al., 2000; Pereira et al., 2006; Ruhnau et al., 2005).

In order to test the performance of the HPTV system, two groups of testing cases are designed. The first group will take the snap shots for the moving particles. The exposure time is set to be zero; therefore, all particle images are individual dots. The purpose is to validate the temporal tracking and hybrid spatial matching algorithm. The second group will take the particle

images given relatively long exposure times. Under this configuration, the images for high velocity particles will be short streaks. The purpose of these simulations is to validate the object hybrid algorithm.

All of the simulated particles are randomly released within a cubic volume. The total releasing number and the particle density are designed in advance. They all have square image shape and their locations are precisely recorded because we do not want to involve particle center detecting error in this algorithm validation. The particles move in the physical space through designed helix curves. The diameter and step size of these curves are pre-defined. The particle velocity is controlled by varying the helix revolution times. The whole simulation procedure consists of thirty frames at intervals of 0.0333s. The total simulated time is 1s.

The camera parameters used in the simulation are consistent with the experiment conducted within the aircraft cabin, which will be introduced in the next chapter.

4.2.1 Simulate particle images with zero exposure time

In this simulation, the purpose is to investigate the relationship between the performance of the algorithms and the particle density. Both the traditional three-way tracking algorithm PTV and the proposed spatial hybrid algorithm will be employed to track the particle trajectories. By comparing the tracking result, we will evaluate the performance of the HPTV algorithm.

Assume that the initial location of one particle is (x_0,y_0,z_0). The helix curve it moves through is designed as:

$$\begin{cases} x_i = x_0 - D/2 + \dfrac{D}{2}\cos(t_i) \\ y_i = y_0 + \dfrac{D}{2}\sin(t_i) \\ z_i = z_0 + \dfrac{s}{2\pi}t_i \\ t_i = \dfrac{n_r\pi}{15}i \end{cases} \tag{4.15}$$

Where

i - time step

D - diameter of the helix curve

t_i - time of the ith step

s - displacement of the helix center per revolution

n_r - total number of helix revolution

The parameters are set according to Table 4-1. A total of five cases are simulated. The number of released particles varies from 5 to 25 in these cases. Figure 4-11 and Figure 4-12 shows two typical cases with 5 and 25 released particles respectively. All the particles are released within a cubic space with the dimension of 100x100x100mm^3. They are rotating along the circles in the X-Y plane and moving in the Z direction at the same time. Figure 4-11 (d)-(f) and Figure 4-12 (d)-(f) display the particle images recorded by the cameras. A better visualization of the particle trajectories is shown in Figures 4-11, 4-12 where we have combined all frames onto one image. The particles are colored by their frame index from blue (beginning) to red (ending).

Table 4-1. Simulation parameters of synthetic images with zero exposure time.

Parameters	Values
Helix diameter	200mm
Displacement of the helix center per revolution	200mm
Total number of helix revolution	0.5
Total number of released particles	5, 10, 15, 20, 25

The first step of the algorithm is temporal tracking within each camera. The same particle at different frames are connected and constructed as 2D trajectories and plotted on the original images. The tracking results are shown in Figure 4-13.

It is observed that the algorithm performs better if the simulated case has a lower particle density. For example, in Figure 4-13 (a), (b) and (c), trajectories are completely detected, while in Figure 4-13 (d), (e), (f) many particles are missed. To quantitatively evaluate the difficulty of temporal tracking, we introduce a parameter "particle density rate." It is defined as follows:

$$r_{pd} = \frac{d_f}{d_d} \tag{4.16}$$

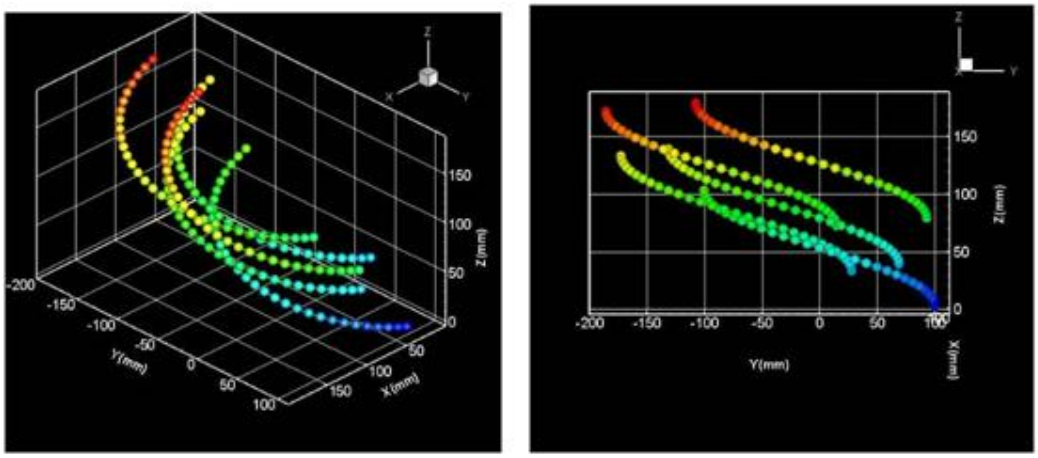
Where

r_{pd} - the particle density rate

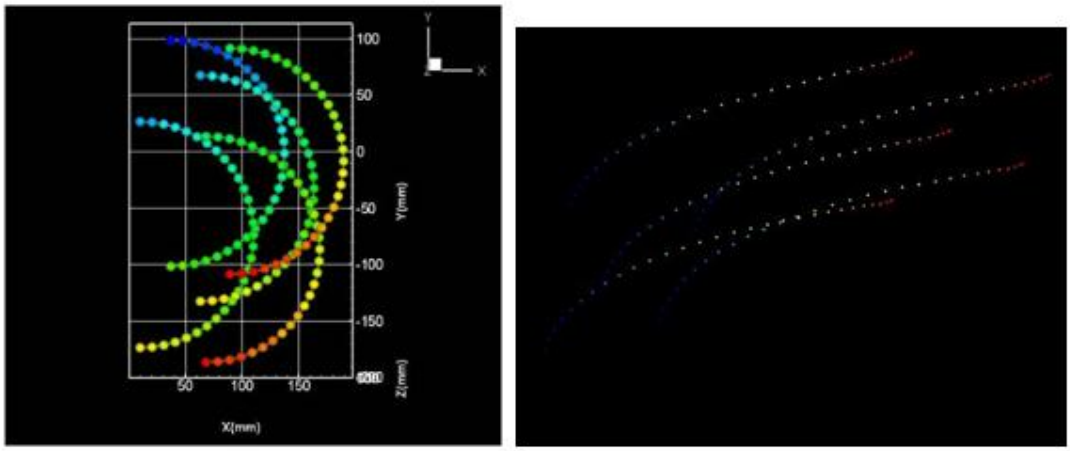
d_f - the average minimum distance between particles at each frame

d_d - the average particle displacement between frames

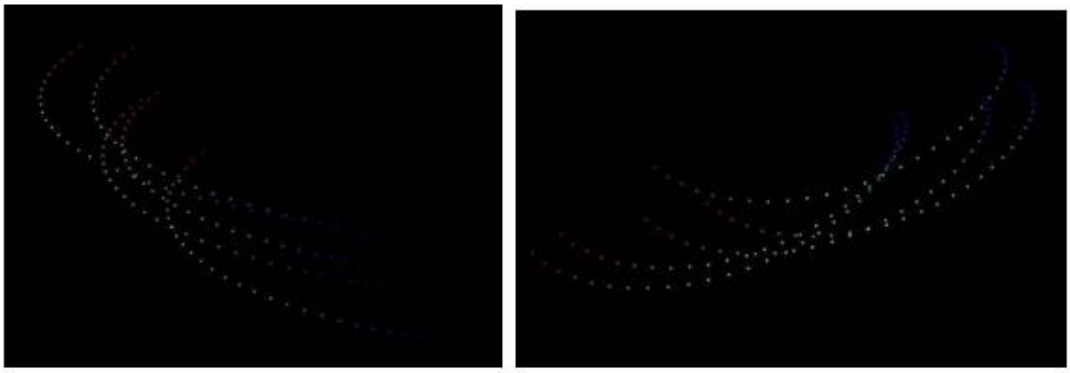
A large r_{pd} means the particles are widely separated. Their behavior is only taking place at their local region. No such intersection happened among particle trajectories. In practice, it is impossible to access d_d before tracking is completed. Therefore, we use d_c, the average minimum distance between particles at the image combined all frames, to approximate d_d. Figure 4-14 shows two extreme images with high and low r_{pd} respectively. Ten frames are combined in these cases. Assume that the particles are moving from left to the right. The solid line indicates the real trajectories. In the temporal tracking algorithm, we always search the candidate particles within a given radius. In $d_f >> d_c$, the trajectories are well isolated. For example, in Figure 4-14 (a), the distance between particles in the same frame (d_f) is much larger than the average minimum distance at the combined image (d_c). It is unlikely that the algorithm would mess up the trajectories. However, in Figure 4-14 (b), d_r and d_c are comparable for some particles. Searching particles within a given radius becomes quite challenging. It is highly possible that some particles belonging to the other trajectories are selected as the candidate of the next frame. For example, it will be hard to reject the fake trajectories l_1 and l_2 in Figure 4-14 (b) because d_c and $\sqrt{d_f^2 + d_c^2}$ is comparable. According to our previous experience, if the naked eyes can hardly distinguish the trajectories, it would be difficult for the computer as well.



(a) 3D view of designed helix curve (5 particles) (b) 2D view of designed curve (Y-Z plane)

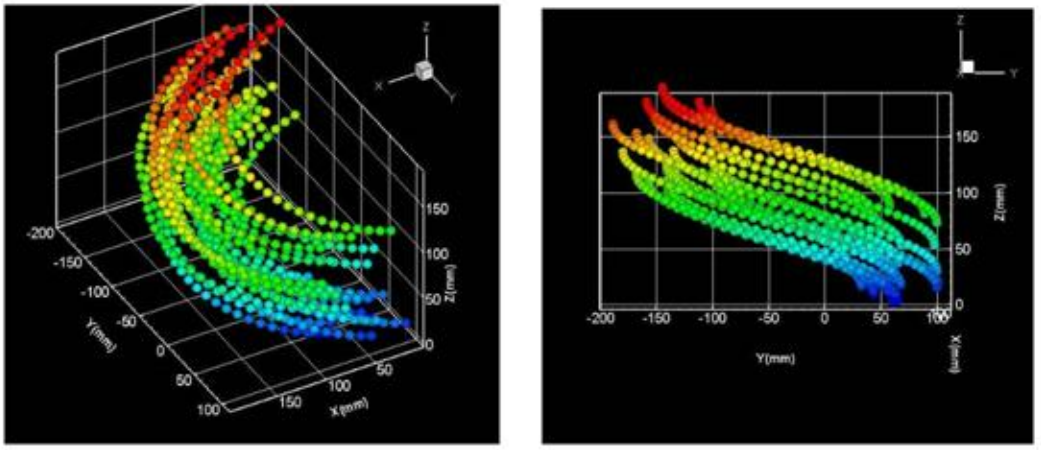


(c) 2D view of designed curve (X-Y plane) (d) Image combination of camera 1

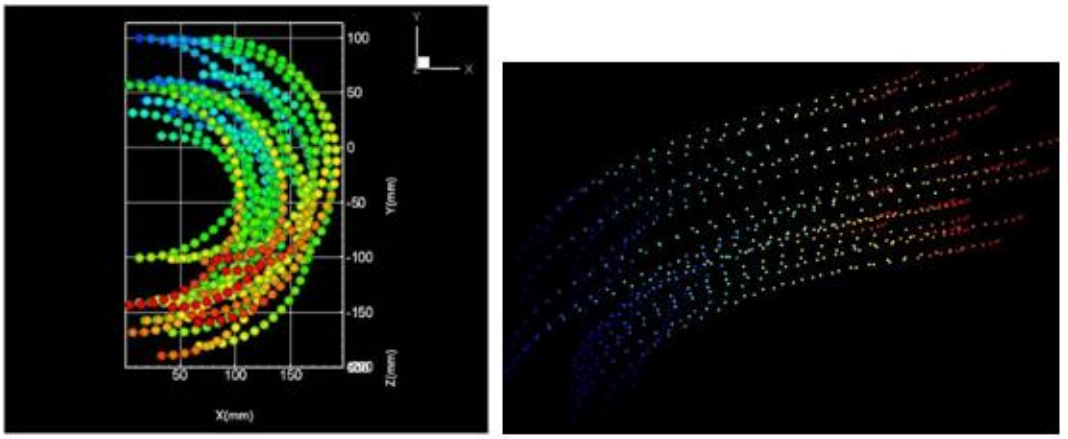


(e) Image combination of camera 2 (f) Image combination of camera 3

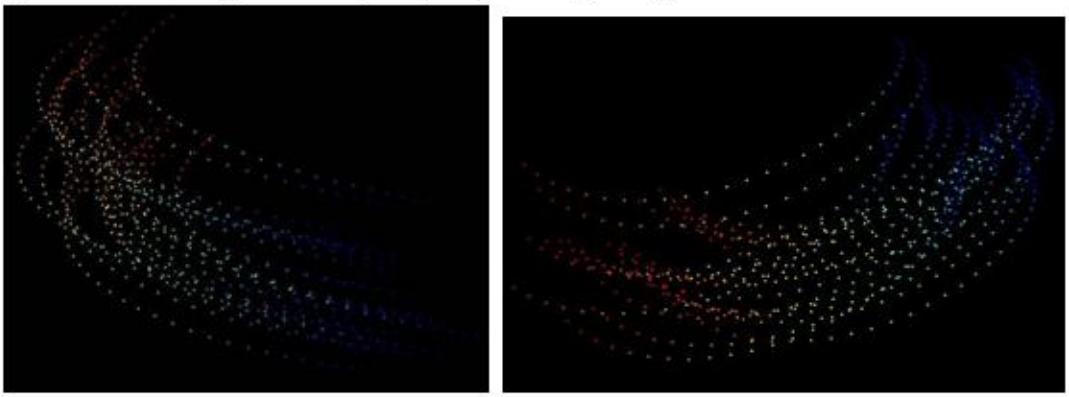
Figure 4-11. Simulated particle images with zero exposure time (5 particles).



(a) 3D view of designed helix curve (5 particles) (b) 2D view of designed curve (Y-Z plane)



(c) 2D view of designed curve (X-Y plane) (d) Image combination of camera 1



(e) Image combination of camera 2 (f) Image combination of camera 3

Figure 4-12. Simulated particle images with zero exposure time (25 particles).

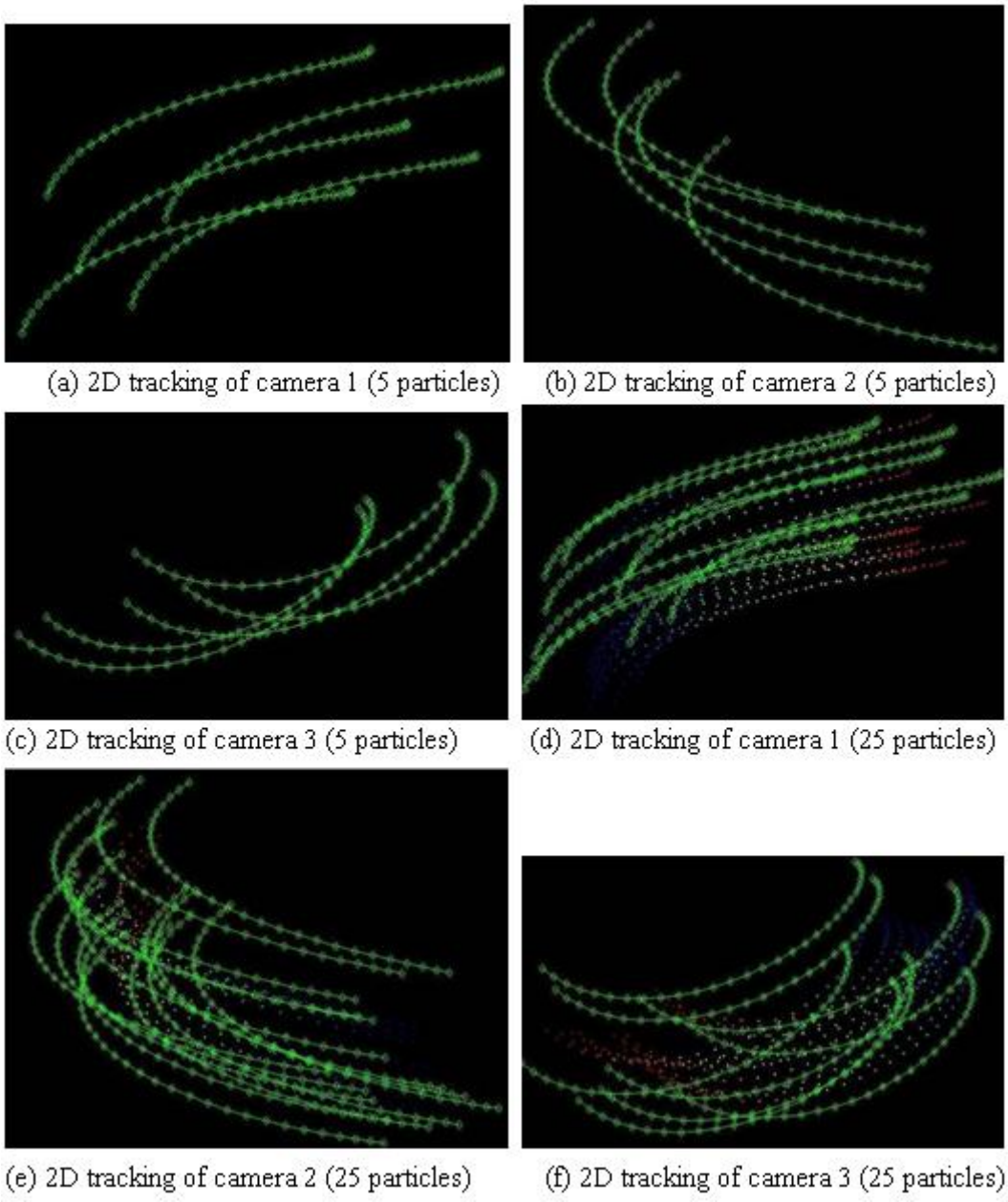


Figure 4-13. Temporal tracking of simulated particle images with zero exposure time (5 and 25 particles).

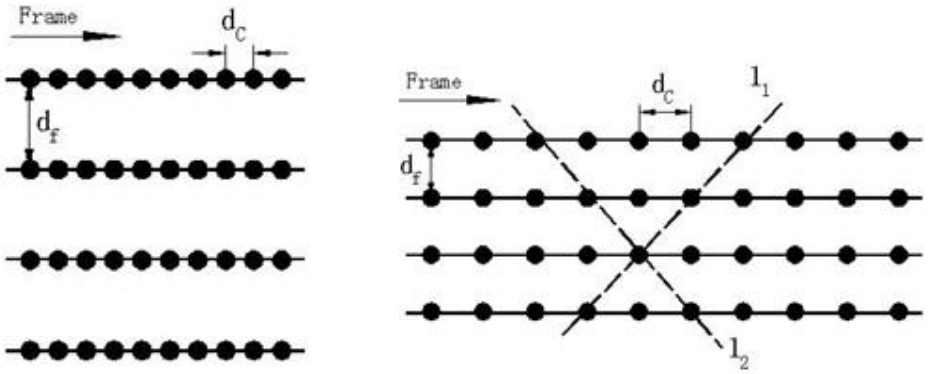


Figure 4-14. Compare trajectories with different r_{pd}.

Figure 4-15 shows two typical experimental examples. The second image has a lower r_{pd}, therefore, applying temporal tracking will be difficult. On the other hand, the trajectories in the left side image, with relatively high r_{pd}, are well isolated. A better algorithm performance could be expected.

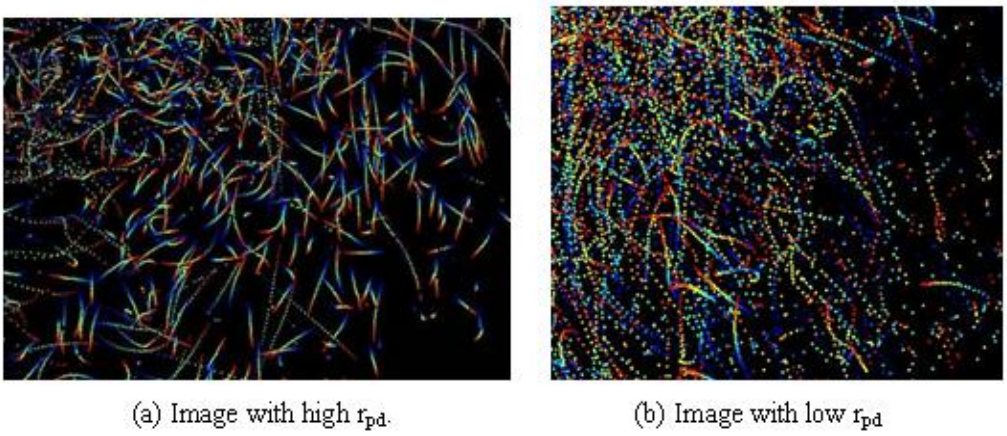
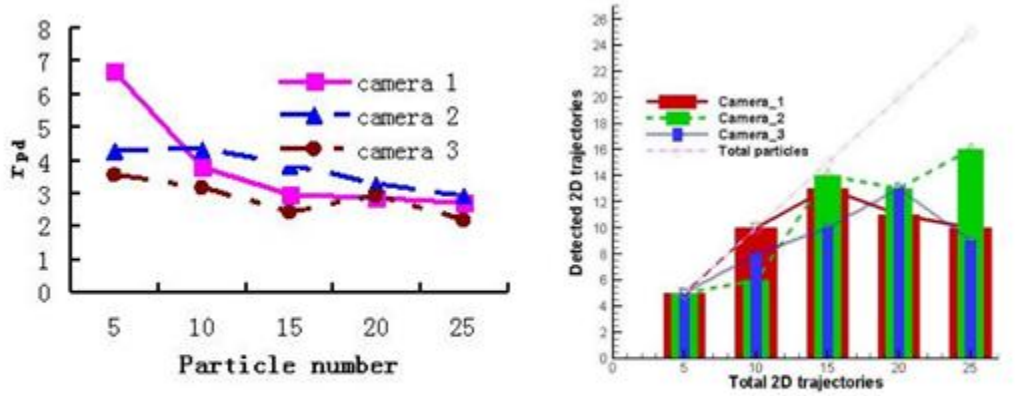


Figure 4-15. Compare experimental images with different r_{pd}.



(a) Relationship between r_{pd} and particle density. (b) 2D trajectories detecting rate and particle density

Figure 4-16. Evaluation of the temporal tracking algorithm.

For given measurement space, the number of released particles is a key factor affecting the detecting rate as well. Figure 4-16 (a) shows the relationship between r_{pd} and the released particle number. Increasing the particle number from 5 to 25, results in an r_{pd} decreased from 6.5 to less than 3 in camera 1. Similarly, the r_{pd} of camera 2 and 3 is decreased too. In response to that, the 2D trajectories detecting rate dropped accordingly. Figure 4-16 (b) investigated how the total number of released particles affected the detecting rate. The pink straight line indicated the 100 percent detecting rate. Due to the low r_{pd}, only half of the released particles are detected when the total particle number equals to 25. It is also worth noting that the detecting rate relies on the viewing direction as well. The performance of the algorithm is not equivalent in three cameras even if the same amount of particles were released.

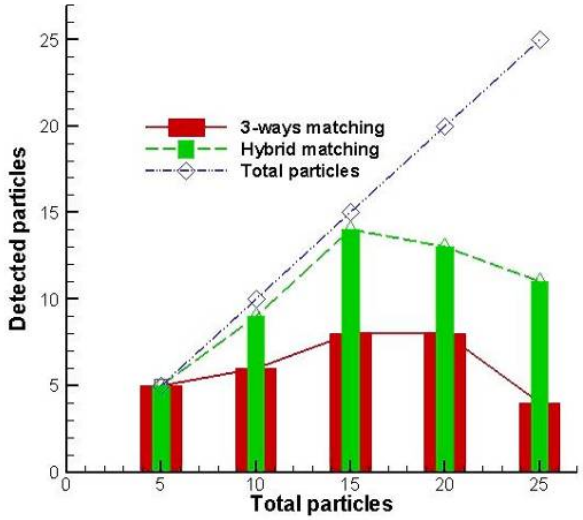


Figure 4-17. Comparing the performance of the hybrid spatial matching and the 3-way spatial matching algorithm.

After obtaining the 2D trajectories, the hybrid spatial matching and reconstruction approach is applied next. The detailed algorithms have been described in Chapter 3. In order to evaluate the performance of this algorithm, we compared the detecting rate using the new hybrid spatial matching approach and the traditional three-way matching approach. As described in Figure 4-17, the red columns indicated the detected particles by the three-way matching algorithm, while the green columns indicated those detected by the hybrid spatial matching algorithm. The straight line shows the 100% detecting rate. Since the hybrid algorithm will reconstruct the trajectories even if a particle is observed only by two cameras, its detecting rate is usually higher than the three-way matching approach. Overall, the detecting rate tends to decrease as more particles are released. When the total released particles achieved 15, even the curve indicating number of detected trajectories goes down. It means not only the detecting rate decreased, but also the detecting number decreased. Figure 4-18 directly compared the reconstructed 3D trajectories using these two approaches. Obviously, the hybrid matching algorithm shows more advantage.

4.2.2 Simulate particle images with given exposure time

Hybrid spatial matching algorithm contributes to broaden the measurement space; on the other aspect, hybrid object algorithm is aimed at extending the detectable velocity range. The simulation in this section is to investigate the performance of this algorithm. The simulation is conducted in three steps. First, we will show why increasing particle velocity will affect the performance of the algorithm and demonstrate that hybrid spatial object tracking is necessary. Then a pure streak tracking case will be simulated. In this case, only the high velocity particles are released and tracked. The streak tracking algorithm itself will be examined and validated. Finally, the whole HPTV algorithm, including both hybrid spatial matching and hybrid object tracking, is further validated. This simulation will mimic the real experimental images. Both low and high velocity particles will be released. Dot and streak shape particles will be tracked by their corresponding algorithms respectively to demonstrate the capability of the system.

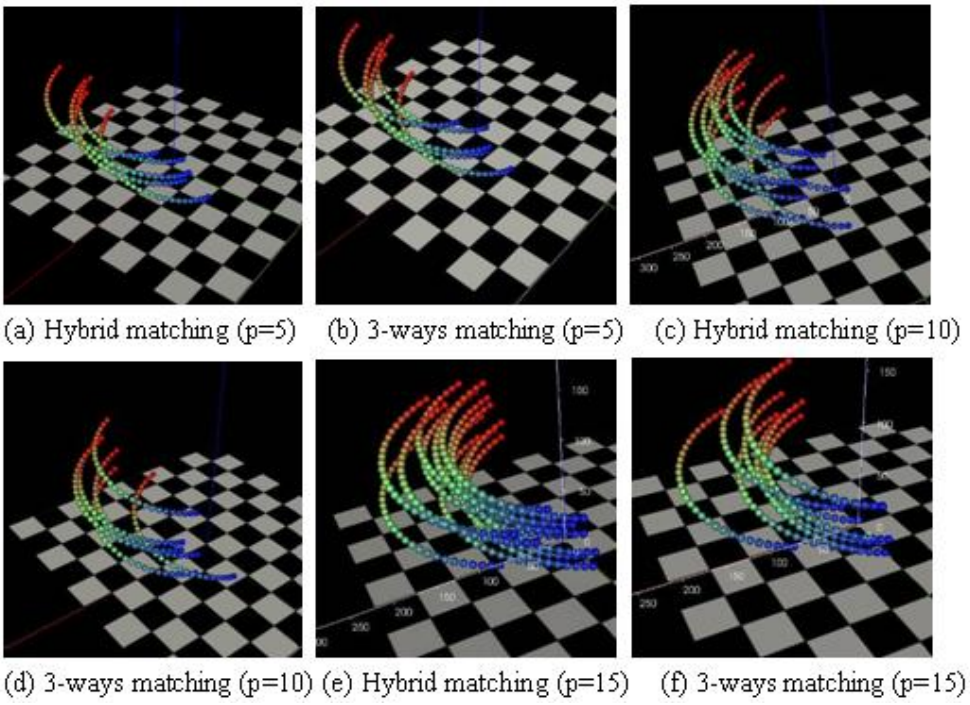


Figure 4-18. Compare the constructed 3D trajectories using the hybrid and 3-way matching algorithms (varying the number of released particles).

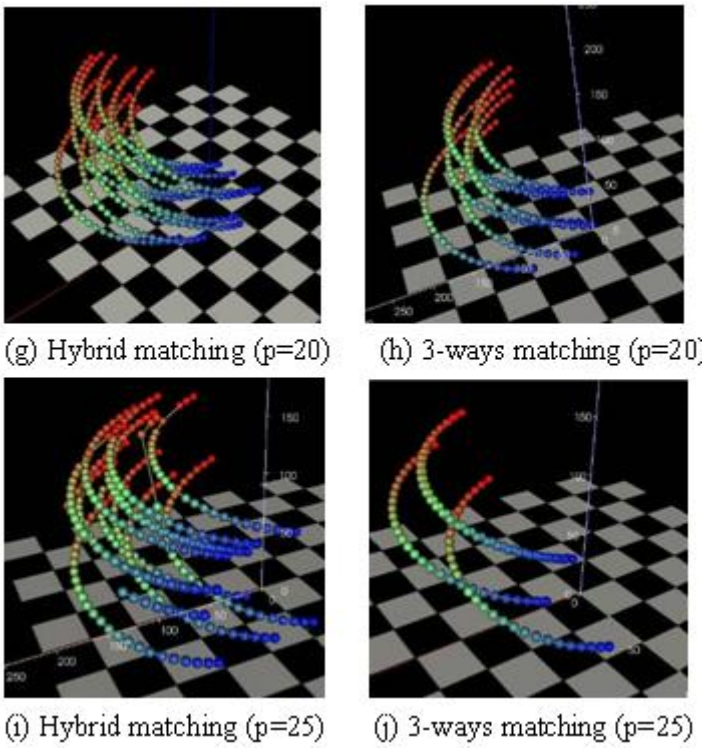


Figure 4-18. (cont.).

Case 1: Study the response of temporal tracking algorithm to particle velocity

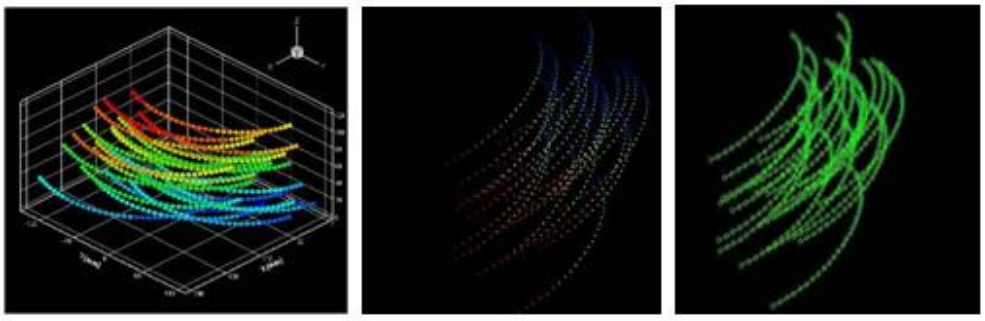
As we discussed in a previous section, r_{pd} is the criteria to evaluate the difficulty of conducting temporal tracking. A large r_{pd} means the trajectories are well isolated, and there is little interaction between particles producing ambiguity during temporal tracking. For a given camera configuration, frame rate and exposure time, r_{pd} is directly related to the particle density and particle velocity. The effect of particle density has been discussed previously. The effect of particle velocity will be discussed in this section.

The released particles still move along helix curves according to Equation 4.15. The velocity is controlled by the total number of helix revolutions n_r. Detailed simulation parameters are listed in Table 4-2.

Table 4-2. Simulation parameters of synthetic images with different particle velocity.

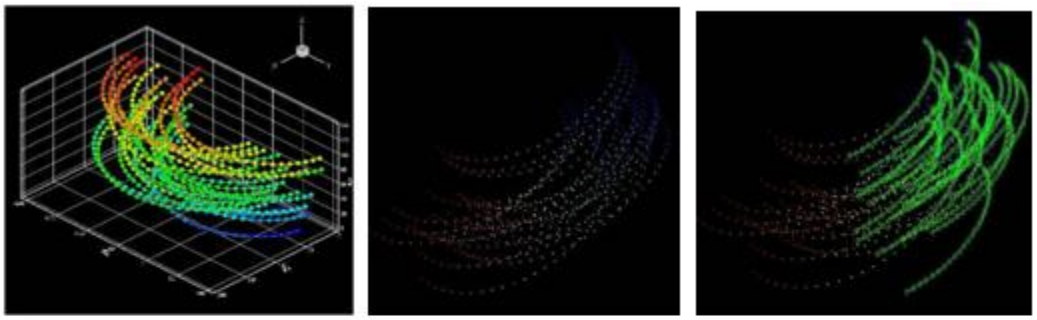
Parameters	Values
Helix diameter	200mm
Displacement of the helix center per revolution	200mm
Total number of helix revolution	0.3, 0.5, 0.7, 1
Total number of released particles	25

The simulated particle movement and the temporal tracking results are shown in Figure 4-19. The first case (n_r=0.3) has the lowest particle velocity. Frame displacement is much smaller than the average distance among particles; therefore, its r_{pd} is small. Nearly all particle trajectories are detected. Increasing the particle velocity results in the overlap and interaction among trajectories. In the second case (n_r=0.5), the particles images are messed up at the early frames due to the high particle velocity. The tracking would be more difficult if the velocity was further increased. For example, in the third and fourth case (n_r=0.7, n_r=1), a lot of trajectories are missed and some broken trajectory pieces are detected (Figure 4-19 (c), (d)). Figure 4-20 further demonstrated this conclusion. It shows the relationship between the detecting rate and the helix revolution times which is equivalent to the particle velocity. According to this figure, high particle velocity will significantly affect the detecting rate. For example, in camera 3 the detecting rate dropped nearly 50% as n_r increased from 0.3 to 0.7. Although the number of detecting trajectories seems to increase again at n_r=1, this is only because more broken pieces are captured. In fact, the number of detected particles is less than the low velocity cases.

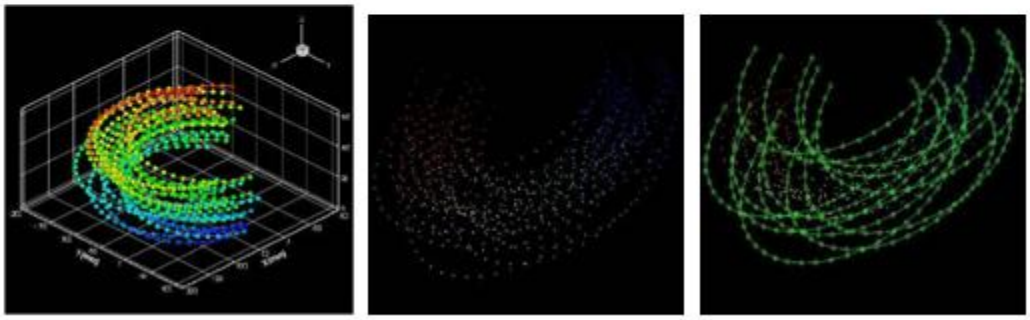


(a) Case 1: n_r=0.3

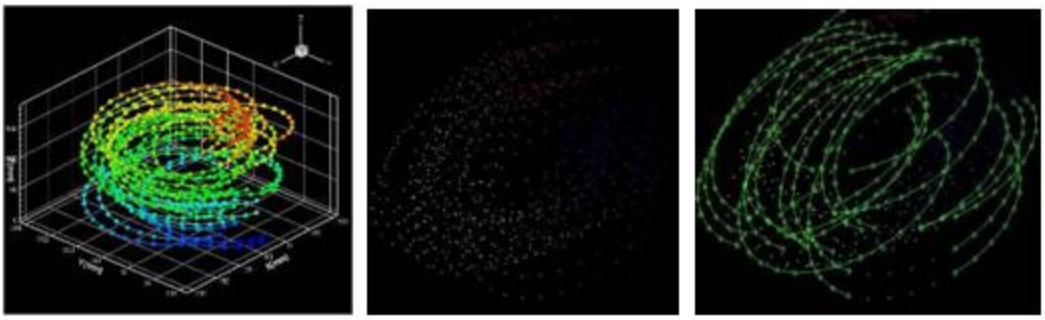
Figure 4-19. Designed curves, simulated images, and the 2D tracking result (varying particle velocity).



(b) Case 2: n_r=0.5



(c) Case 3: n_r=0.7



(d) Case 4: n_r=1

Figure 4-19. (cont.).

From these simulations and discussion, we concluded that the traditional PTV algorithm is not a proper choice for high velocity particle tracking. It is necessary to develop an alternative approach to overcome this limitation. The hybrid object tracking algorithm is proposed to fulfill this demand.

Case 2: Pure streak simulation and tracking

In previous simulations, it is assumed that the exposure time was zero. All images are instant snap shots of the moving particles. However, in reality the exposure time of CCD camera

is always a finite time instead of an instance. In order to mimic this situation, we modified the simulation strategy. Let's assume the frame rate is still 30fps and the exposure time is 1/60s. First, estimate the maximum displacement of particles between frames. Denote this displacement as d_f. For the current CCD camera we used, the exposure time is half of the frame rate. The particle displacement within the exposure time is approximately $d_f/2$. Therefore, the streak length taken during this period should be the image of this short 3D trajectory with $d_f/2$ length. Equation 4.15 is still used as the designed curve particle moving along. Unlike what is described in section 4.2.1, in which only one dot was the plot for one particle at each frame, in the current simulation the whole trajectory will be projected and plotted to construct the whole streaks. The coordinate of the streak on the image is calculated based on the camera models. The simulation parameters are shown in Table 4-3. The designed particle movement curves and simulated images are shown in Figure 4-21. In these images all particles have the same velocity. The different length of the streaks is because of the observing direction of the cameras.

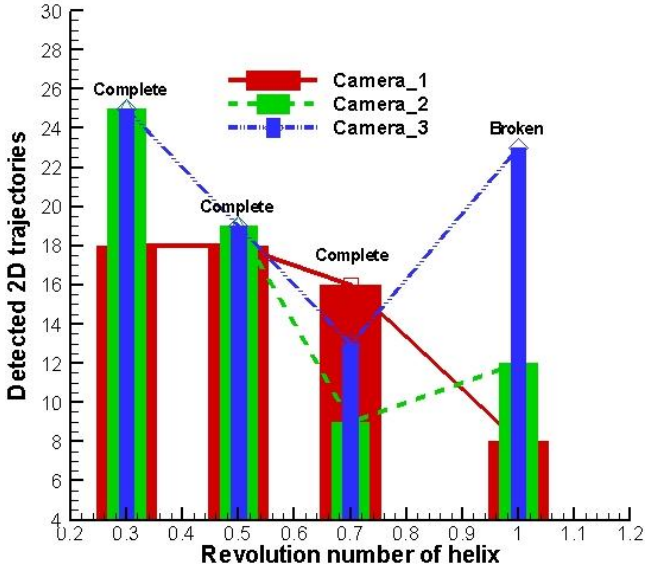


Figure 4-20. The relationship between the detected trajectories and the helix revolution times.

After applying the streaks tracking and reconstruction algorithm, the 3D streaks of the particles are obtained. The tracking result is shown in Figure 4-22. Figure 4-22 (a) visualized the

constructed 3D streaks over a reference chess board. Figure 4-22 (b-d) combined the simulated images of all frames to indicate the 2D particle trajectories. As we discussed in previous chapters, the epi-polar constrain is not a satisfied but a necessary requirement of particle matching. It is dangerous to perform streak matching and construction based on only two cameras. In the current algorithm, the three camera matching strategy was applied. Comparing the raw streak images and the back projected images, we found that they agreed well with each other. However, in Figure 4-21 (b-d) and Figure 4-22 (b-d), some streaks observed in camera 1 and camera 3 are missed in the final result. Re-examining the streaks in camera 2, we noticed that these streaks are out of the field view of that camera. That explains exactly why these streaks are not recovered in the final result. Through this pure streak tracking study, we are confident that this algorithm could be safely combined with the dot tracking part to constitute the whole HPTV system.

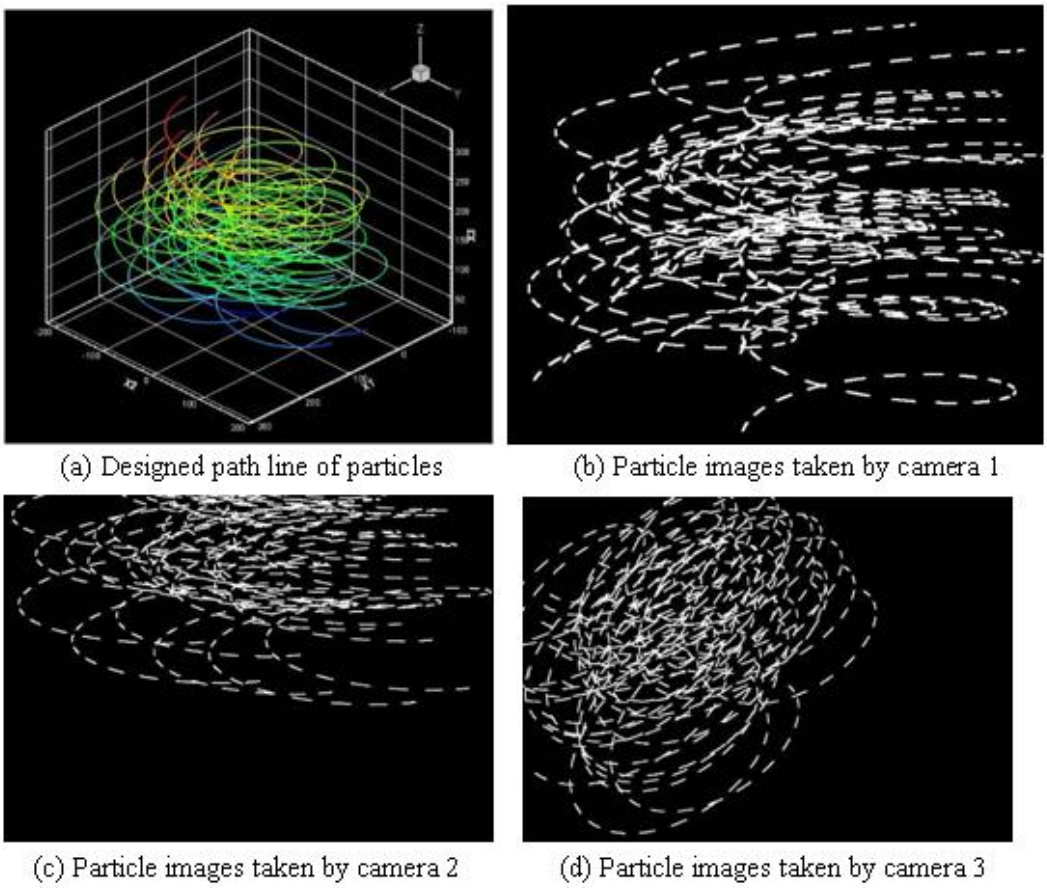


Figure 4-21. Pure streaks simulation.

Table 4-3. Simulation parameters of synthetic images of pure streaks.

Parameters	Values
Helix diameter	200mm
Displacement of the helix center per revolution	80mm
Total number of helix revolution	1.5
Total number of released particles	25
Total pixels per streak	24

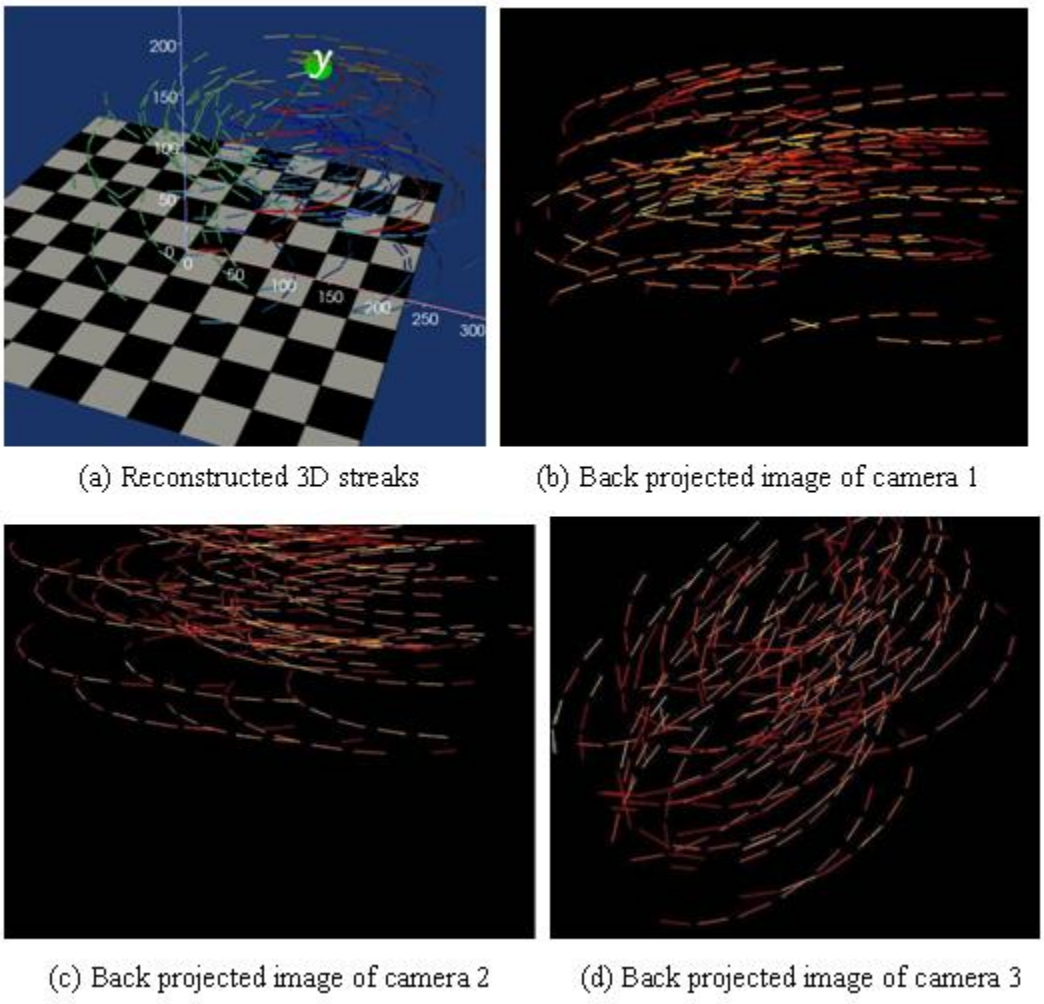


Figure 4-22. Pure streaks reconstruction and back projection.

Case 3: Hybrid spatial matching and objects tracking

A practical application of the HPTV system will track both the high and low velocity particles simultaneously. For example, the airflow field within the aircraft cabin generated by the passenger's coughing could be regarded as an impinging jet flow. Using the HPTV system to track the seeding bubbles carried by this flow would be quite challenging. The initial peak cough velocity could be up to 20m/s. It decays quickly along both spatial and temporal dimension, and eventually the bubbles will follow the airflow pattern produced by the ventilation system. In order to capture such airflow field with large velocity variance, it is necessary to use the hybrid tracking approach proposed in previous section.

A particle acceleration scene is designed to mimic the real application. The simulated particles still move along the helix curves as previous cases. Assume the tracking time is one second. At the initial 2/3 second, the particles are moving at constant low speed. Their angle speed in x-y plane is ω_0. After that, they accelerate at an acceleration rate α. The total number of helix revolutions is n_r. The displacement in z direction for each revolution is s. Again, we assume the frame rate of the camera is 30fps and the exposure time is 1/60s. 24 pixels are projected during each exposure time so as to ensure the continuity of the streaks. The governing equations for the simulation are Equation 4.17. The simulation parameters are listed in Table 4-4. The simulated curves and the produced images are shown in Figure 4-23.

$$\begin{cases} \alpha = \dfrac{2(2\pi n_r - \omega_0)}{(1-2/3)^2} \\ \theta_k = \omega_0 t_k \qquad (t_k \le 2/3) \\ \theta_k = \omega_0 t_k + \dfrac{\alpha}{2}(t_k - 2/3)^2 \qquad (t_k > 2/3) \\ x_k = x_0 - \dfrac{D}{2} + \dfrac{D}{2}\cos(\theta_k) \\ y_k = y_0 + \dfrac{D}{2}\sin(\theta_k) \end{cases} \tag{4.17}$$

As shown in Figure 4-23 (a), 25 particles are randomly released within a 200x200x200mm^3 cubic volume. At the initial stage of their movement, their images are dot

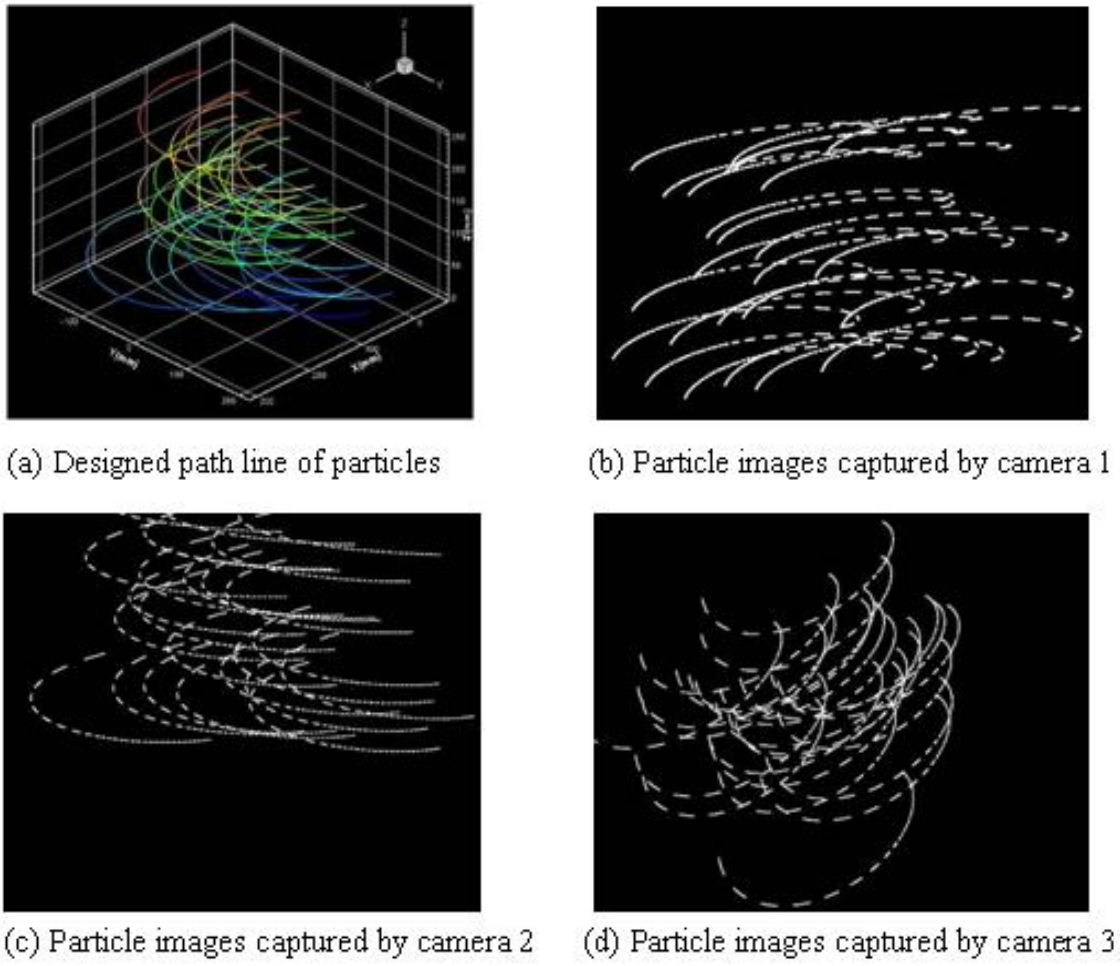
shape due to their low velocity. Figure 4-23 (b-c) illustrated how this dot shape was stretched into short streaks by increasing their velocities. But it is worth clarifying that the length of streaks is not only determined by the velocity, it is also related to the camera configuration. For example, in Figure 4-23 (c), the streaks length is very short as they move toward or away from the camera lens. Figure 4-24 shows the tracking result after applying the hybrid particle tracking algorithm.

Table 4-4. Simulation parameters of synthetic images for hybrid object tracking.

Parameters	Values
Helix diameter	200mm
Displacement of the helix center per revolution	80mm
Total number of helix revolution	0.6
Total number of released particles	25
Total pixels per streak	24
Initial angle speed	0.5π

By comparing Figure 4-24 (a) and Figure 4-23 (a), it is observed that the detected 3D dot and streaks well captured the flow characterized of the released particles. A traditional PTV or pure streak tracking algorithm could only recover half of the simulated curves because of lacking a hybrid tracking capability. The current algorithm significantly increased the detectable particle number and velocity range of the system.

The tracking result is validated in Figure 4-24 (b-d) by projected back to three cameras views. These images could be compared with the simulated images in Figure 4-23 (b-d). They are consistent with each other and demonstrated the reliability of the system. With this validation, this system can be safely applied to the full-scale indoor air flow study conducted within the aircraft cabin mock-up.



(a) Designed path line of particles

(b) Particle images captured by camera 1

(c) Particle images captured by camera 2

(d) Particle images captured by camera 3

Figure 4-23. Synthetic images for hybrid objects tracking.

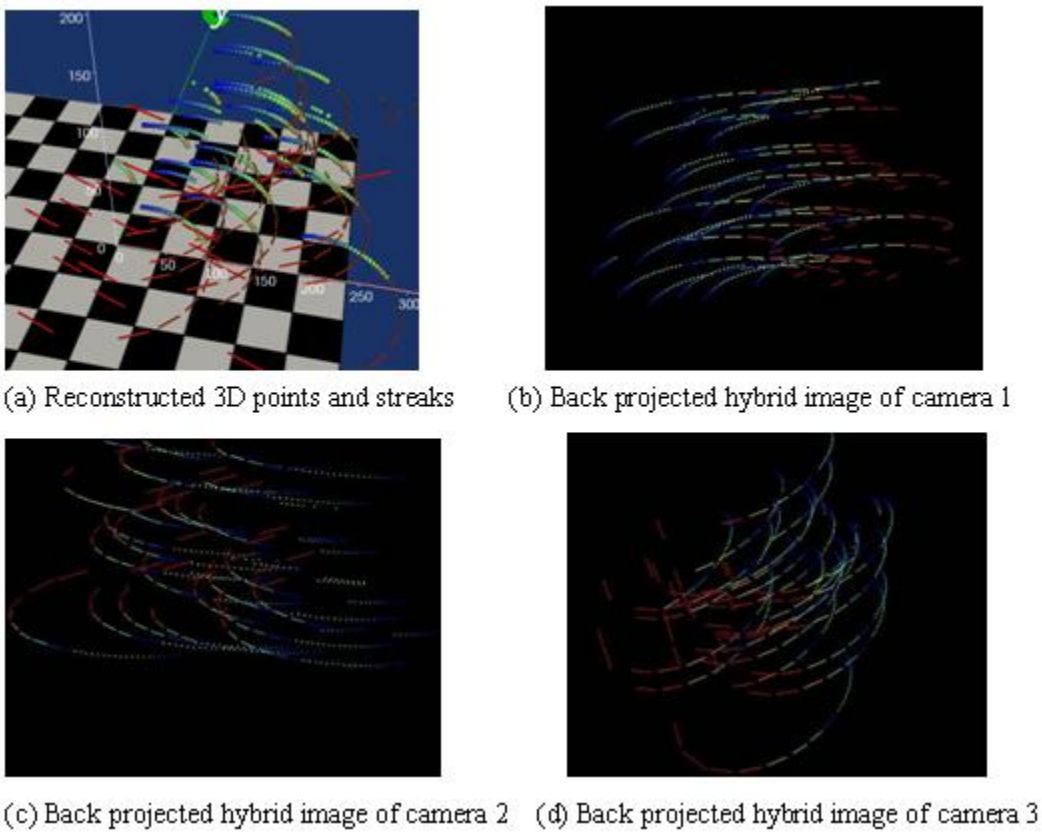


Figure 4-24. Hybrid objects reconstruction and the back projection images.

4.3 Summary

In this chapter, synthetic images with known trajectories are adopted to analyze the error of HPTV system. With the known exact locations of the simulated particles, the tracking results are compared with the "right answer." Each sub-algorithm, including image processing, camera calibration, hybrid spatial matching, and hybrid object tracking, is discussed separately in this chapter.

The semi-synthetic image, plotting the simulated particles onto the real experimental scene, was used to evaluate the performance of the image processing algorithm. It has been proven that it is more representative than the approaches using pure black image as the background. Based on the simulation, it is observed that the detecting rate increased as we

increase the particle intensity or particle size. Usually the detecting errors of particle locations are smaller than 2 pixels.

A camera calibration toolbox was used to analyze the camera calibration errors. The constructed feature points on the calibration board were projected back to each camera view, and compared with the "exact location" obtained by corner extraction to evaluate the errors. According to the projecting result, we found that the errors caused by camera calibration are less than 3 pixels. Most of the them are close to 1 pixel. Based on the stereo vision model, we also, theoretically, derived the equations that control the reconstruction error.

After that, we validated the performance of the tracking algorithm by tracking three particle moving cases. First, we tested the response of the algorithm to the particle density and velocity and explained why hybrid spatial and object tracking are necessary. Then the synthetic images that mimic high velocity and accelerating particles are produced and tested by the algorithm. Through validation, we found that the reconstructed results agree well with the simulated images, and demonstrated the measurement capability of the HPTV system.

CHAPTER 5 EXPERIMENTAL DESIGN AND PROCEDURE DEVELOPMENT

5.1 Introduction

Many complaints about indoor air quality in aircraft cabins could be solved by improving thermal comfort and controlling airborne pollutants (Yan et al., 2009). Airflow pattern created by an air distribution system is one of the key factors affecting the thermal environment and pollutant dispersion at the occupied zone (Bosbach et al., 2006; Jacobs and De Gids, 2006; Sandberg, 2007; Wang, 2006; Yan et al., 2009; Zhang and Chen, 2007; Zhang et al., 2009). However, limited by current experimental capability, little reliable experimental data has been obtained. This situation hindered both the experimental and numerical modeling research communities because there is no widely accepted data for model validation. The goal of current research is to fulfill this demand. The HPTV system is capable of consecutive tracking of the particle movement at high frequency (up to 30 Hz), and the dimension of the measurement space can be several meters, which makes it a suitable equipment for an indoor airflow study.

The experiment will be conducted within a full-scale aircraft cabin mock-up. Several objectives are expected to be achieved: (1) demonstrate the measurement capability of the HPTV system; (2) develop a measurement procedure for using the HPTV system; and (3) acquire reliable experimental data for numerical model validation. The experiment will be conducted at two levels: (1) investigate the global airflow pattern generated by the ventilation system at the whole cabin level and (2) investigate the airflow pattern generated by a passenger's sneezing/coughing at the local level. The airflow is visualized by seeding neutral buoyant helium filled bubbles into the airspace. These bubbles are simultaneously observed from arbitrary viewing directions by three CCD cameras. The consecutive sets of images will be recorded at a specified sampling frequency. The HPTV algorithm will be adopted to process these images to reconstruct both the 3D trajectories and streaks. The particle velocity is obtained by multiplying the displacement between frame and the frequency. This velocity is regarded as the local fluid

velocity. A basic measurement system must consist of four components: the chamber with ventilation system, bubble generator, illumination system, and image accruing system as shown in Figure 5-1.

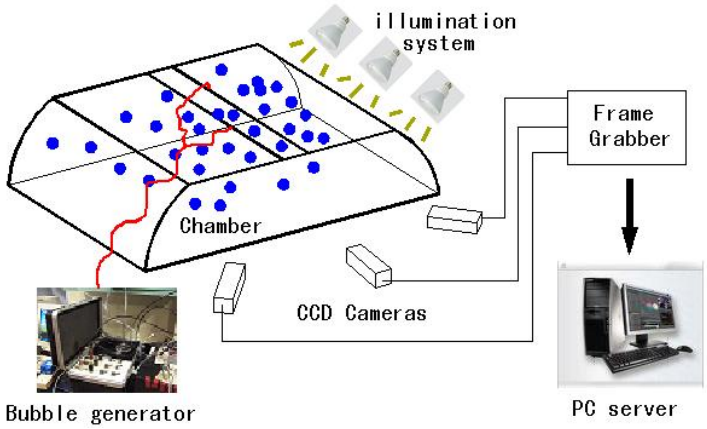


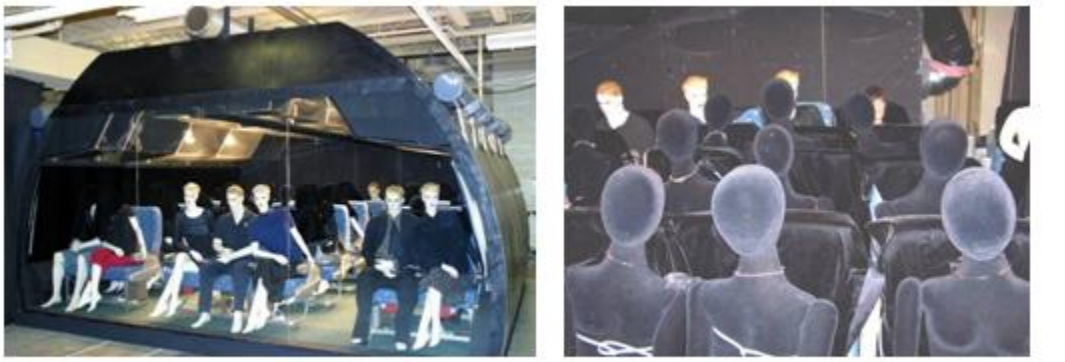
Figure 5-1. Sketch of the experimental setup.

5.2 Experimental system

5.2.1 Aircraft cabin and ventilation system

The experimental chamber is a full-scale Boeing 767-300 aircraft cabin mock-up section simulator, which was funded by the Centers for Disease Control and Prevention (CDC) and the Boeing Company in a previous project. It consists of five rows of 35 accurately configured aircraft seats. Full size manikins are placed on these seats to simulate passengers. All manikins are dressed in typical clothing. The configuration of this facility is shown in Figure 5-2. The dimension of the chamber is $4.4 \times 4.2 \times 2 m^3$.

The mainframe and walls are wooden structures. The ceiling is made of transparent plastic for illumination purposes. To minimize the light reflection, most surfaces are painted black or covered by black fabric. The front wall is built with plastic glass to allow installing CCD cameras from outside.



(a) Front view of the aircraft cabin (b) Manikins and seats covered by black fabric

Figure 5-2. The full-scale aircraft cabin facilities.

The ventilation system is designed to produce the same airflow pattern as the Boeing 767-300 aircraft cabin. The fresh air is supplied by a long narrow overhead diffuser and exhausted though the grilles located at the bottom of the side walls. As shown in Figure 5-3 (a), the airflow is forced by an air delivery system installed outside the cabin. The air delivery system is capable of cooling down the cabin interior temperature to 5°C in several minutes to simulate cruise conditions. The fresh air is delivered by a duct connected to the distribution nozzles installed above the ceiling and run along the cabin depth, which is shown in Figure 5-3 (b). The control panel is installed outside the cabin. The fan speed is displayed on the panel in real time.

(a) Fan installed behind the cabin (b) Duct and nozzle above the ceiling

Figure 5-3. HVAC system of the aircraft cabin.

5.2.2 Illumination system

A laser is usually used as the light source for the traditional PTV system (Doh et al., 2004; Hassan et al., 1992; Khalighi and Lee, 1989; Nishino et al., 2000; Suzuki and Kasagi, 2000). It is selected based on two facts: (1) the seeding materials are very tiny particles. The visibility of the particles is proportional to d_p^2 (Hering et al., 1997). Without a powerful light source, it would be difficult to visualize them. (2) For some high speed CCD cameras, the exposure time could be very short as 0.1-100ns. Usual light sources are either not powerful enough or not stable enough to illuminate the particles.

However, using a laser as the light source for an indoor airflow study is difficult considering its large measurement volume. Jones et al. (1997) estimated that a 10 W laser is capable of illuminating an area on the order of 1 m^2. Suppose the illumination depth of this estimation was 1 cm, then to illuminate a target volume with a dimension of 2 m × 2 m × 1.5 m, a laser with power up to 6000W is required. Handling such a powerful laser in the lab is extremely difficult. According to the literature review, a laser has not been used in any large scale PTV study. Instead, sometimes it is used in large scale PIV studies which primarily are two-dimension illumination. Recently, a researcher (Ma et al., 2009) reported a PIV study within the upper deck of an A380 aircraft cabin. They have used an Nd-YAG double oscillator laser system with a pulse energy of 350 mJ/pulse. The thickness of their light sheet is controlled between 2 and 5 cm in the measurement region.

For a large scale PTV study, a popular seeding material is the neutrally buoyant helium filled soap bubbles. The diameter of the bubbles range from several hundreds of micrometers to a few millimeters (Ma et al., 2009; Sun et al., 2005). These bubbles provided a much higher light scattering efficiency compared to the tiny fluorescent particles or oil droplets. They could be illuminated by continuous light sources such as spot lamps, projector lamps or arc lamps. For example, Querzoli (1996) illuminated the non-buoyant particles at 200μm diameter with a 1000W arc lamp. Mo et al. (2003) used a multiple-wave length flash lamp to illuminate the tracer

particles at 15 to 25μm. Muller et al. (1997) used spot lamps to illuminate the helium filled soap bubbles ranging from 3 to 7mm.

The adopted illumination system in the current experiment is similar to our previous set-up (Sun et al., 2005). Totally 36 incandescent spot lights are installed above the ceiling of the middle three-row section. Each bulb had a capacity of 120W. As shown in Figure 5-4 (a), these bulbs are wired into six groups and connected to the switches on the outside wall of the cabin. Two copper lightings are put in front of the cabin to provide the additional light source. Air around the bulbs is circulated with an exhaust fan to cool the light. The copper lights are controlled manually with a switch beside the computer. They are turned on only for a few seconds when the CCD cameras are recording. Therefore, their contribution to heat gain of the cabin is negligible.

(a) Spot lamps on above ceiling (b) Additional copper lighting in front of the cabin

Figure 5-4. Illumination equipment for the experiment.

5.2.3 Seeding particle dynamics and the bubble generator

The choice of seeding particles is a compromise between short response time and high visibility. In general, we have several requirements for seeding materials: (1) neutral density, (2) short relaxation time, (3) high light scattering efficiency, (4) uniform size, and (5) long lifetime.

For large scale airflow field measurement, helium filled soap bubbles are regarded as one of the most proper seeding materials (Kessler and Leith, 1991; Ma et al., 2009; Muller et al., 1997; Sun et al., 2005; Suzuki and Kasagi, 2000; Zhang et al., 2005; Zhao et al., 1999).

The motion of the seeding particles is controlled by the particle shape and the forces applied to them. The shape of the particle will significantly affect the relaxation time and the dragging force. For helium filled bubbles, they have approximated spherical shape; therefore, their shape effect is neglected. In the following discussion, we will assume that the external forces (gravitational, centrifugal and electrostatic) are negligible. We won't consider the interaction between particles either. Under these assumptions, the governing equation of a suspended bubble in a continuous medium reads as (Melling, 1997):

$$\begin{cases} \dfrac{\pi d_p^3}{6}\rho_p \dfrac{d\vec{U}_p}{dt} = -3\pi\mu d_p \vec{V} + \dfrac{\pi d_p^3}{6}\rho_f \dfrac{d\vec{U}_f}{dt} - \dfrac{1}{2}\dfrac{\pi d_p^3}{6}\rho_f \dfrac{d\vec{V}}{dt} \\ \qquad\qquad + \dfrac{\pi d_p^3}{6}\left(\rho_f - \rho_d\right)g - \dfrac{3}{2}d_p^2\left(\pi\mu\rho_f\right)^{1/2}\displaystyle\int_{t_0}^{t}\frac{d\vec{V}}{d\xi}\frac{d\xi}{\left(t-\xi\right)^{1/2}} \quad (1) \\ \vec{V} = \vec{U}_p - \vec{U}_f \quad (2) \end{cases} \tag{5.1}$$

In Equation 5.1(1), the first item is the acceleration of the particle. The viscous resistance according to Stokes's Law is defined in the second item. The third item describes the pressure gradient in the vicinity of the particle produced by the acceleration of the fluid. The fourth term represents the resistance of an inviscid fluid to the acceleration of the sphere. The buoyancy effect is considered in the fifth term. And the last term is the Basset history integral defining the resistance caused by the unsteadiness of the fluid. Equation 5.1(2) defined the relative velocity between particle and fluid.

For indoor airflow study, the bubble size is usually larger than the smallest turbulence eddies. Therefore, we do not expect the bubbles to respond to the turbulent fluctuation. We further assume that the far-field velocities are steady-state, and the bubbles are exactly neutral buoyant. Under these assumptions, we have:

$$\rho_f \approx \rho_p \quad \text{and} \quad \frac{d\vec{U}_f}{dt} \approx 0 \tag{5.2}$$

To simplify the problem, the final term in equation 5.1(1) will not be considered either. Equation 5.1 could be rewritten as:

$$\frac{d\vec{U}_p}{dt} = \frac{-12\mu}{\rho_p d_p^2}(\vec{U}_p - \vec{U}_f) \tag{5.3}$$

If the initial velocity of the bubble is zero, then the integrating Equation 5.3, we could obtain:

$$\begin{cases} \left|\dfrac{\vec{U}_p - \vec{U}_f}{\vec{U}_f}\right| = e^{-t/\tau} \\ \tau = \dfrac{\rho_p d_p^2}{12\mu} \end{cases} \tag{5.4}$$

τ is referred to as the relaxation time of the particle. It indicates the time required for a particle to follow the air flow pattern. Figure 5-5 plotted the relationship between the relaxation time and the bubble diameter. Clearly, smaller bubbles are preferred if they are visible because they have a shorter relaxation time.

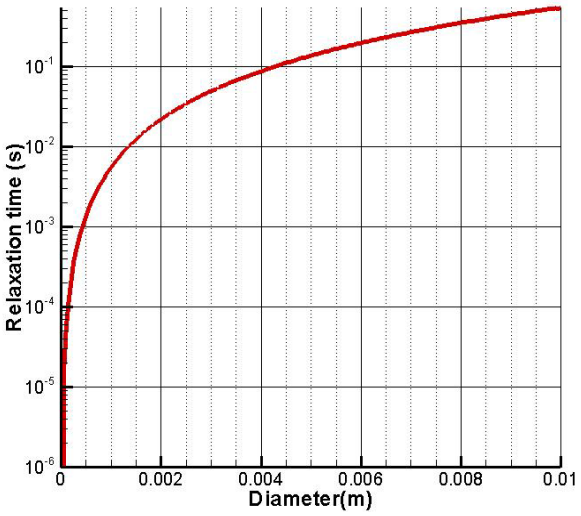
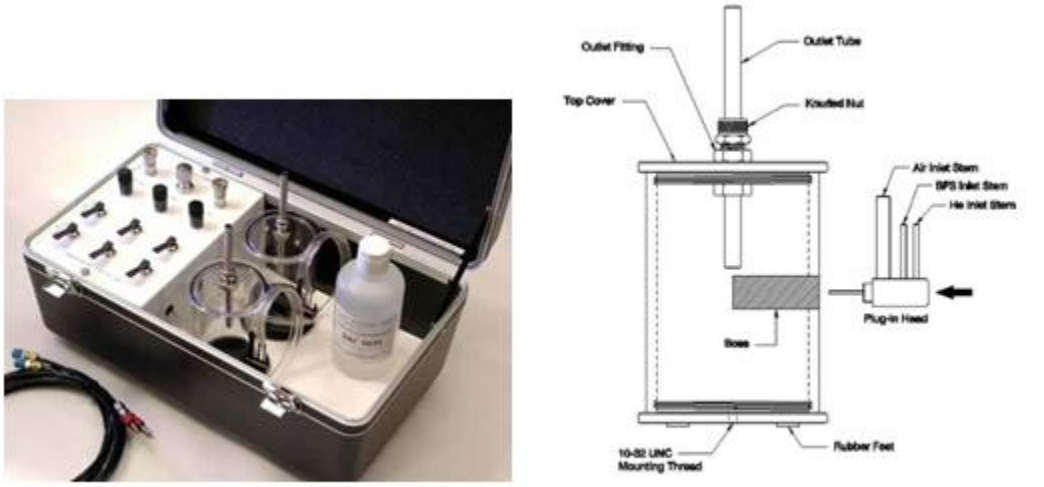


Figure 5-5. The relaxation time of bubbles with different diameters.

In current research, a commercial SAI model-5 bubble generator with two heads made by Sage Action Inc (Ithaca, NY) is adopted. The system is shown in Figure 5-6. The "heart" of the

system is a needle shaped plug-in head. It consists of two co-axial tubes with different diameters. The bubble fluid solution is pumped through the outer pipe while the helium is blown through the central duct. Helium filled bubbles are generated at the end of the needle. A much larger, concentric jet of air, in turn, blows the bubbles continuously into the bubble separating tank. Some of the bubbles are too heavy/light to follow the airflow and regarded as "bad" bubbles. The bubble separating tank will remove these "bad" bubbles through a cyclonic motion created by the compressed airflow from the needle. The centrifugal forces will separate all non-neutral bubbles. The parameter of the bubble generator is listed in Table 5-1.



(a)The whole bubble generator (b) Sketch of the bubble separating tank

Figure 5-6. The particle seeding system.

Table 5-1. Technical parameters of the bubble generator.

Technical parameters	Value
Bubble Generation Rate	300 - 400 bps / Head
Bubble Diameter	1.27 – 3.81mm
Bubble Life	1 - 2 min
Helium Flow Rate	200 ccm / Head
Nominal Air Flow Rate	1.2 cfm / Head

According to Figure 5-5, the relaxation time of the bubbles produced by this generator is approximately $5x10^{-3}$~$8x10^{-2}$s. In practice, the bubble generator is usually shut down for 20 seconds before taking pictures in order to eliminate the effect of the jet flow releasing bubbles.

This time is much longer than the relaxation time; therefore, we assume that all bubbles are following the airflow pattern when recording pictures.

5.2.4 Image acquisition system

The image acquisition system consists of three "TMC – 1402 CL" CCD cameras (JAI PULNIX Inc., San Jose, CA), two Grablink Expert 2 Frame Grabbers (Euresys Inc., San Jose, CA), and a host computer (Pentium IV). The optical sensor of the camera is a 1/2" progressive scan interlines transfer CCD. The active area is 6.47x4.84mm^2 with 1392(H)x1040(V) pixels. The frame rate is 30Hz or 15Hz under full resolution. Higher frame rate could be obtained if it is working on partial scanning mode. The camera lens we chose is Megapixel Fixed Focal Length Lenses (Edmund Optics Inc, Barrington, NJ). Detailed technical parameters are listed in Table 5-2.

Table 5-2. Technical parameters of the camera lens.

Technical parameters	Value
Focal Length FL (mm)	12
Maximum Sensor Format	2/3"
Angular Field of View, 1/2" Sensor (°)	30.00
Minimum Working Distance (mm)	150.00

For a given camera sensor and lens, the field of view (FOV) could be easily obtained through some simple calculations. In the current experimental configuration, the working distance ranges from 1.5m to 4m. The corresponding FOV ranges from 0.8x0.6 m^2 to 2.13x1.6m^2.

To successfully apply the HPTV system, it is important to trigger three camera shutters simultaneously, which is called camera synchronization. The trigger signals are sent out by the hosting computer through frame grabbers. Each grabber could hook up two cameras; therefore, two grabbers are required in the current experiment set-up. A monitoring clock is used to validate the status of the synchronization. The clock pointer will rotate 360^o within one second. During

this time, thirty images are taken by each camera. The position of the clock pointer taken by different cameras is compared. Typical validation images are shown in Figure 5-7.

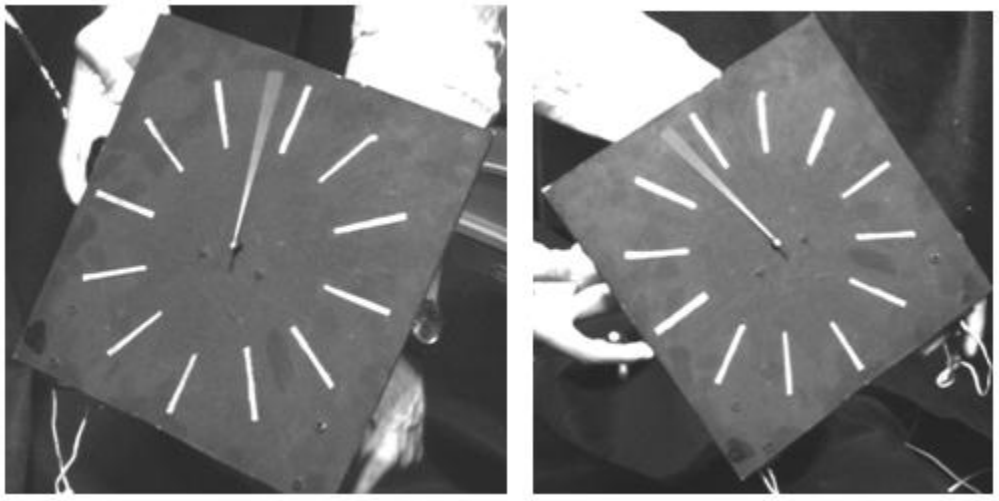


Figure 5-7. Verifing the synchronization of cameras.

5.3 Experimental design and procedure development

5.3.1 Experimental design

As discussed in section 5.1, the experiments are conducted at two levels: global level and local level.

For the global level study, we will investigate the airflow pattern produced by the air-conditioning system of the cabin. Figure 5-8 illustrates the proposed experimental set-up. Three tripod-mounted CCD cameras are placed in front of the cabin. The ventilation system is started at least 30 minutes prior to image acquisition to allow the system to achieve a steady-state (Sun et al., 2005). The seeded bubbles are injected into the cabin through the ceiling diffuser before the image recording starts. The system will run on three different ventilation rates: 816 m^3/h (479.5 cfm), 1052 m^3/h (618.18 cfm), and 1259 m^3/h (739.8 cfm), corresponding to 80%, 100% and 120% of the full ventilation load of the operating condition, respectively. The size of

the inlet diffuser is 4.37mX0.07m. Assume that the inlet velocity is uniformly distributed; therefore the Reynolds numbers at the inlet diffusers are 2590, 2870 and 3920 respectively. The tracking time of the current study is one second and the frame rate is set to be 30fps. A total of 90 images will be recorded for each case. The exposure time of cameras should be chosen according to the illumination condition and the particle velocity. In the current study, it is set to 1/120s.

For the local environmental study, we will investigate the airflow pattern after the coughing/sneezing of a passenger. Figure 5-9 indicates the position of the coughing passenger and the experimental set-up at the target volume. A nozzle is installed on the head of the manikin to mimic the mouth. Helium filled bubbles are released from that nozzle. Three cameras are installed nearly perpendicular with each other to capture the bubbles at the local environment after they are released. The duration of measurement is one second. Ninety images will be recorded by three cameras. Both streaks and dot shape particles will be captured and processed by the HVPT algorithm. Similar to the global environmental study, the experiment will be conducted at three different ventilation rates: 816 m^3/h (479.5 cfm), 1052 m^3/h (618.18 cfm), and 1259 m^3/h (739.8 cfm) corresponding to 80%, 100% and 120% of a full load. The air velocity at the nozzle is 7.72m/s. The corresponding Reynolds number is about 30880. Table 5-3 provides a summary of these experimental parameters.

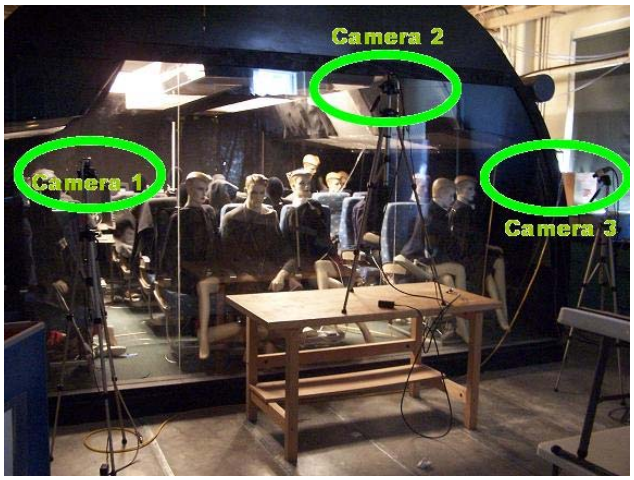
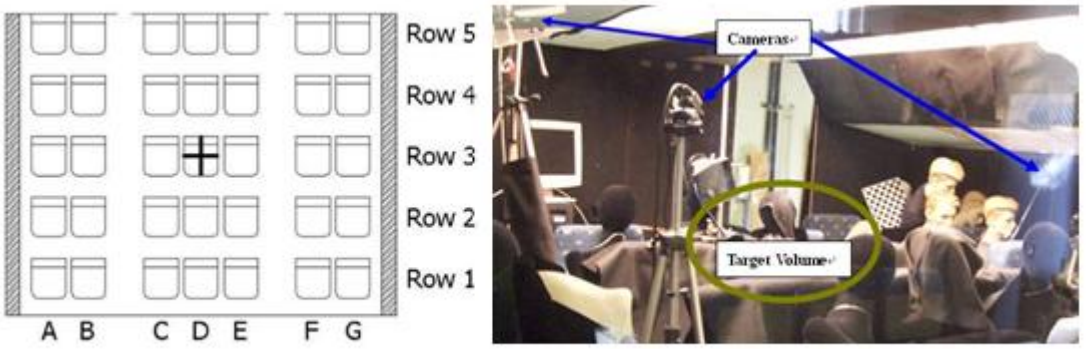


Figure 5-8. Experimental setup for the air cabin global study.



(a) The location of index passenger (b) The experimental setup toward the target volume

Figure 5-9. Experimental setup for local environmental study.

Table 5-3. Summary of the testing cases.

Test Case	Target level	Ventilation rate	Frame rate	Exposure time
Case 1	Global	816 m^3/h	30fps	1/120s
Case 2	Global	1052 m^3/h	30fps	1/120s
Case 3	Global	1259 m^3/h	30fps	1/120s
Case 4	Local	816 m^3/h	30fps	1/120s
Case 5	Local	1052 m^3/h	30fps	1/120s
Case 6	Local	1259 m^3/h	30fps	1/120s

5.3.2 Experimental procedure development

The following steps are scheduled to conduct as a complete experimental procedure: (1) Camera calibrations; (2) Camera position optimization; (3) Error estimation; (4) 2D particle tracking; (5) 3D particle reconstruction; and (6) Data visualization and post-processing.

5.3.2.1 Camera calibrations

Camera calibration is the first step of the HPTV measurement. The purpose is to recover the intrinsic and extrinsic parameters. The calibration target for global airflow measurement is a grid paper similar to a black and while chess board as shown in Figure 5-10. It has 10 squares in both horizontal and vertical directions. The side length of each square equals to 30mm. A smaller board with grid size equals to 15mm was used for local environment measurement. Its grid number is set to be 12X8 to accustom with the smaller space.

(a) Calibration board in global environment. (2) Calibration board in local environment

Figure 5-10. The calibration board for global and local environment measurement respectively.

An open-source camera calibration toolbox is adopted to process the calibration (Zhang, 1999). As described in section 4.1.2, twelve arbitrary board positions are selected to take images. The first eleven positions are used to recover the camera parameters while the last position is to define the world coordinate for particle tracking. As shown in Figure 5-10, the X-Y plane defined the calibration board. X-axis is a vertical downward direction, while the Y-axis is a horizontal rightward direction. According to the right hand rule, the Z-axis is pointing out of the board.

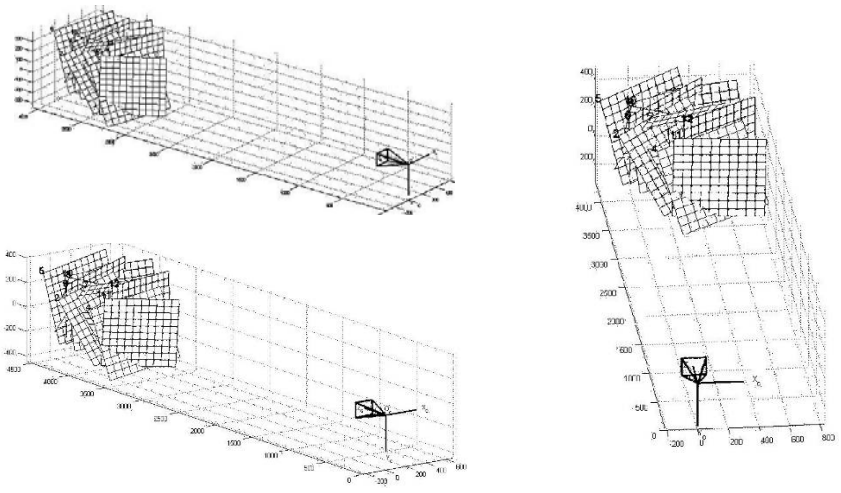


Figure 5-11. Calibration position of global level experiment.

Figure 5-11 and 5.12 show the selected calibration board positions for the global level and local level experiment. These positions covered most of the measurement region of the experiments.

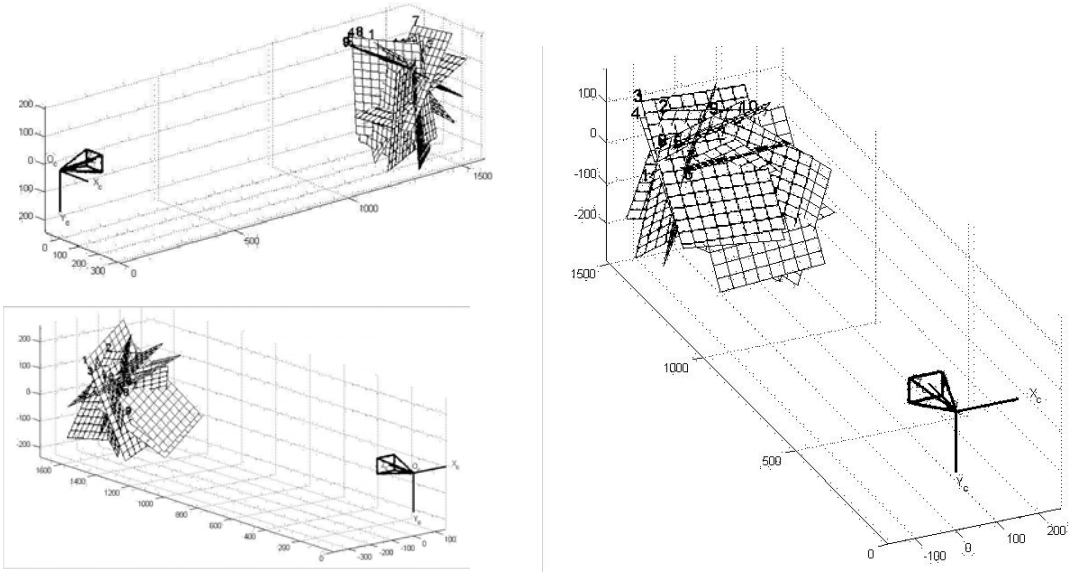


Figure 5-12. Calibration position of local level experiment.

5.3.2.2 Camera position optimization

Ideally, we want three cameras to be perpendicular with each other in order to minimize the reconstruction errors. However, the configuration of cameras is limited by many factors, including chamber construction, hardware communication, illumination, and target volume, etc. A trial and error analysis is usually necessary to obtain the optimized positions.

A lot of attention is also paid to maximize the detectable volume besides the accuracy consideration. Orthogonal camera setup has the smallest reconstruction error, but it also has the smallest detectable volume. A larger detectable volume is always preferred without sacrificing the accuracy. In our experiment, a common view analysis algorithm has been developed to estimate the detectable volume for a given camera configuration. We divide the physical domain

into small grids. For each grid, the location is represented by its central coordinates. The grid value is set to be 0, 1, 2 or 3. We obtain the images for these grids by projecting them onto three cameras based on the calibrated camera parameters. Assume the produced image coordinate of a given grid is (X,Y). If (X,Y) is within the resolution range of the CCD camera, in our case the resolution is 1392X1040, it is defined located in the field view of this camera. The grid value is set to be 0 if it is located in none of the three cameras. Its value is set to be 1 if it is located only in one camera, and similarly it could be set as 2 or 3. The result is visualized as Figure 5-13.

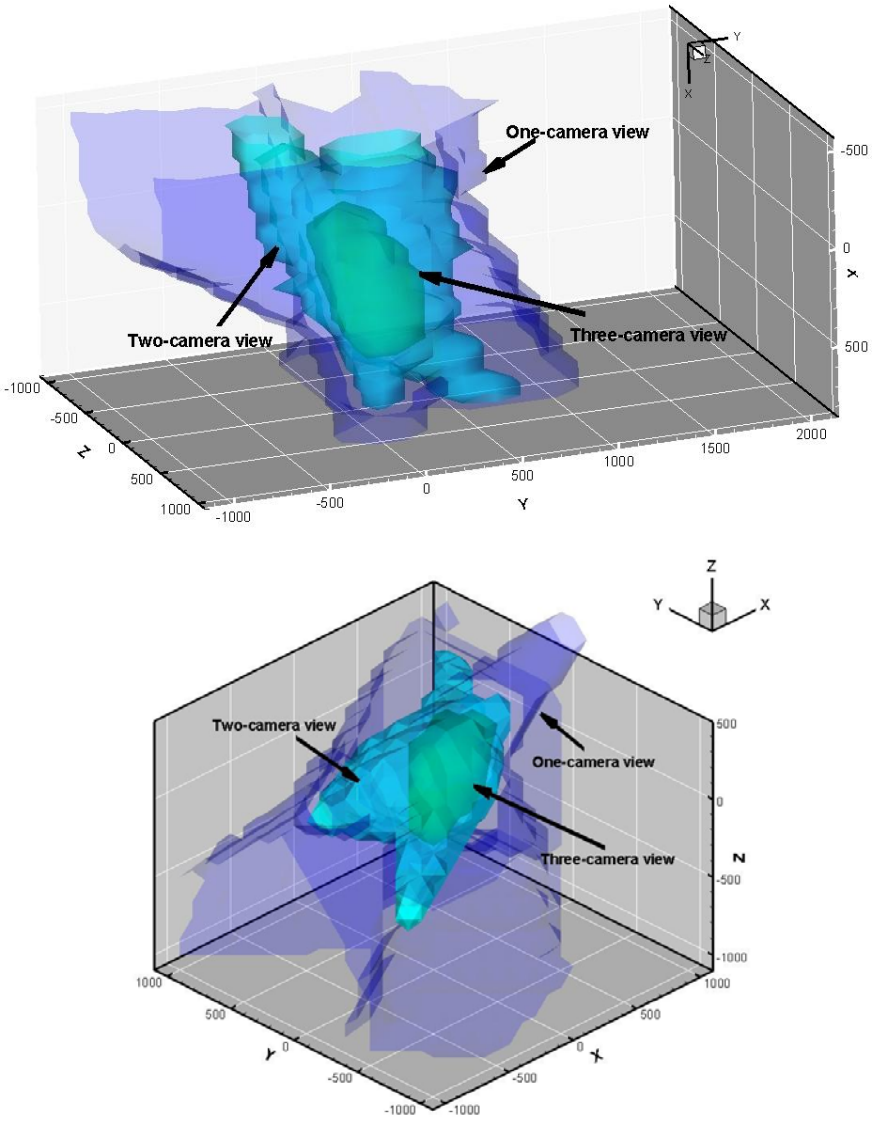


Figure 5-13. The common camera view of global (upper) and local (lower) level measurement.

5.3.2.3 Error estimation

As mentioned in section 4.1.3, the reconstruction error is associated with many factors including particle location, camera calibration error, and camera configuration (extrinsic parameter), etc. Therefore, there is no universal criterion to describe the accuracy of a HPTV system. However, it is possible to estimate the error of the system based on statistical analysis. The error analysis procedure is proposed as following:

(1) Randomly release particles within the target volume. In current research, a total of about 2,000 particles are released.
(2) Project these particles onto three cameras. Neglect the particles beyond the field of view of all cameras.
(3) Add normal distributed noise onto these image coordinates. The noise is designed consistent with the realistic error discussed in section 4.1.1.
(4) Reconstruct the 3D coordinate of the particles based on two-way and three-way matching algorithm respectively.
(5) Evaluate the difference between the reconstructed coordinates of the particles and their exact coordinates.

In this proposed approach the whole HPTV system is treated as a black box. Impose well defined noise as the input signals of the system. These signals are manipulated or amplified by the system. Since the noise is comparable with the errors in practical applications, the system response is regarded representing the real error in experiment.

In the current project, the input noise is designed as Gaussian distribution centered at 0 with standard deviation close to 0.7. It is plotted in Figure 5-14 (a). The maximum noise is 3 pixels, but most of them are smaller than or equal to 1 pixel. If we consider in 2D, the noise is usually smaller than 1.44 pixels, which is consistent with the description in section 4.1.1.

The error analysis is processed as following: First, we analyze the overall error of the whole system regardless of whether the result is obtained by three-way or two-way matching. The results are listed in Tables 5-4 and 5-5. Each case has four measurement criteria: overall

error and error in X, Y, Z directions. The overall error is the absolute distance between the reconstructed particles location and the exact particle location. It is always larger than 0. Errors in X, Y, Z directions are defined as the coordinate difference between the reconstructed location and the exact location at each direction. They could be larger or smaller than 0. The overall error estimation provided the evidence to evaluate the performance of the whole algorithm; however, it clarified little about how the hybrid approach has improved the system, and what are the benefits and limitation of this hybrid approach. In response to that, we analyzed the detecting error of various common view regions of the cameras respectively. We started from the three cameras common view region, and then the common view of camera 1 and 2, 1 and 3, and finally 2 and 3. Four measurement criteria including the overall error and the error in X, Y, Z directions are evaluated for each case.

The statistical parameters compared in current research include average error, maximum error, error standard deviation, the detected particle numbers, and three quartiles Q1, Q2 and Q3. According to the simulation result shown in Figures 5-14 to16, the error in X, Y, Z directions has normal distribution. Therefore, their standard deviations could naturally serve as reasonable error indicators. The overall error is defined as the absolute distance between the reconstructed and exact locations. It is not normal distributed as shown in Figure 5-14 (b), 5.15 (a), etc. Therefore, standard deviation is not applicable for them. Instead, their average value and three quartiles are selected as the error indicator.

For the global level experiment case, the analysis result is listed in Table 5-4, Figure 5-14, Figure 5-16 and Figure 5-18. A total of 1162 particles (58.1%) are captured by the algorithm. Among them, 919 particles are matched by all three cameras; 171 are matched by camera 1 and camera 3; and 72 are matched by camera 2 and 3. There are no particles captured by camera 1 and 3. The hybrid approach has increased the detecting rate from 46% to 58.1%. The overall average error of the system is 1.988 mm. The maximum error has achieved 10.321 mm. Assume the velocity was calculated through dividing the frame particle displacement by the time step. The average velocity error is approximately 0.11m/s (the camera rate is 30fps). Considering the

velocity magnitude within indoor environment is within the range of 0.1~1m/s, this error is a significantly large value. Table 5-4 explained the contribution of the overall errors from three dimensions. The standard deviation of the Z-direction has the largest contribution which is 2.087 mm. This number is more than two times of the error in Y-direction and almost three times of those in X-direction Table 5-4 also compared the maximum error of the three directions. The error in Z-direction is 10.255 mm, while those in X and Y directions are only 2.212 mm and 2.746 mm respectively. Obviously, for current system, the resolution in Z-direction is poor compared with the other two directions. This could be explained by the camera configuration shown in Figure 5-8. Limited by the experimental facilities, all cameras have to be installed in front of the cabin. None of them can obtain the high resolution in the depth direction (Z-direction). The whole system is similar to the human vision system. Because two human eyes are close to each other, it is difficult to determine the velocity toward them.

The spatial matching algorithm will also severely affect the accuracy of the system. Table 5-4, Figure 5-14 and 5-16 analyzed the error contributions from both three-way and two-way construction algorithms. As we expected, the three-way reconstruction algorithm has smaller error. Its average error is only 1.575mm compared with the error of two-way reconstruction which is 2.535 mm and 2.358 mm respectively. It was also noticed that there are no particles recovered by camera 1 and 2, because that region is beyond the target volume defined under current camera configuration. Within each case, the error in Z-direction is much larger than those in the X and Y directions. For example, in the three-way construction case, the error in the Z-direction is more than two times than the other two directions. In two-way construction cases, for example the camera 1 and 3 case, the error even achieved four times. Within the two two-way reconstruction cases themselves, their errors are not equivalent either. The error of the camera 1 and 3 case is a little bit larger than those of camera 2 and 3.

Three quartiles in table 5.4 described the error distribution in global experimental case from another aspect. Figure 5-18 summarized these information using boxplots. The center half of the boxplot figure, represented by a rectangle, is a interquartile range defined by Q3-Q1. It

indicated an error range where half of the reconstruction error will fall into. According to Figure 5-18 (a), the interquartile range at the Z-direction is close to two times of those in X and Y directions. Comparing the interquartile range obtained by three-way matching and two-way matching algorithm, shown in Figure 5-18 (b) (c) and (d), it is observed that interquartile range of (c) and (d) is much larger than (b) which is consistent with the previous histogram analysis.

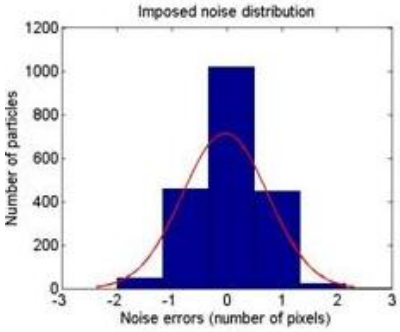
However, for the local level experiment cases, the situation is quite different. As shown in Figure 5-9 (b), the three cameras are installed more perpendicular with each other than the global level experiment. The simulated results are listed in Table 5-5, Figure 5-15, Figure 5-17, and Figure 5-19. A Gaussian distributed noise is imposed onto the particle images as well. The overall averaged error of the whole system is only 0.783 mm. The corresponding velocity error is about 0.047m/s. It reduced nearly three times compared with the global level experiment. Consider the error contribution from two reconstruction algorithms: The error from the three-way construction case is 0.542 mm. The interquartile range is 0.377mm. The errors from the two-way construction cases are 0.745 mm, 0.762 mm and 0.836 mm respectively. Their interquartile ranges are 0368mm, 0.456mm, and 0.743mm. All these errors are comparable. The interquartile range obtained by camera 2-3 is a little bit larger. The three-way construction algorithm has a higher accuracy compared with two-way construction. In regard to the error in different directions, the X-direction has the lowest resolution. Y and Z direction are comparable. But there is no significant difference among them.

Table 5-4. Error ranges table of the global level experiment case.

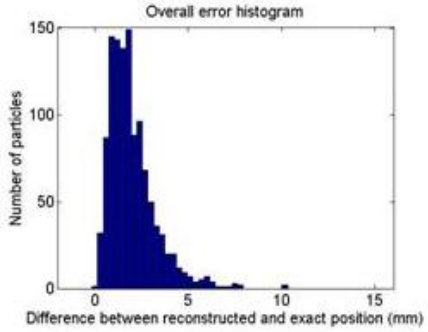
Error classification	Average	Max	Std dev	Count	Q1	Q2	Q3
Imposed noise (pixels)	-0.028	3	0.776	2000	-1	0	1
Overall error (mm)	1.989	10.322	N/A	1162	1.089	1.611	2.442
X (mm)	-0.050	2.212	0.711	1162	-0.484	0.011	0.478
Y (mm)	-0.020	2.746	0.821	1162	-0.562	-0.120	0.539
Z (mm)	0.093	10.255	2.087	1162	-1.269	0.086	1.111
Error of cam 1-2-3(mm)	1.575	7.631	N/A	919	1.038	1.507	2.190
X, Cam 1-2-3(mm)	-0.078	2.212	0.673	919	-0.484	-0.018	0.429
Y, Cam 1-2-3(mm)	0.022	2.569	0.761	919	-0.558	-0.026	0.519
Z, Cam 1-2-3(mm)	0.016	7.349	1.693	919	-1.143	0.010	0.983
Error of cam 1-2(mm)	N/A	N/A	N/A	N/A	N/A	N/A	N/A
X, Cam 1-2(mm)	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Y, Cam 1-2(mm)	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Z, Cam 1-2(mm)	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Error of cam 1-3(mm)	2.535	10.322	N/A	171	1.451	2.496	4.347
X, Cam 1-3(mm)	0.101	2.028	0.806	171	-0.518	0.063	0.622
Y, Cam 1-3(mm)	-0.048	1.778	0.783	171	-0.548	0.020	0.540
Z, Cam 1-3(mm)	0.626	10.255	3.325	171	-2.300	0.175	1.845
Error of cam 2-3(mm)	2.358	10.271	N/A	72	1.334	2.402	3.286
X, Cam 2-3(mm)	0.153	2.124	0.886	72	-0.369	0.137	0.817
Y, Cam 2-3(mm)	-0.270	2.746	1.401	72	-0.775	0.031	0.694
Z, Cam 2-3(mm)	0.594	9.611	2.703	72	-1.516	0.558	1.708

Table 5-5. Error ranges table of the local level experiment cases.

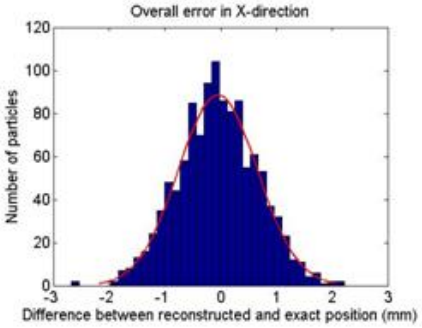
Error classification	Average	Max	Std dev	Count	Q1	Q2	Q3
Imposed noise (pixels)	0.002	2	0.753	2000	-1	0	1
Overall error (mm)	0.783	3.158	N/A	553	0.457	0.677	0.944
X (mm)	0.027	2.435	0.706	553	-0.458	-0.014	0.391
Y (mm)	0.029	1.769	0.380	553	-0.257	-0.026	0.227
Z (mm)	0.007	1.581	0.404	553	-0.254	0.015	0.234
Error of cam 1-2-3(mm)	0.542	1.475	N/A	155	0.337	0.527	0.714
X, Cam 1-2-3(mm)	0.028	1.208	0.469	155	-0.212	0.086	0.313
Y, Cam 1-2-3(mm)	0.003	0.624	0.315	155	-0.216	-0.005	0.212
Z, Cam 1-2-3(mm)	-0.014	0.950	0.337	155	-0.187	-0.004	0.201
Error of cam 1-2(mm)	0.745	2.000	N/A	31	0.507	0.729	0.875
X, Cam 1-2(mm)	0.027	1.542	0.652	31	-0.247	0.102	0.586
Y, Cam 1-2(mm)	0.063	0.811	0.473	31	-0.387	-0.038	0.289
Z, Cam 1-2(mm)	0.003	0.885	0.427	31	-0.227	0.091	0.206
Error of cam 1-3(mm)	0.762	2.252	N/A	260	0.489	0.693	0.945
X, Cam 1-3(mm)	-0.043	2.240	0.758	260	-0.528	-0.065	0.382
Y, Cam 1-3(mm)	0.047	1.002	0.341	260	-0.233	-0.016	0.227
Z, Cam 1-3(mm)	-0.035	1.035	0.363	260	-0.241	0.023	0.232
Error of cam 2-3(mm)	0.836	3.158	N/A	107	0.520	0.823	1.263
X, Cam 2-3(mm)	0.120	2.435	0.838	107	-0.575	-0.158	0.428
Y, Cam 2-3(mm)	0.117	1.769	0.500	107	-0.385	-0.106	0.219
Z, Cam 2-3(mm)	0.082	1.581	0.545	107	-0.395	-0.001	0.403



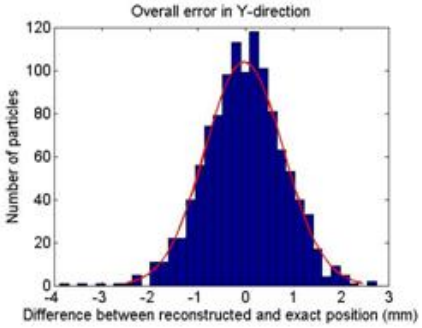
(a) Imposed noise errors



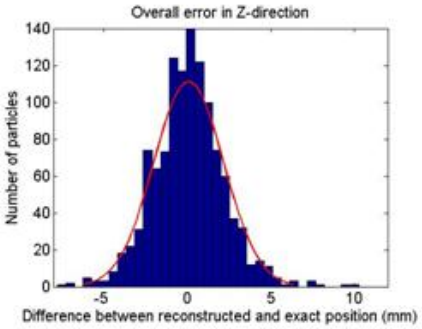
(b) Overall error of the algorithm



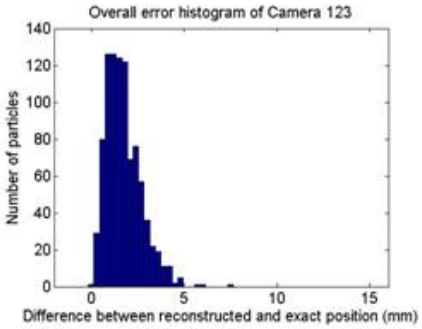
(c) Error in X-direction



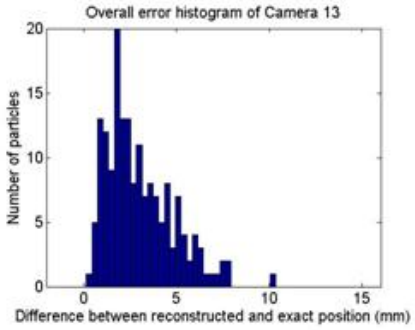
(d) Error in Y-direction



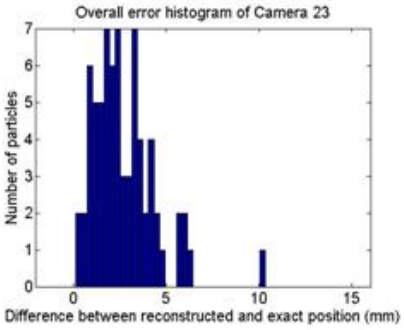
(e) Error in Z-direction



(f) Error in common view of three cameras



(g) Error in common view of camera 1&3



(h) Error in common view of camera 2&3

Figure 5-14. Error analysis of global level experiment case.

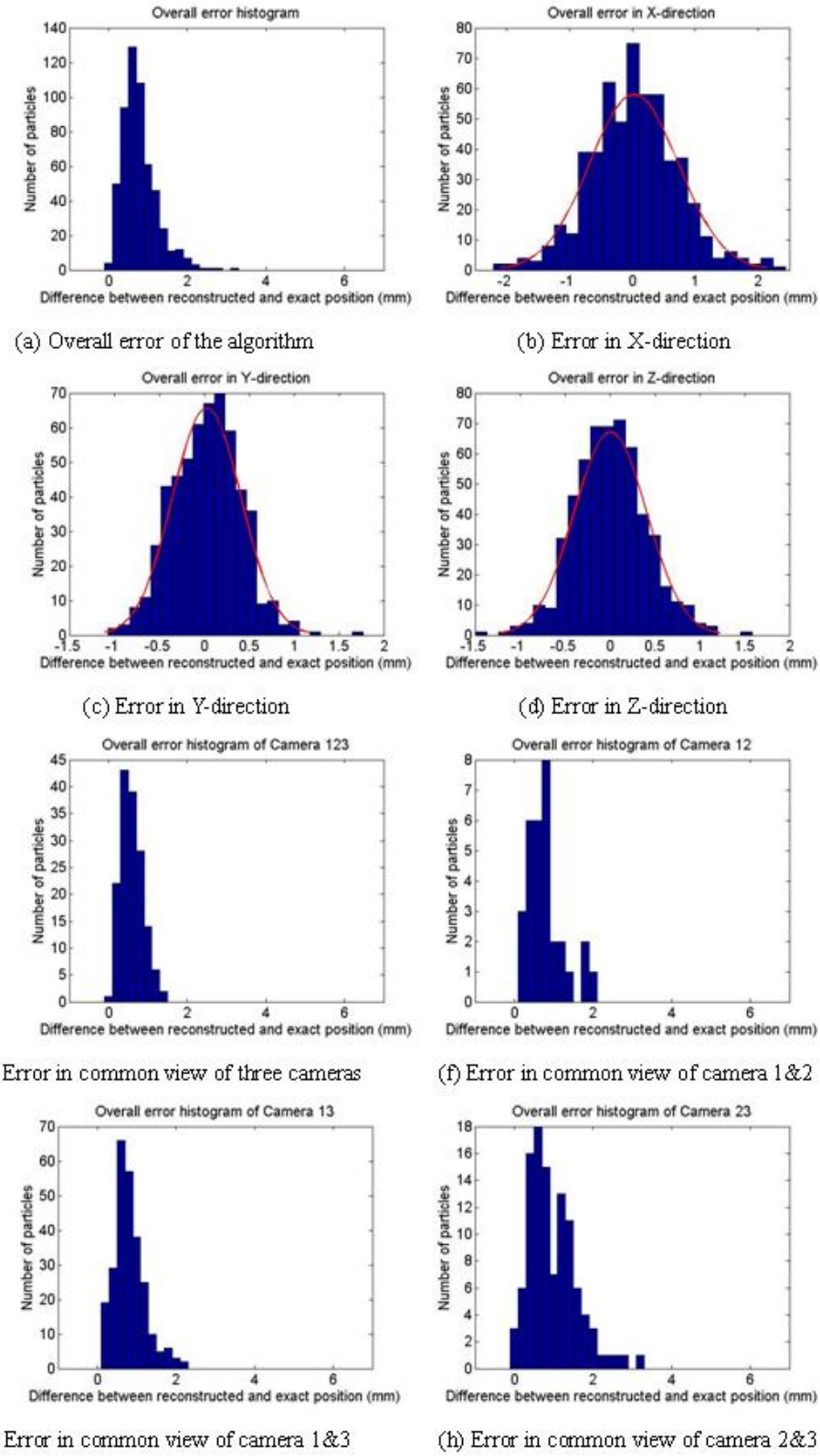
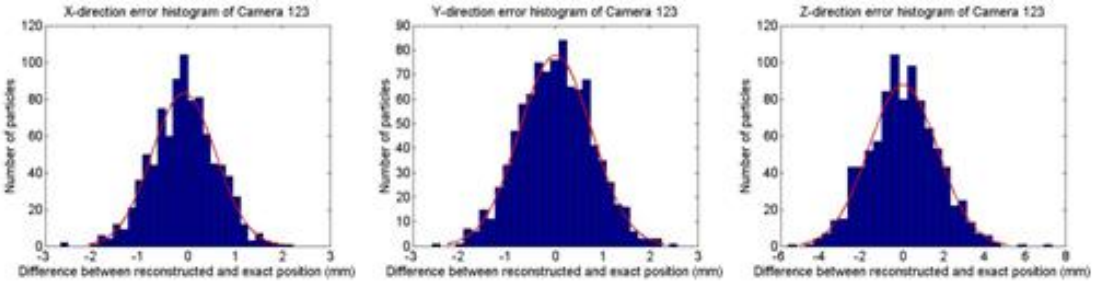
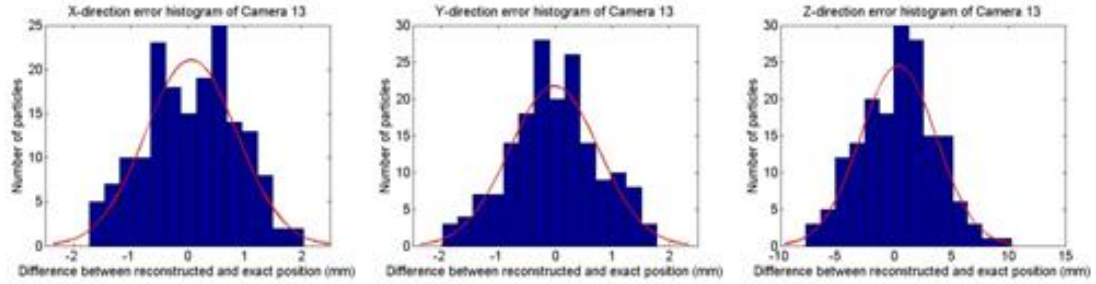


Figure 5-15. Error analysis of local experiment case.



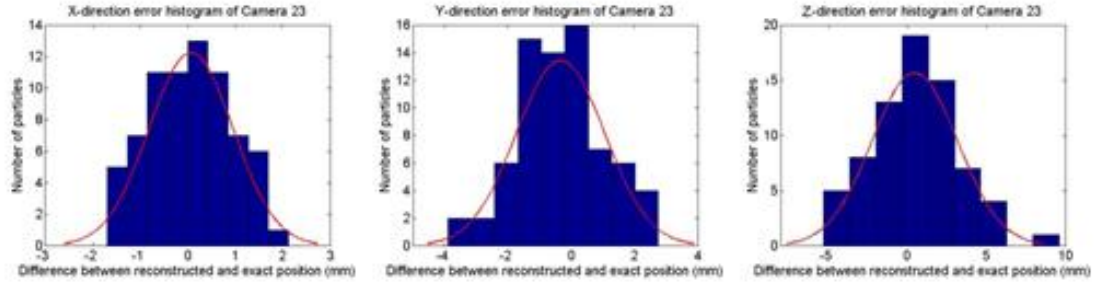
(i) X-direction (ii) Y-direction (iii)Z-direction

(a) Error analysis for common view of Camera 1-2-3



(i) X-direction (ii) Y-direction (iii)Z-direction

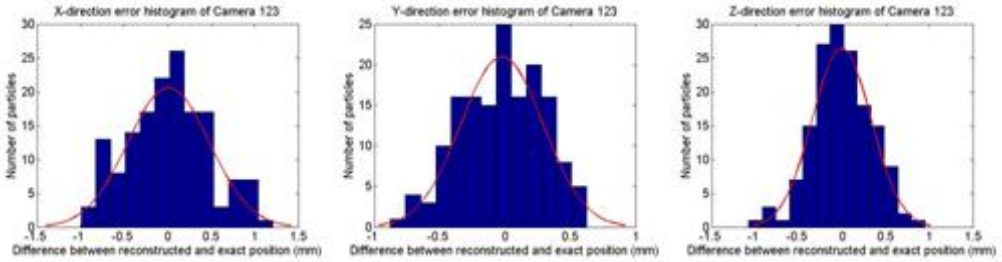
(b) Error analysis for common view of Camera 1-3



(i) X-direction (ii) Y-direction (iii)Z-direction

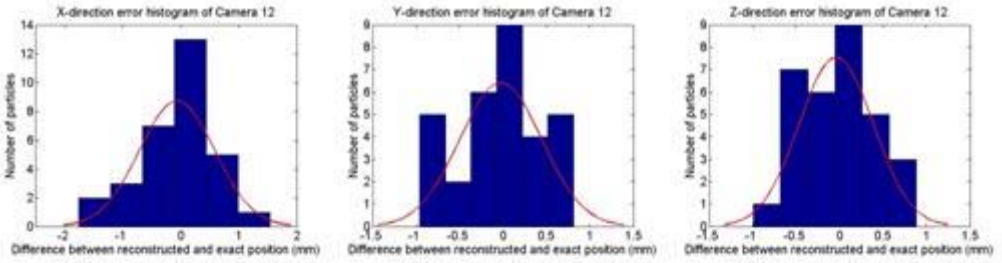
(c) Error analysis for common view of Camera 2-3

Figure 5-16. Error analysis for camera common view of global level experiment case.



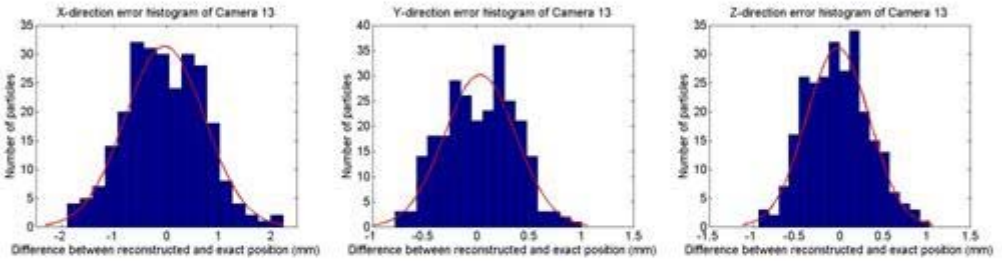
(i) X-direction (ii) Y-direction (iii)Z-direction

(a) Error analysis for common view of Camera 1-2-3



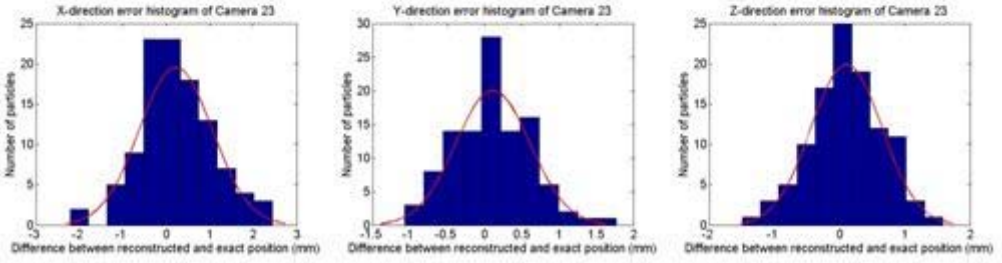
(i) X-direction (ii) Y-direction (iii)Z-direction

(b) Error analysis for common view of Camera 1-2



(i) X-direction (ii) Y-direction (iii)Z-direction

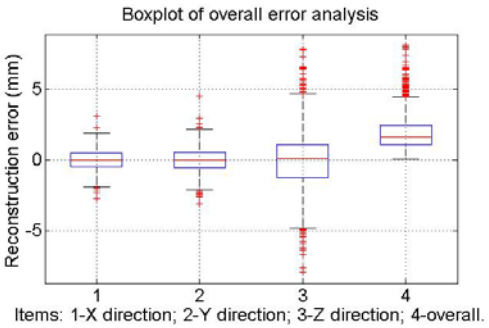
(c) Error analysis for common view of Camera 1-3



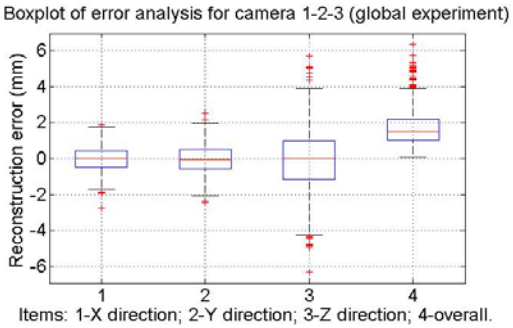
(i) X-direction (ii) Y-direction (iii)Z-direction

(d) Error analysis for common view of Camera 2-3

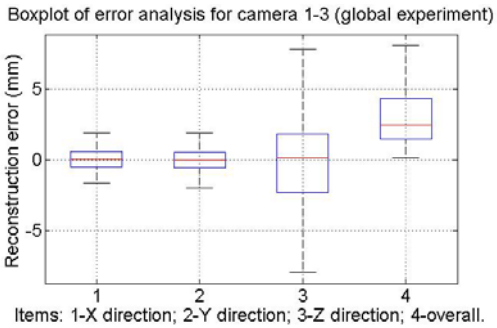
Figure 5-17. Error analysis for camera common view of local level experiment case.



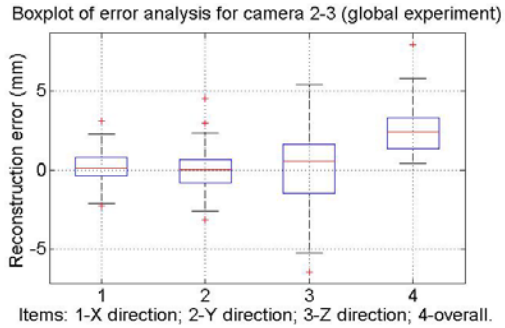
(a) Boxplot of overall error analysis



(b) Boxplot of error analysis for camera 1-2-3

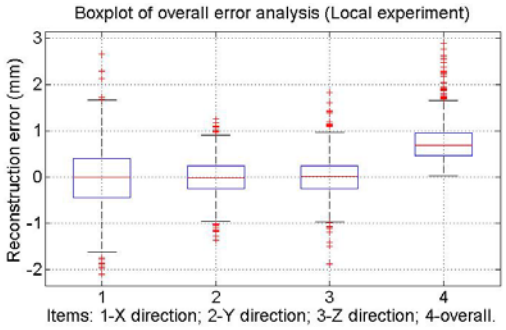


(c) Boxplot of error analysis for camera 1-3

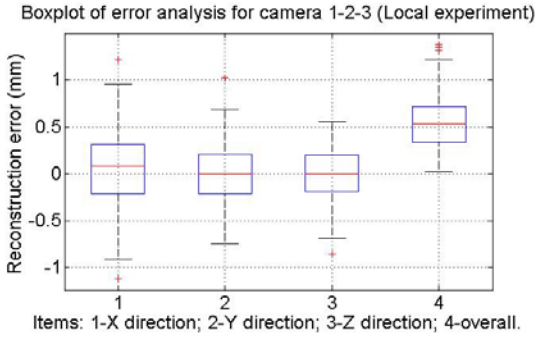


(d) Boxplot of error analysis for camera 2-3

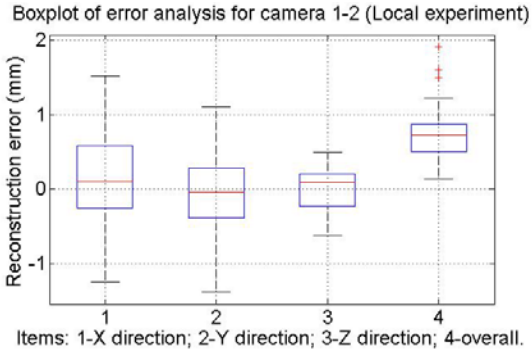
Figure 5-18. Boxplot error analysis of global experimental case.



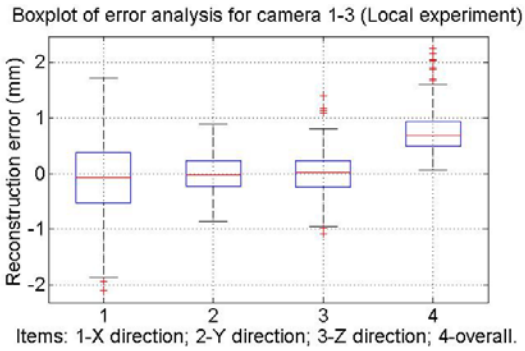
(a) Boxplot of overall error analysis



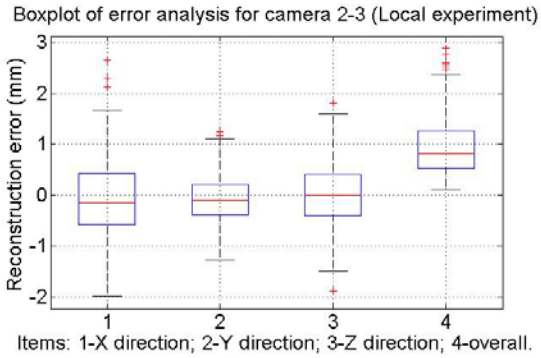
(b) Boxplot of error analysis for camera 1-2-3



(c) Boxplot of error analysis for camera 1-2



(d) Boxplot of error analysis for camera 1-3



(e) Boxplot of error analysis for camera 2-3

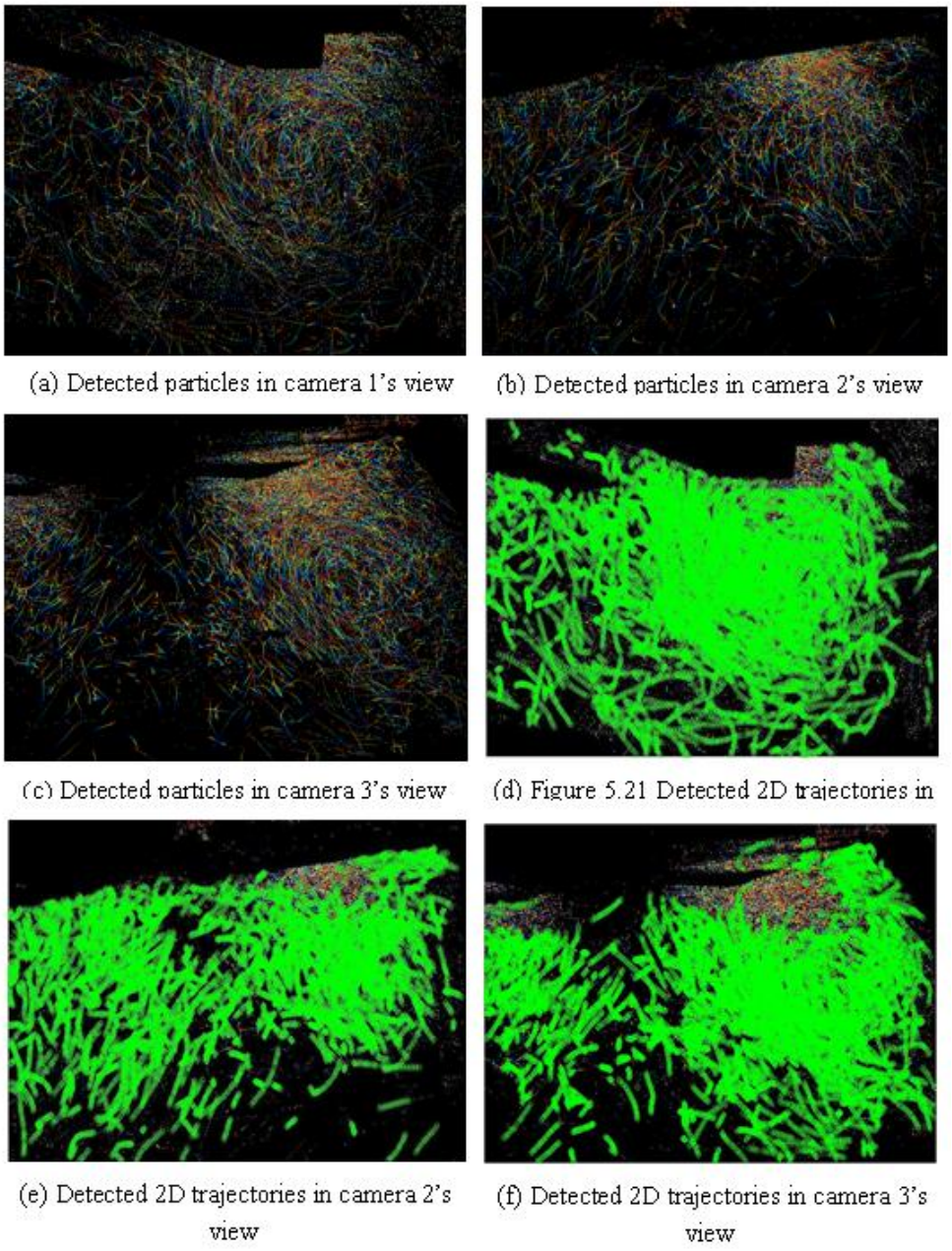
Figure 5-19. Boxplot error analysis of local experimental case.

5.3.2.4 2D particle tracking

In current research, the image recording duration is 1 second. The CCD camera rate is 30fps. A total of 90 images were taken for each global and local level case.

The recorded images are processed by particle detecting and 2D tracking algorithms sequentially as described in chapter 3. To visualize the result, the detected particle is rendered with various colors according to the frame index they belonged to. Particles of all thirty frames

taken by one camera are combined into one image and the identical particles are linked together to visualize their moving trajectories and directions. After 2D tracking, the linked 2D trajectories are plotted onto the recorded images to verify the accuracy of the algorithm. The resultant images for the global and local level experiment are shown in Figure 5-18 and Figure 5-19 respectively.



(a) Detected particles in camera 1's view (b) Detected particles in camera 2's view

(c) Detected particles in camera 3's view (d) Figure 5.21 Detected 2D trajectories in

(e) Detected 2D trajectories in camera 2's view (f) Detected 2D trajectories in camera 3's view

Figure 5-20. Detected particles and 2D trajectories for global airflow case.

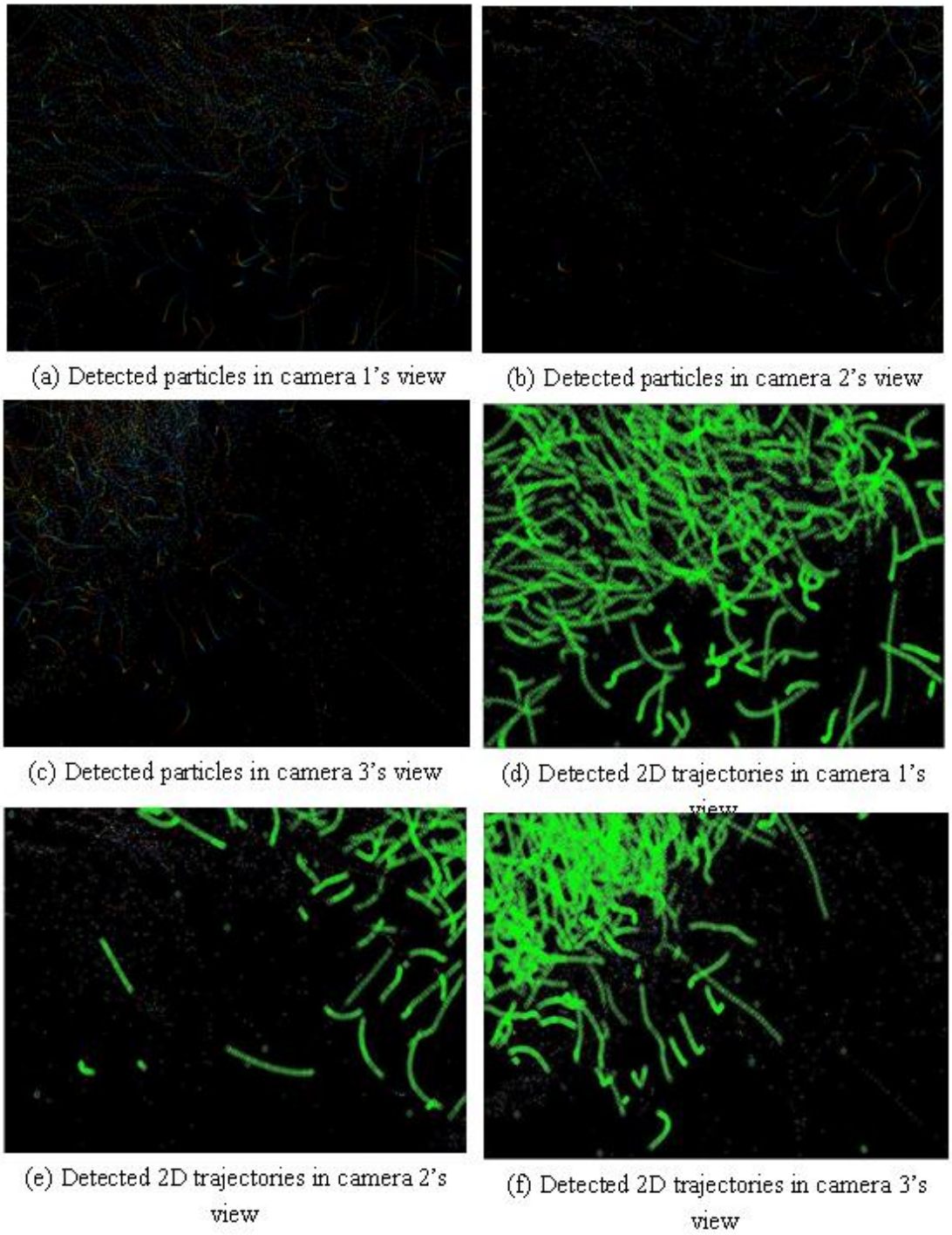


Figure 5-21. Detected particles and 2D trajectories for local airflow case.

5.3.2.5 3D particle reconstruction

Spatial matching and 3D reconstruction algorithms have been proposed in section 3.2.2 and section 3.3. Both 2D trajectories and streaks will be processed following that procedure. Figure 5-20 shows the typical detected 3D trajectories. It is the tracking result of the global level experiment. Obviously, without proper post-processing, this result is neither informative nor straight forward.

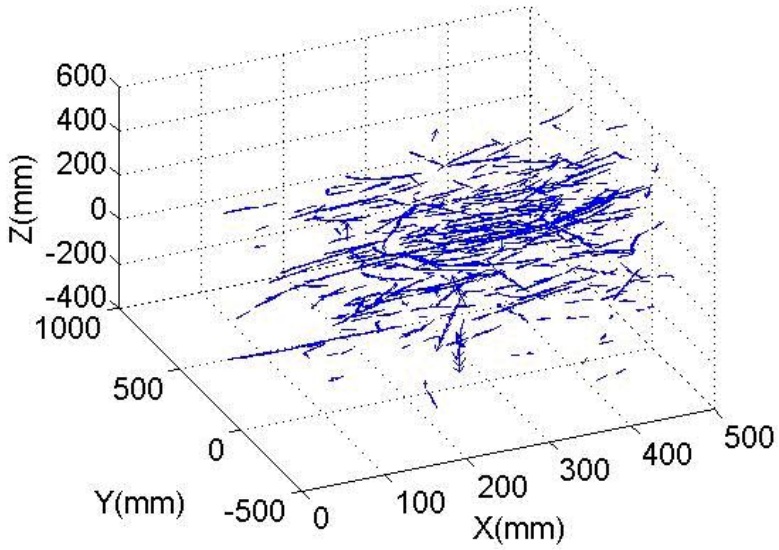


Figure 5-22. Detected 3D trajectories visualized with Matlab.

5.3.2.6 Data validation and post-processing.

The resultant data set of the HPTV system consists of lots of randomly distributed discrete particles within the testing volume. To conduct data-validation and to compare with numerical simulation results in the future, this data set has to be manipulated into a straight forward framework. Three approaches are proposed to achieve this purpose: (1) Image-based particle back projection; (2) 3D discrete trajectory visualization; and (3) Uniform grid data interpolation.

Image-based back projection is exactly a reverse procedure of the HPTV system. Given a 3D location of a particle and the corresponding camera parameters, its particle images positions in each camera are uniquely determined. The overall accuracy of the HPTV system can be evaluated in situ by projecting all the detected particles back to the original raw images frame by frame. Through analyzing the difference between the projected particles images and the recorded particle images, the particle detecting rate and accuracy could be directly evaluated. The alternative approach is to project the detected particles in all frames onto one image and compare it with the image combined all frames as discussed in section 5.4.2.4.

The back-project approach provides an effective way to validate the resultant data. But it is not a good way for data visualization because it completely drops the information in the third dimension. This limitation is severe if the HPTV system is applied to track the velocity field or pollutant dispersion within large scale testing volume. The purpose of the 3D discrete trajectory visualization approach is to overcome this limitation. A visualization toolbox based on the Visualization ToolKit (VTK) has been developed. It has the capability to visualize the discrete particles, interpolate the 3D trajectories, and process animation related tasks. It has proven to be a powerful visualization tool for HPTV system.

One motivation for developing the HPTV system is to generate data for numerical model validation. However, the obtained velocity data through HPTV are randomly located within the whole volume. A tool has been developed to manipulate the raw data into uniform grid format for comparison purpose. This is accomplished by 3D grid data interpolation. The resultant interpolated data could be further visualized by some conventional data processing software, for example TecPlot, to produce vector, contour or streamline maps and compare directly with the simulated result.

5.4 Summary

In this chapter, we first introduced the experimental facilities where the HPTV system is applied to. The detailed experimental parameters and requirements are described and discussed. Two experimental configurations are proposed: global flow measurement and local flow measurement. In each configuration, three different ventilation rates, 80%, 100% and 120% of the full ventilation load, are tested.

A complete experimental procedure is developed. Through application examples, the detail approach to optimize the camera setup, evaluate the system accuracy, and perform the post-processing has been explained and documented.

The errors of the HPTV system under the experimental camera configurations are analyzed based on particle simulation and statistical methods. It is approved that the error in the local experimental case is almost three times smaller than those in global experimental case because of the perpendicular camera configuration in the latter case. It is also observed that three-way reconstruction algorithm has a higher accuracy than two-way construction algorithm; but its detectable volume is smaller. The errors contributed from X, Y, and Z directions are not equivalent either. The camera configuration, especially the camera angle, is the key factor affecting the accuracy.

CHAPTER 6 STUDY AIRFLOW PATTERN WITHIN CABIN WITH COMPUTATIONAL FLUID DYNAMICS

Experimental and computational fluid dynamics are two of the most important approaches to investigate indoor air quality problems. Compared with experimental methods, computational fluid dynamics (CFD) could provide more detailed information with less expense. The earliest numerical study for aircraft cabins was conducted by Aboosaidi et al. (1991). With the development of IT technology, the capability of computers has increased rapidly. More realistic cabin geometry and manikins have been adopted in the CFD study. Singh et al.(2002) used a heated cylinder to represent a passenger in an aircraft cabin. Lin et al. (2005; 2006;) used more detailed geometry to study the airflow and pollutant transport in cabins. Zhang and Chen (2007) used a realistic cabin to study novel air distribution for a commercial aircraft cabin.

Sorensen and Nielsen (2003) have given a good recommendation on how to control the quality of CFD applying to IAQ simulation. A proper turbulence model should be selected according to the airflow characters in the room and the requirement of the accuracy. The most frequently used models for indoor environment research include the zero-equation model, standard k-ε model, LRN k-ε model, RNG k-ε model, k-ω model, Reynold stress model, large eddy simulation, etc. For an indoor environment, the velocity is usually small. Many researchers have used the LRN k- ε model to do the simulation, (Hayashi et al., 2002; Murakami, 2004; Topp et al., 2002; Zhu et al., 2005) because it is good at treating the near-wall layer. It uses modeling instead of wall function to treat the near-wall domain. However, very fine grids are needed to do this simulation, which will increase the cost of the computing. The standard k-ε model and RNG k-ε model are also widely used in IAQ studies (Gao and Niu, 2004; He et al., 2005). They use wall function to treat near wall air flow, therefore no fine grids are needed at the wall boundary. Chen (1995) compared different k-ε models for indoor airflow simulation and recommended the RNG k-ε model due to its better performance than other models. Based on these researches, we chose the RNG k-ε model to close our N-S equations.

6.1 Mathematical model

Since the purpose here is not developing the turbulent models, this section will only briefly review the existing RNG k-ε model developed by Yakhot and Orszag (1986). Detail derivation could be found on his original paper. The isothermal incompressible turbulent flow is controlled by the following governing equations (Gebremedhin and Wu, 2003):

$$\begin{cases} \dfrac{\partial u_j}{\partial x_j} = 0 \\ \dfrac{\partial u_i}{\partial t} + \dfrac{\partial}{\partial x_j}\left(\rho u_j u_i\right) = -\dfrac{\partial p}{\partial x_i} + \dfrac{\partial}{\partial}\left(\mu_f\left(\dfrac{\partial u_i}{\partial x_j} + \dfrac{\partial u_j}{\partial x_x}\right) - \rho\overline{u_i' u_j'}\right) \end{cases} \tag{6.1}$$

The most difficult item in equation 6.1 is the Reynolds stress term $\rho\overline{u_i' u_j'}$. The Reynolds stress term cannot be expressed as a function of mean flow variables without a meaningful model. The purpose of a turbulence model is to build a relationship between the velocity fluctuation and the mean velocity, which is also called equation/turbulence closure.

One of the most successful approaches is Boussinesq's eddy-viscosity equation. It assumed that Reynolds stress is proportional to mean velocity gradient:

$$-\rho\overline{u_i' u_j'} = -\mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) + \frac{2}{3}\rho\delta_{ij}k \tag{6.2}$$

And the eddy viscosity μ_t is further related to the turbulent kinetic energy k and its dissipation rate ε.

$$\mu_t = \rho c_p \frac{k^2}{\varepsilon} \tag{6.3}$$

The kinetic energy and dissipation rate is controlled by their transport equations as follows (ANSYS, 2007):

$$\begin{cases} \dfrac{\partial}{\partial t}(\rho k)+\dfrac{\partial}{\partial x_i}(\rho k u_i)=\dfrac{\partial}{\partial x_j}\left(\alpha_k \mu_t \dfrac{\partial k}{\partial x_j}\right)+G_k-\rho\varepsilon \\ \dfrac{\partial}{\partial t}(\rho \varepsilon)+\dfrac{\partial}{\partial x_i}(\rho k \varepsilon_i)=\dfrac{\partial}{\partial x_j}\left(\alpha_c \mu_t \dfrac{\partial \varepsilon}{\partial x_j}\right)+C_{1c}\dfrac{\varepsilon}{k}G_k-C_{2c}\rho\dfrac{\varepsilon^2}{k}-R_\varepsilon \end{cases} \tag{6.4}$$

In these equations, G_k is the production rate of turbulent kinetic energy given by:

$$G_k=\mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)\frac{\partial u_i}{\partial x_j} \tag{6.5}$$

α_k and α_c are the inverse effective Prandtl number for k and ε. In the high-Reynolds-number limit, they are constants. R_ε is a unique term introduced by RNG k-ε model. Its equation reads as:

$$R_\varepsilon=\frac{C_\mu\rho\eta^3(1-\eta/\eta_0)}{1+\beta\eta^3}\frac{\varepsilon^2}{k} \tag{6.6}$$

R_ε could combine with $C_{2c}\rho\dfrac{\varepsilon^2}{k}$ and serve as an correction for C_{2c}. This correction term makes the RNG k-ε model perform better than the standard k-ε model in the regions of large strain rate ($\eta>\eta_0$), because the R_ε term will make a negative contribution. As a result, the RNG model yields a lower turbulent viscosity than the standard k-ε model. For weakly to moderately strained flows, the RNG model yields a comparable result to the standard k-ε model.

6.2 Geometry of the ventilation space and manikins

The aircraft cabin geometry for CFD simulation is developed according to the size of the full-scale experimental facilities. Detail dimension and setup is shown in Figure 6-1.

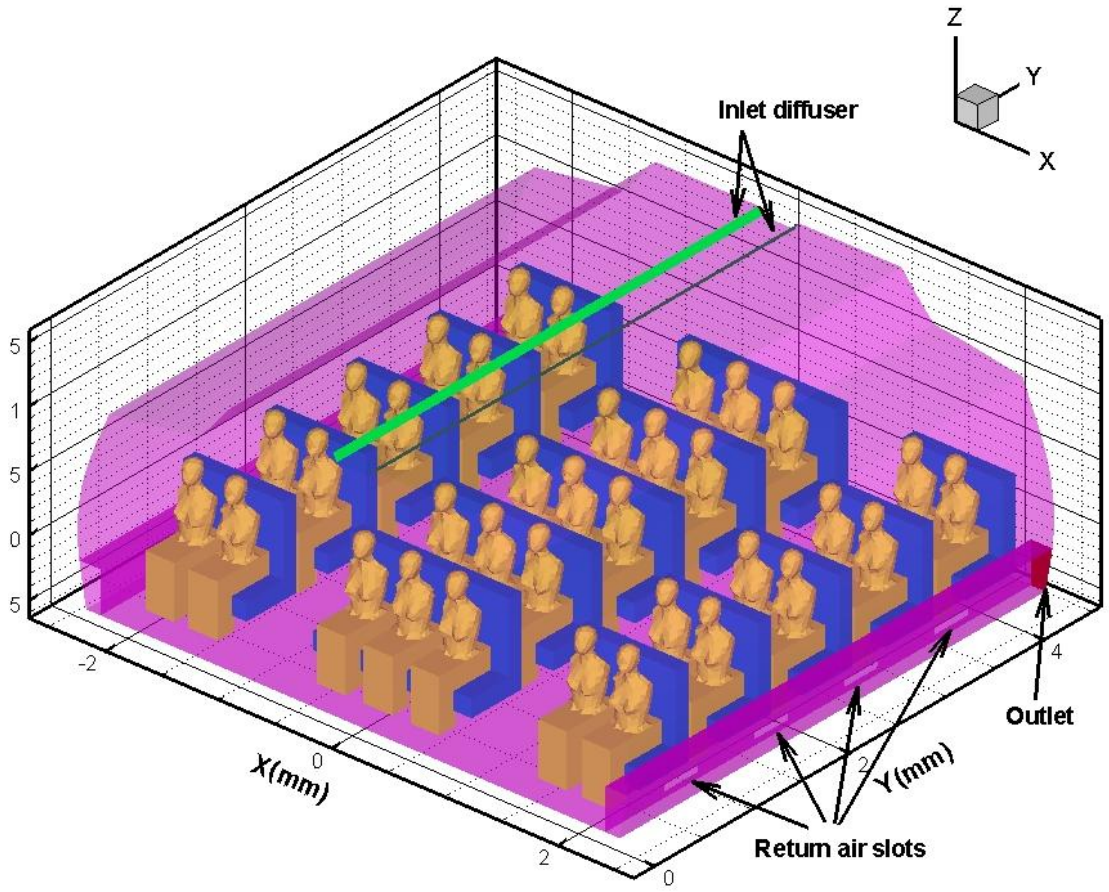
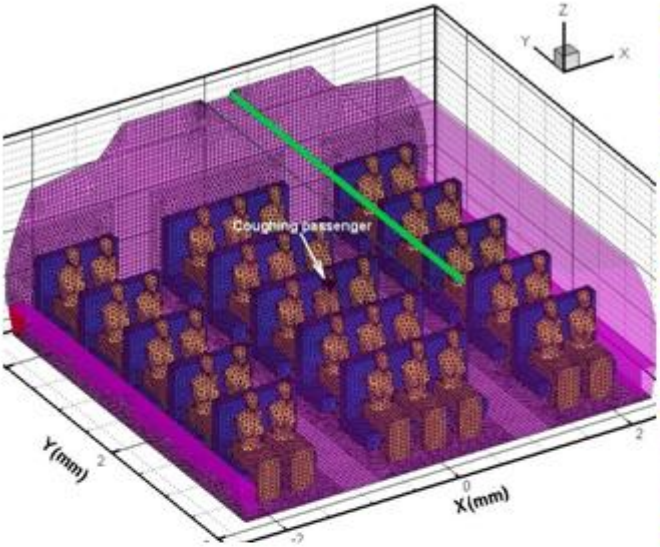


Figure 6-1. Aircraft cabin geometry setup.

According to the report of Topp et al. (2002), box-shape manikins are usually sufficient for global airflow simulations. However, it is not sufficient to analyze the airflow pattern produced by the coughing of a passenger at the local environment, which is also one of the research purposes for the current work; therefore, a complicated manikin with more detail features is used in this simulation. The geometry is developed based on a manikin of the author's previous research (Wei et al., 2009). The original manikin is a standing person obtained by laser scanning technique. To fix it in the narrow seats space within the cabin, only the upper body was used. The arms are removed too. The lower body is substituted by two simplified box as shown in Figure 6-1.

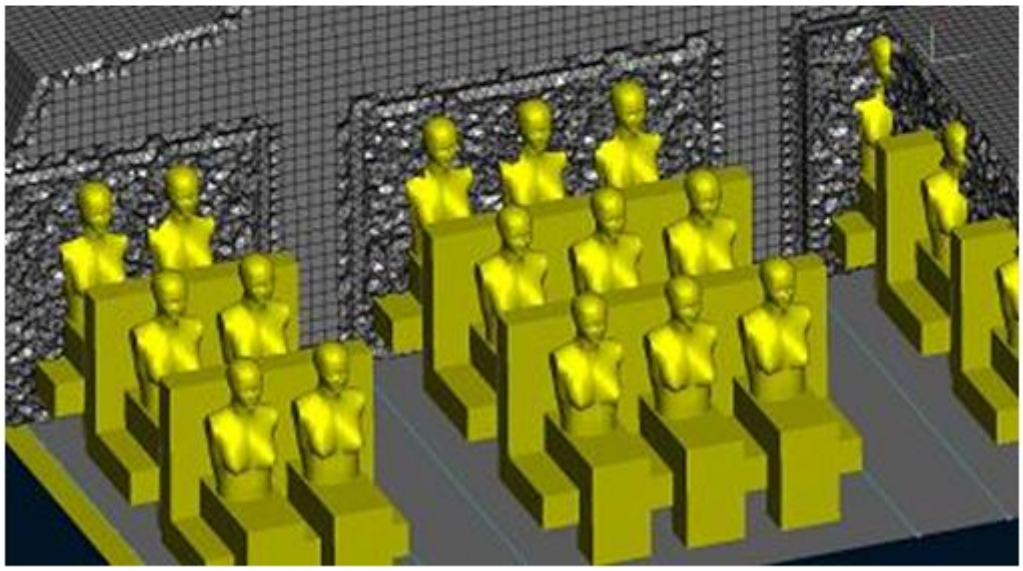
6.3 Mesh generation

The computational domain was discretized using hybrid structured hexahedron and unstructured tetrahedral finite volume mesh grids. Generating a computational mesh with sufficient resolution is critical to achieve the desired accuracy in CFD simulation. If the mesh is too coarse, small scale velocity gradients may be poorly represented, resulting in an inaccurate simulation result. At the other extreme, an overly fine mesh will waste a lot of computational resources, and therefore, should also be avoided. To determine the appropriate mesh size for an accurate simulation, several tests must be conducted using successively finer meshes to analyze the sensitivity of the solution to the mesh size. The final mesh size selected should be the coarsest mesh that fully resolves the desired flow phenomenon. The alternative meshing approach is to divide the volume into separate computational zones whose mesh size is assigned according to the complexity of the features within the zone itself. After several tests, the final mesh setup for current geometry design is shown in Figure 6-2. The mesh is not uniformly generated. The domain close to the manikins is meshed with unstructured grids, while the outer domain is meshed with structured grids. The passenger releasing the coughing jet is meshed with the finest grids with the minimum size equal to 0.02m. The size of the other unstructured and structured grid is 0.05m. An interior condition is defined at the interface. The reason for this arrangement is that the geometry of manikins is very detailed; therefore, we expected complicated airflow structure at this region compared with the outer domain. Finer mesh is necessary to achieve good accuracy. The total number of resultant cells is 2,001,594.



(a) Overall mesh generation setup

(b) Mesh generation for the coughing passenger



(c) The slice view of the cabin grids

Figure 6-2. Mesh generation of the aircraft cabin and passengers.

6.4 Computational facilities

The simulation work was performed on a workstation in the BioEnvironmental Engineering Laboratory (BEE) at the University of Illinois at Urbana-Champaign. It has 8 Intel Xeon CPUs working at 2.40 GHz frequency. Each CPU has 4GB DDR memory; therefore, a

total 32GB memory could be used. Four graphic cards are installed in this work station. One NVIDIA Quadro FX 5600 with 1.5GB GDDR3 memory served as the display card. It also has three NVIDIA Tesla GPU card designed for high performance computing purposes, which are not used in the current simulation.

There are many commercialized CFD codes that equipped the RNG k-ε model, such as CFX, PHOENICS and FLUENT. In the present work, the parallel version FLUENT 6.3 was used as the numerical solver. The developing geometry and mesh generation are processed by the commercial package ANSA 12.0.

6.5 Boundary conditions and simulation procedure

Several assumptions are made in this study: (1) we only consider the steady-state ventilation; (2) uniform velocity boundary condition is assumed at the inlet diffuser; (3) neglect the heat transfer of the passengers; (4) gravity and buoyancy effect are neglected; and (5) assume the air is incompressible.

For global level simulation, the inlet velocity magnitude is set to be 0.37, 0.47 and 0.56m/s, which corresponds to the 80%, 100% and 120% ventilation rate recommended by Boeing. The turbulence intensity is set to be 10%. The injection angle of the inlet is 30°, which is defined according to the observation of experiment. The two outlets of the cabin is simply set as “outlet” with equal weight in FLUENT considering the cabin is symmetric. The interfaces between meshing zones are defined as internal.

For the local environment simulation, the mouth jet is also assumed as a uniform velocity inlet. The velocity magnitude is set to be 7.72m/s with 30% turbulent intensity. This value is chosen according to the in situ measurement. The other boundary conditions for ventilation are identical to global level simulation.

To obtain predictive accuracy and a stable solution, it is important to utilize a reliable simulation strategy. First, we process the global airflow simulation. At the initial phase, a laminar flow model without considering the turbulent characteristics is applied to all cells. The very stable first order upwind discretization scheme was used. The resulting data serve as the initial

values for the turbulent model simulation. Then, the turbulent kinetic energy and dissipation equations are coupled, and the second order upwind discretization scheme is used for these equations. Pressure and velocity coupling is handled through the use of the semi implicit pressure linked equations (SIMPLE) method. After we obtained the airflow field for global level simulation, the resultant field is used as the initial value for the local level simulation. Then we couple the boundary conditions for the coughing jet flow and study the local environment affected by the coughing jet of the passenger.

In all these simulations, the result is considered to be converged if the velocity residues in the equations are smaller than 10^{-5}m/s.

6.6 Summary

In this chapter, we first briefly reviewed the previous simulation work on this topic, and the existing turbulent models applied for an indoor air quality study. Then we explained why the RNG k-ε model was chosen for current research. Some detail information of the RNG k-ε model was also introduced. In the next few sections, we described the detail geometry features, mesh generation strategies and how to ensure the high quality of the simulation results. The computational facilities, including hardware and software available for the simulation are also introduced. Finally, we proposed the boundary conditions and the detail simulation procedure.

CHAPTER 7 RESULT AND DISCUSSION

Three objectives are expected to be achieved by applying the HPTV system: visualize the airflow field, re-create the particle movement within the testing volume and produce reliable experimental data for numerical modeling result validation. For validation purposes, two approaches have been established in previous chapters. Project all detected 3D trajectories back onto each camera field of view and compare with the 2D trajectories obtained through image processing based on raw image. This method served as an experimental result self-validation. To validate the numerical modeling result, a uniform grid interpolation technique is used to map the randomly located velocity data onto uniform grids. The resultant data is visualized as contour map, vector map and streamline map, etc., and compared directly with the simulated result. The data is also visualized in a 3D particle movement video, trying to make an exact reproduction of the experimental scenario. This process is accomplished by a toolbox developed based on the VTK library.

As mentioned in the previous two chapters, the experiment and simulation study are conducted at both global and local levels. In each level, three ventilation rates are applied, 80%, 100% and 120% of the full load. Accordingly, the results will be discussed in this order.

7.1 Experimental and numerical result at global level

We assumed that the velocity field within the aircraft cabin at given ventilation rates has achieved steady-state when we conducted the test. For each ventilation rate, two replicated image sets were sequentially recorded by the system. It took one second to generate one set. In each image set, 30 frames were captured by each CCD camera. Therefore, there were 90 images in each image set and 180 images at each ventilation rate. The HVPT algorithm is applied to each

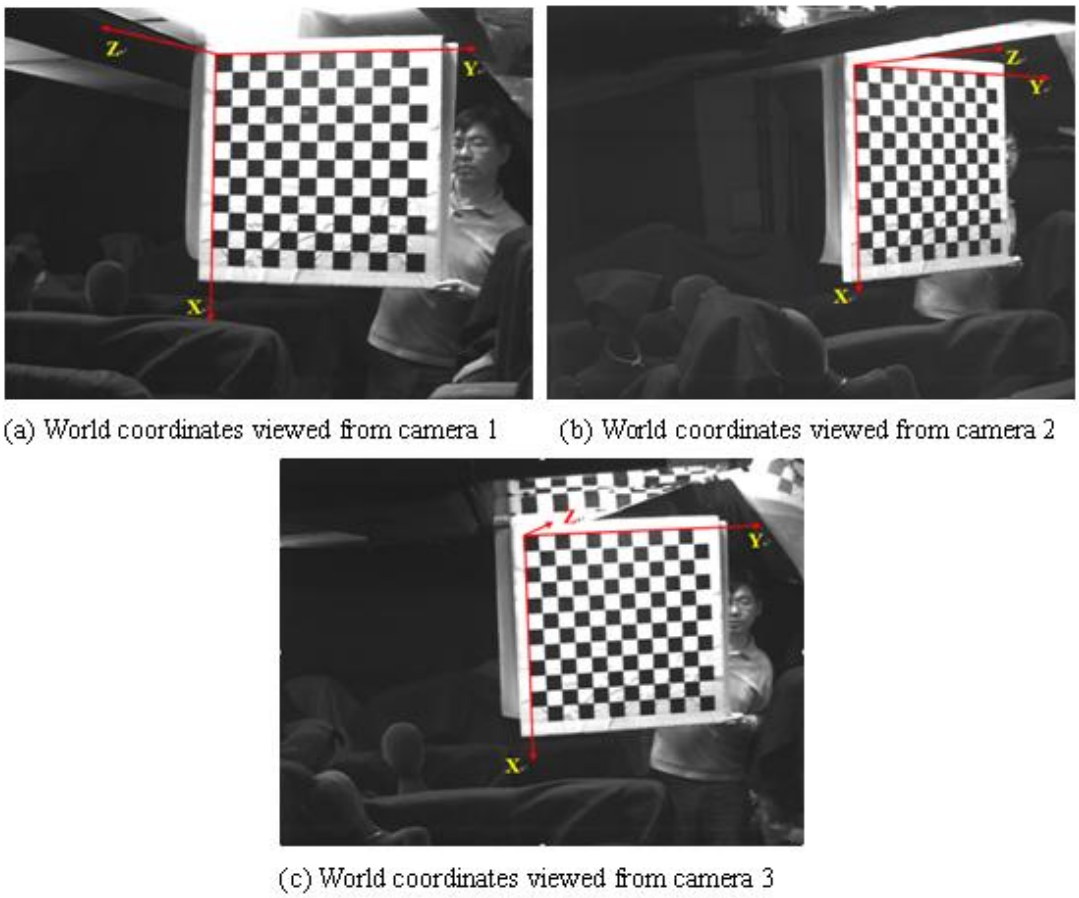
image set independently. The 3D trajectories obtained from these two sets are combined together to increase the spatial resolution of the data.

7.1.1 Coordinate system of the experiment

All detected particles or trajectories should have a common reference coordinate system. This system is determined during the camera calibration process. A good choice of the coordinate system should best represent the particles moving within the target volume. It is defined through a calibration board, which determined the X-Y plane. The Z-direction points outward to the board according to the right-hand rule. The extrinsic camera parameters associated with this position will be used in the coordinate reconstruction of the detected particles and the post-processing as well. All global level experiments had an identical coordinate system shown in Figure 7-1.

7.1.2 Case 1: Global airflow study at 80% ventilation rate

The first study is the global airflow within an aircraft cabin forced by a ventilation system. Figure 7-2 shows the detected 3D trajectories by the HPTV system. Each trajectory indicates the movement path-line of a particle in one second. These trajectories are randomly located in the target volume.



(a) World coordinates viewed from camera 1 (b) World coordinates viewed from camera 2

(c) World coordinates viewed from camera 3

Figure 7-1. Define the world coordinate system for global level experiments.

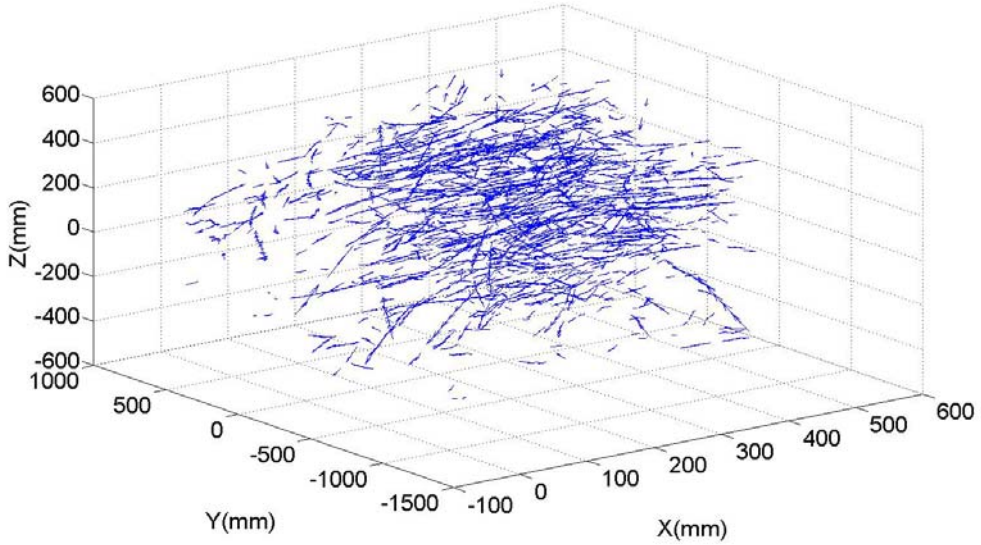
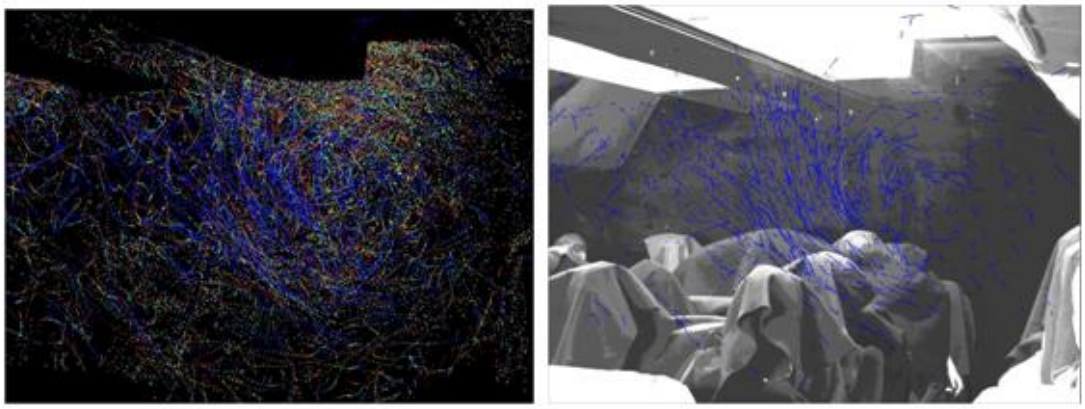


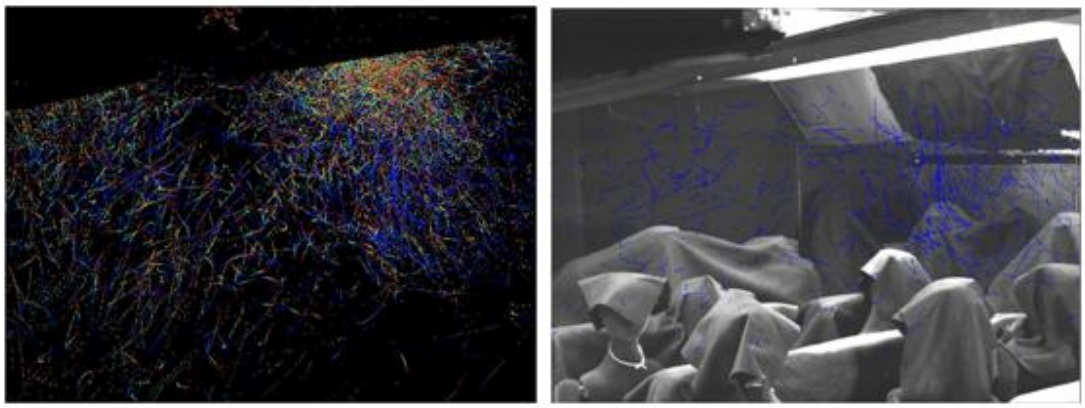
Figure 7-2. Detected 3D trajectories for Case 1 (global airflow study at 80% ventilation rate).

To validate the experiment result, these trajectories are projected into three cameras field of view as shown in Figures 7-3 to 7-5. In the first image of each figure the 3D particles are projected on the 2D trajectories images obtained by image processing. In the second image of each figure, they are projected on the original raw images taken by each camera. It is observed that the projected particles successfully captured the movement trend of the airflow pattern. The large vortex generated by the air diffuser at the upper right region of the cabin is clearly observed. Most of the projected 3D particles could find their corresponding images which they were reconstructed from. Since a hybrid matching strategy has been used in the algorithm, it is possible that some projected trajectories cannot be matched to any 2D ones in one camera field of view. In such a case, it means that they are recovered from the information provided by the other two cameras.



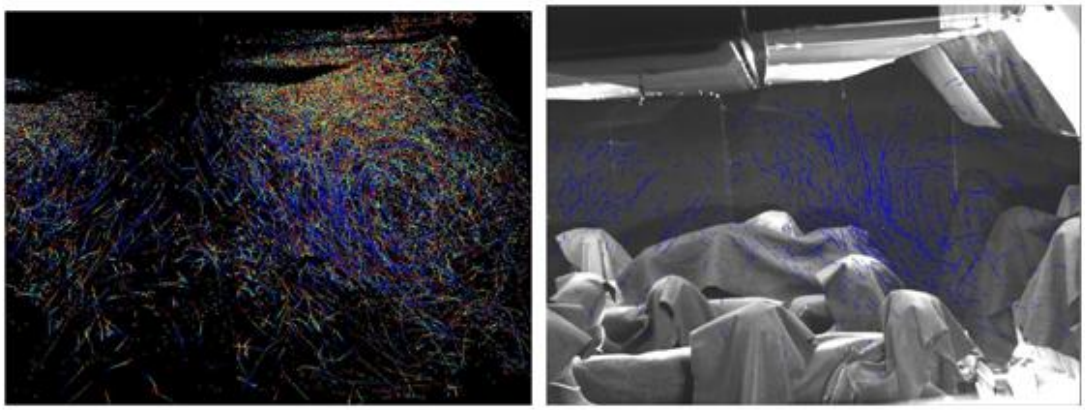
(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-3. Back projecting the detected 3D trajectories onto camera 1 for Case 1.



(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-4. Back projecting the detected 3D trajectories onto camera 2 for Case 1.



(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-5. Back projecting the detected 3D trajectories onto camera 3 for Case 1.

We have discussed the reconstruction errors in chapter 5. We know that for global airflow measurement, the velocity error in Z direction is quite large compared with the velocity range we are measuring. Therefore, only the velocity at X and Y directions are analyzed here.

CFD provided detailed information with high spatial resolution. To compare it with the experimental result, a slice plane at the central location of the cabin (Y=2.2m) is extracted as indicated in Figure 7-6. Notice that the coordinate system of the experiment and simulation is slightly different. The velocity components in the X and Y directions are shown with vector map

colored by velocity magnitude. Figure 7-7 compared the numerical result and the experimental result.

In Figure 7-7 (a), a rectangular box approximately corresponding to the detected region in Figure 7-7 (b) is indicated. Because the spatial resolution of the detected bubbles is not high enough, the experimental result could only be represented in Lagrangian frame. Within the detected box, two obvious airflow pattern characters are observed: the left upward airflow (from X = 1000mm to -400mm) and the large right vortex (from X= -200mm to 800mm). Compare Figure 7-7 (a) and (b), obviously the left upward flow is a small part of a large left vortex generated by the air conditioning system. The detected velocity magnitude of the upward bubbles ranged from 0.1m/s to 0.2m/s. This is consistent with the prediction of the CFD simulation. Limited by the field of view of the camera, the left vortex is not completely captured in the recorded images.

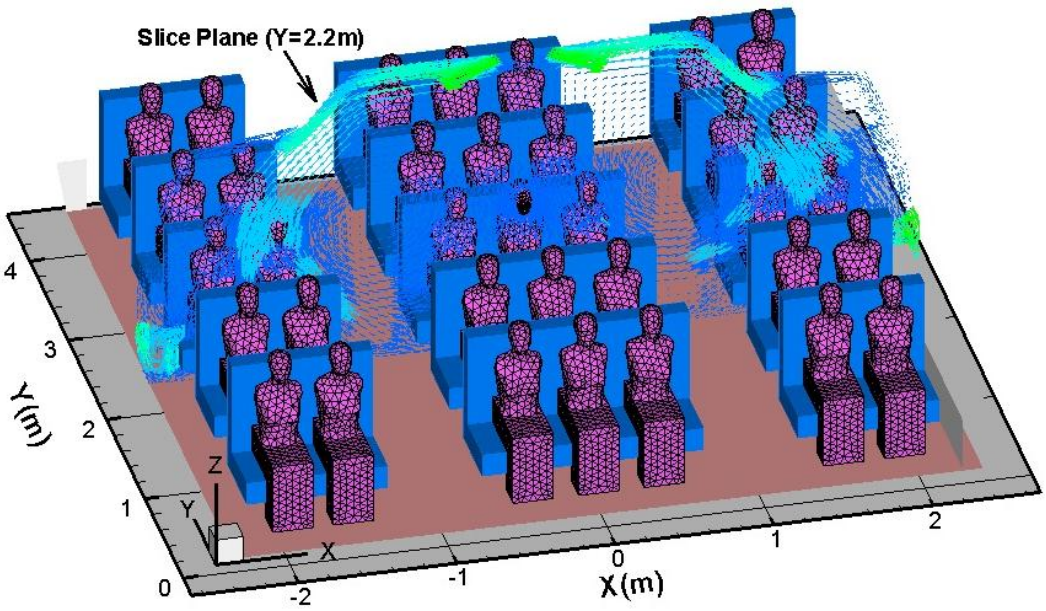
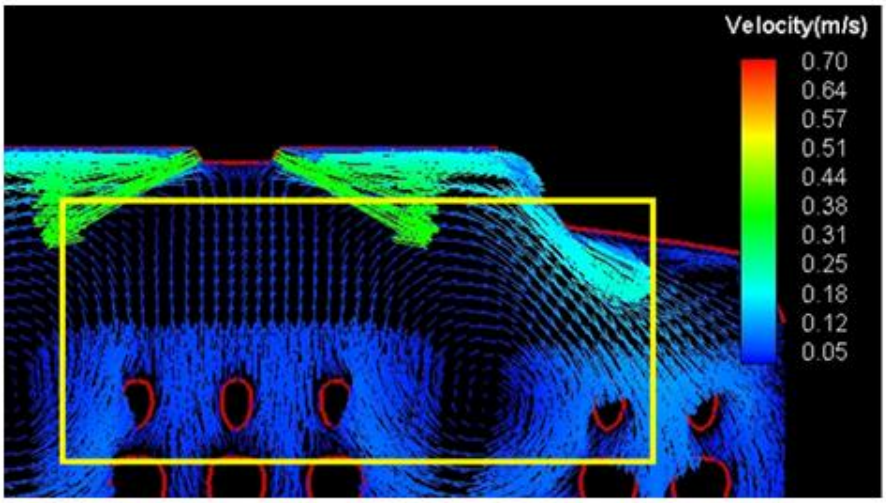
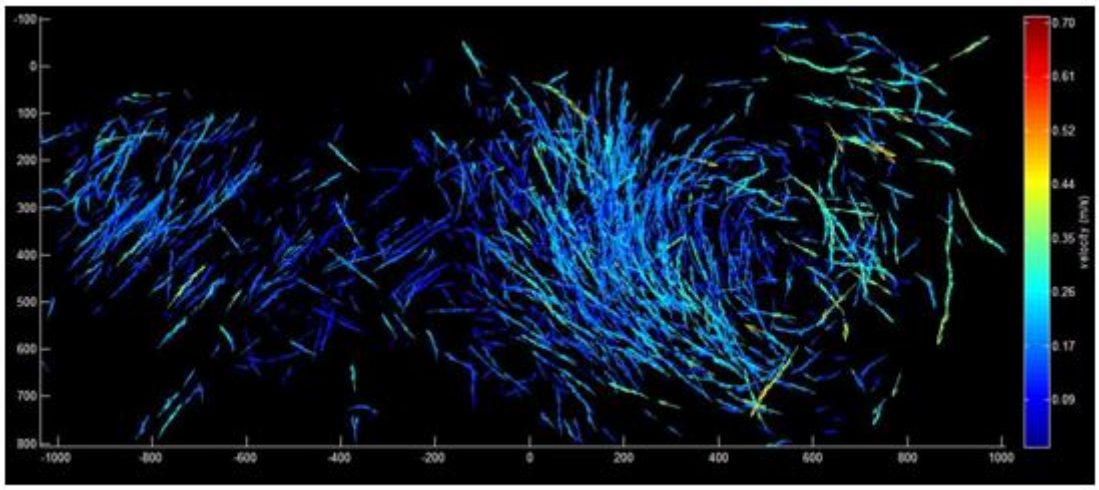


Figure 7-6. Sliced plane within simulated aircraft cabin for global airflow measurement.



(a) Simulated velocity field at experimental detected region



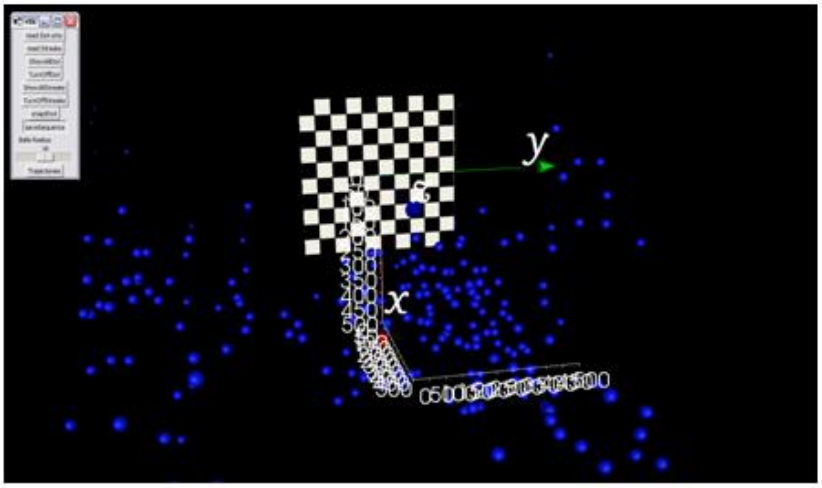
(b) Experimental detected velocity field

Figure 7-7. Compare simulated and experimental flow field at the sliced plane for Case 1.

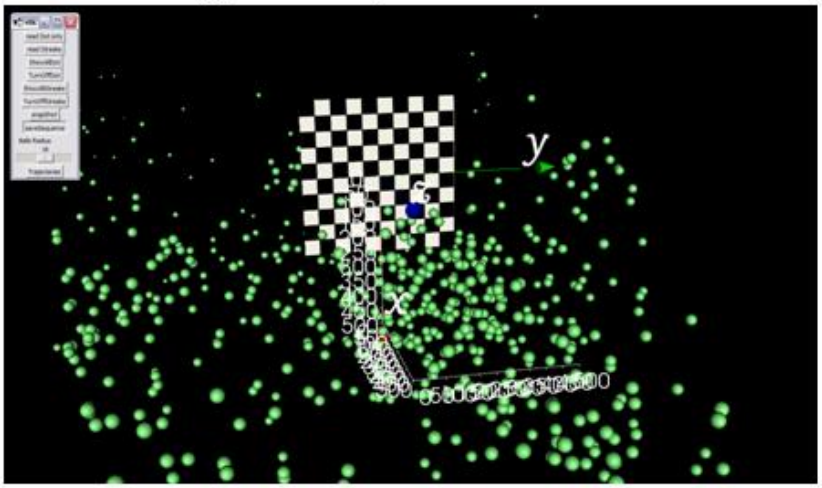
It is also observed from Figure 7-7 (b) that it is difficult to detect trajectories in the high velocity region, for example the upper region of the cabin close to the diffuser. This region is mainly controlled by the jet flow generated by the inlet diffuser as indicated in the simulation result. Two reasons might explain this phenomenon: First, because the bubbles are released from the diffuser, the bubbled density at this region is extremely high, and second, the air velocity at this region is very high. It has been explained in section 4.2.1 that the temporal tracking under such conditions is very difficult. As a result, little 2D trajectories are linked there. This is a

significant limitation of the current 2D temporal tracking algorithm. In addition, because of the poor illumination condition and low camera resolution, no short streaks are captured either. Therefore, object hybrid tracking cannot be processed here.

The detected particles are visualized in 3D volume frame by frame by a VTK based toolbox. These images are further combined into a short movie to reproduce the particle moving scenario. The resultant scenario could be observed from any arbitrary direction and distance as people desired. Figure 7-8 displayed the visualization result of Case 1. Figure 7-9 linked the same particles at different frames to construct their streamline using this toolbox.

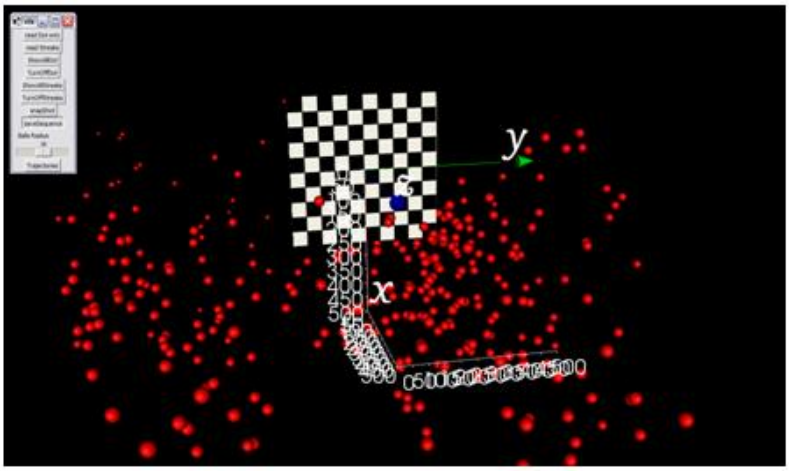


(a) The 1st frame; Ventilation rate = 80%



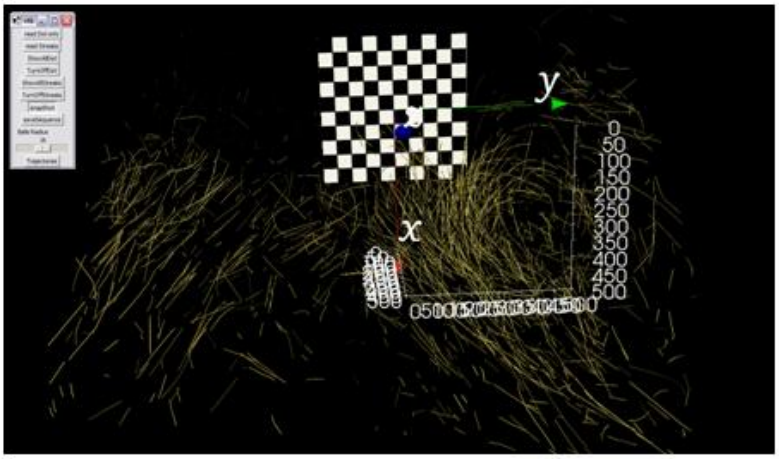
(b) The 15th frame; Ventilation rate = 80%

Figure 7-8. 3D particle visualization with VTK toolbox for Case 1.

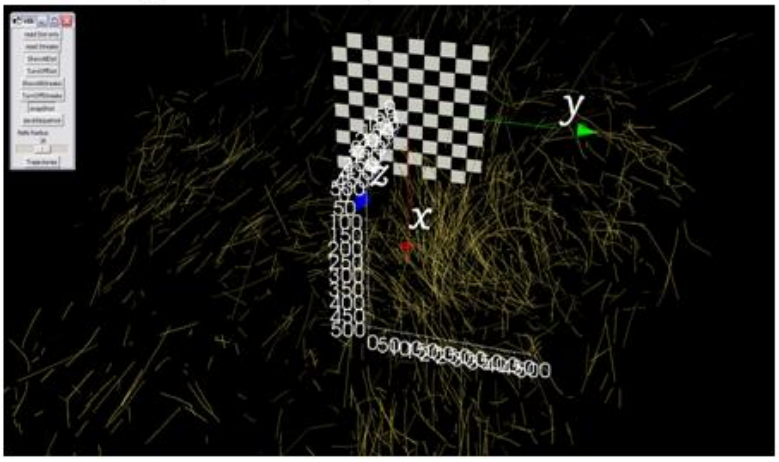


(c) The 30th frame; Ventilation rate = 80%

Figure 7-8. (cont.).



(a) Observation of the path-lines from front camera



(b) Observation of the path-lines from the upper-right camera

Figure 7-9. Visualization of the 3D path-lines with the VTK toolbox for Case 1.

In Figure 7-8, the color is assigned according to the frame index of the particles. The initial bubbles are rendered by blue color, then green and finally red. Figure 7-9 visualized all detected streamlines from two arbitrary directions: one from the front view and the other from the up-right view.

7.1.3 Case 2: Global level study at 100% ventilation rate

The similar post-processing is applied to Case 2 (100% ventilation rate) and Case 3 (120% ventilation rate) as well. The result of Case 2 is shown in Figure 7-10 to 7.15.

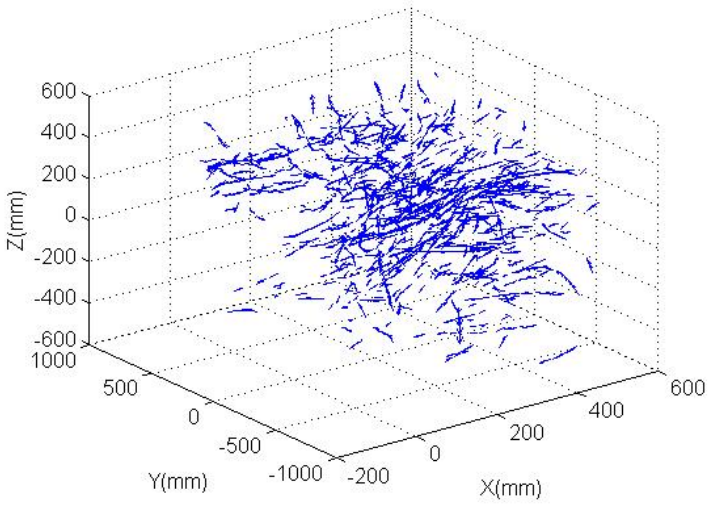
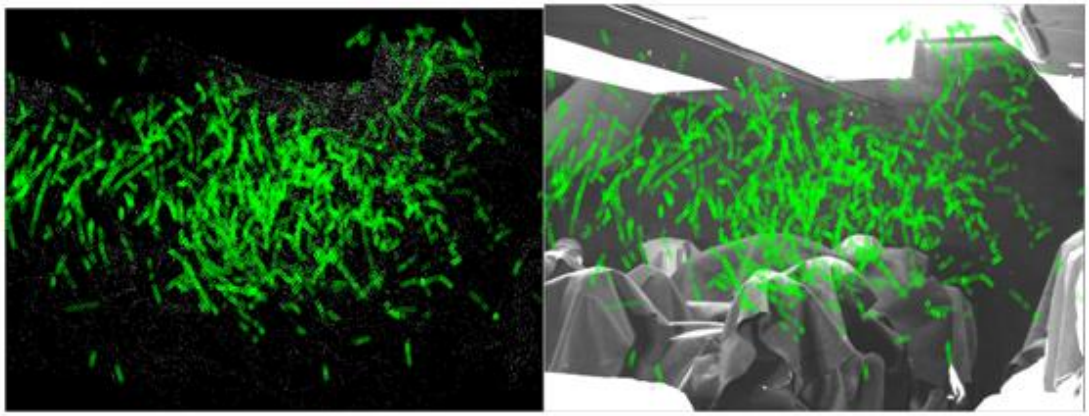
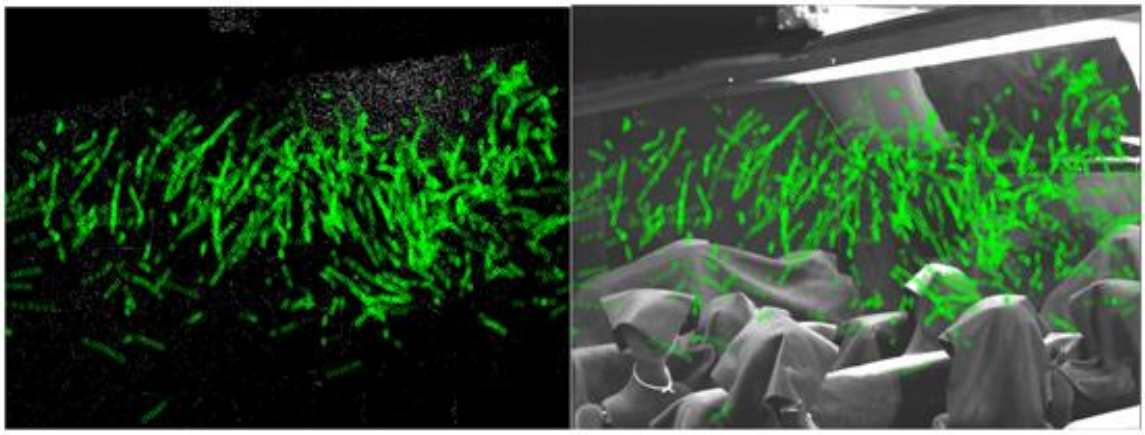


Figure 7-10. Detected 3D trajectories for Case 2 (global airflow at 100% ventilation rate).



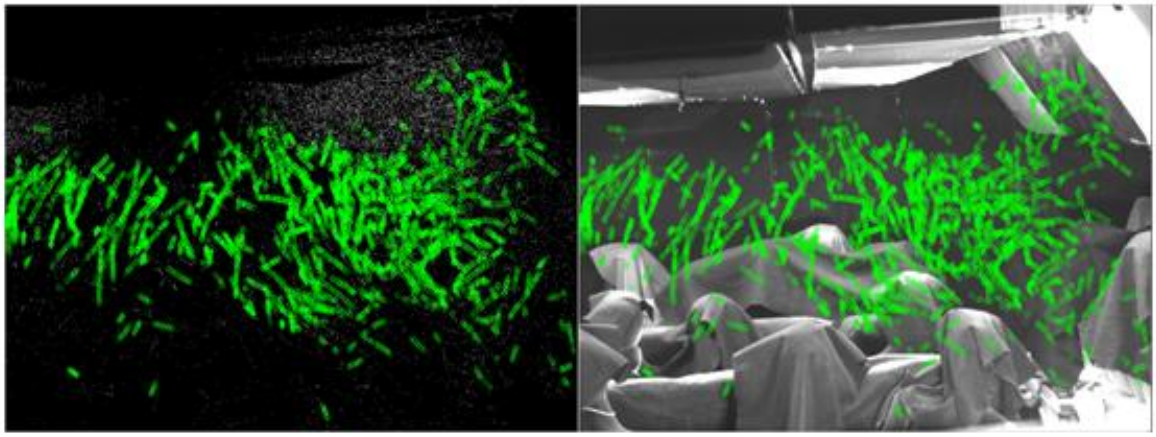
(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-11. Back projecting the detected 3D trajectories onto camera 1 for Case 2.



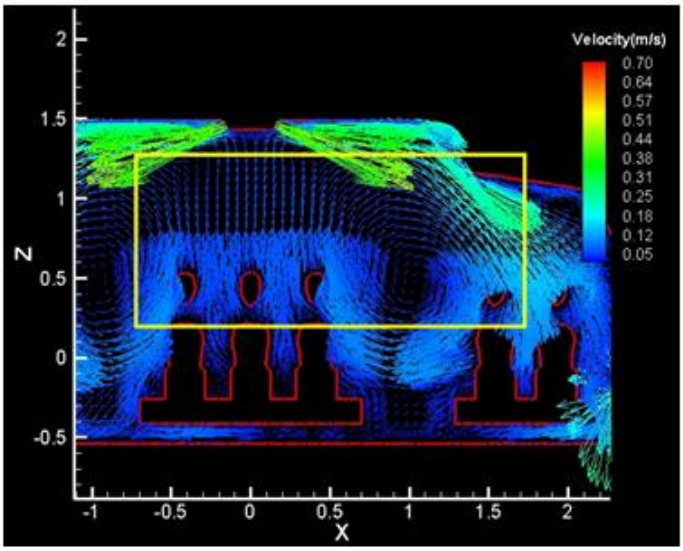
(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-12. Back projecting the detected 3D trajectories onto camera 2 for Case 2.

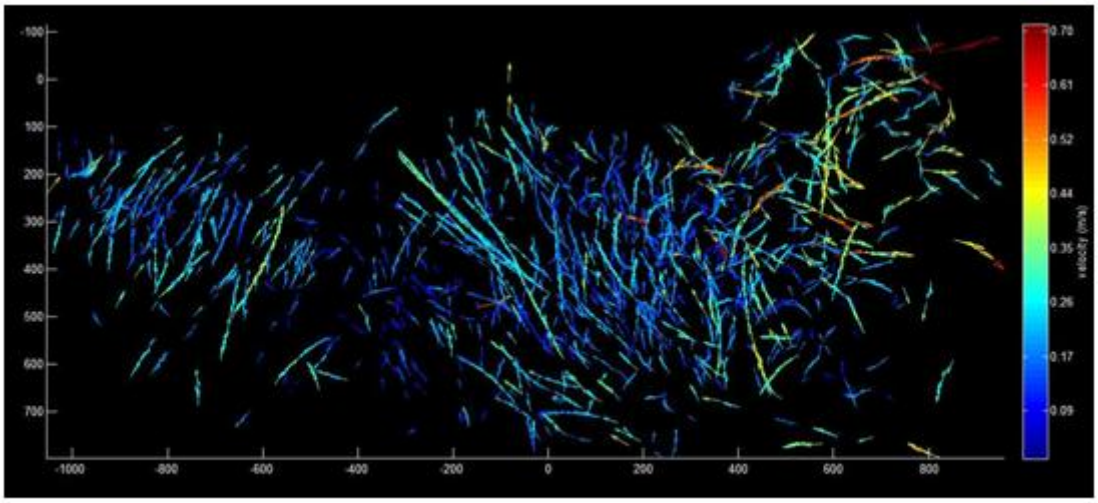


(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-13. Back projecting the detected 3D trajectories onto camera 3 for Case 2.

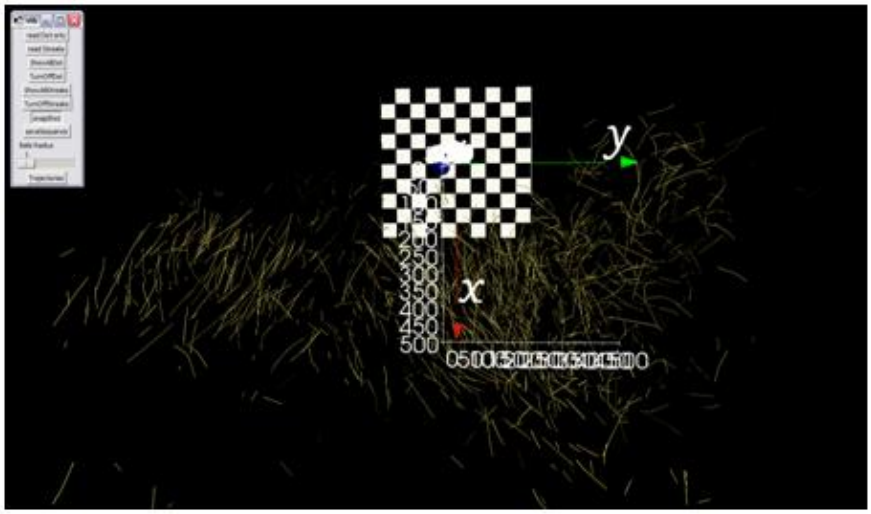


(a) Simulated velocity field at experimental detected region

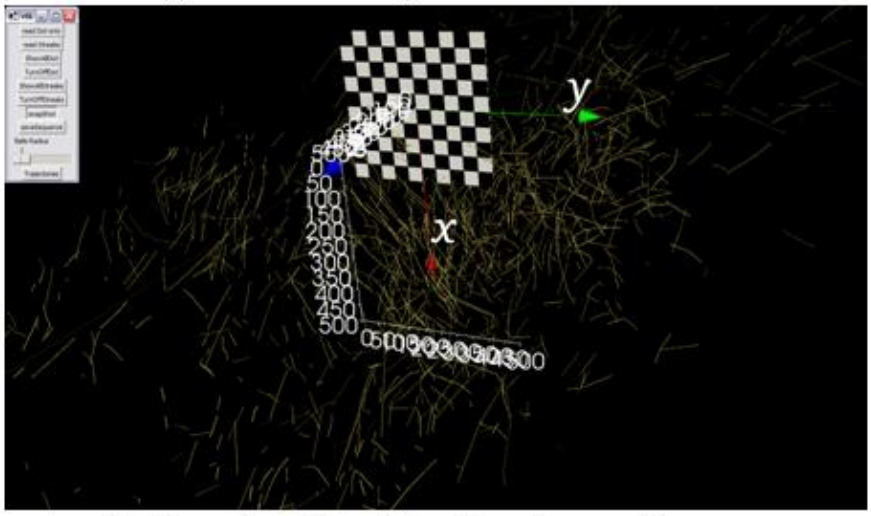


(b) Experimental detected velocity field

Figure 7-14. Compare simulated and experimental result at the sliced plane of Case 2.



(a) Observation of the path-lines from front camera



(b) Observation of the path-lines from the upper-right camera

Figure 7-15. Visualization of the 3D path-lines with VTK toolbox for Case 2.

7.1.4 Case 3: Global level study at 120% ventilation rate

The experimental and simulated result of Case 3 (120% ventilation rate) is shown as Figure 7-16 to Figure 7-22.

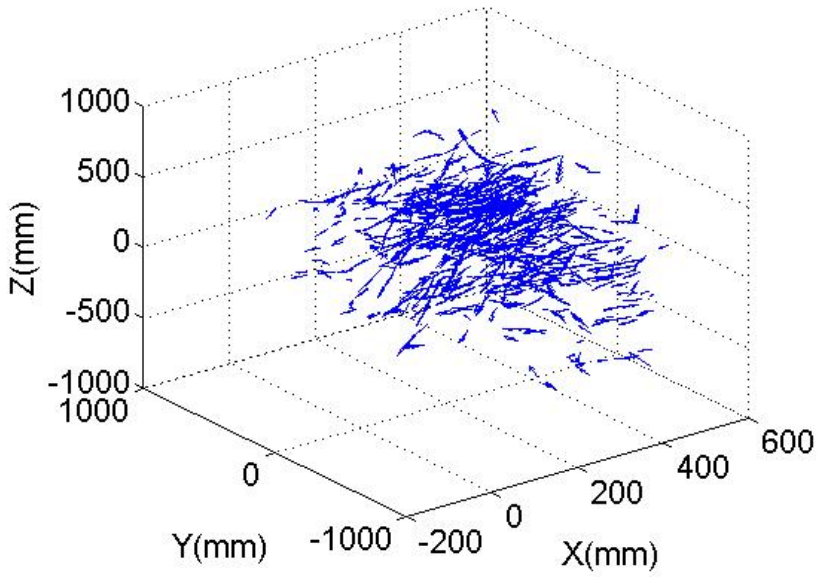


Figure 7-16. Detected 3D trajectories for Case 3 (global airflow study at 120% ventilation rate).

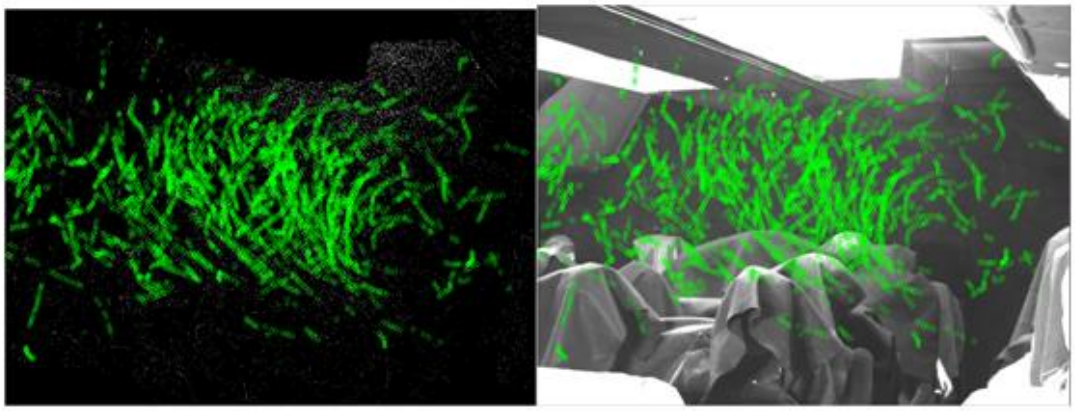
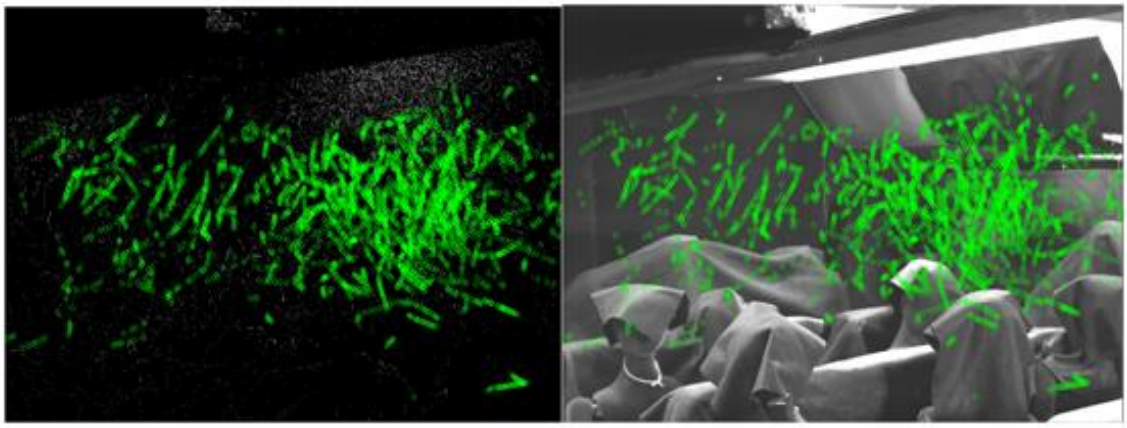
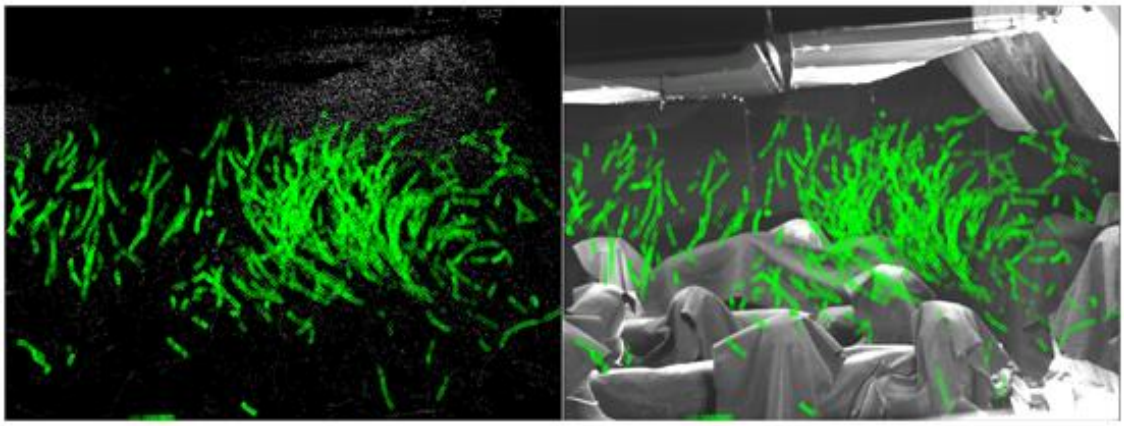


Figure 7-17. Back projecting the detected 3D trajectories onto camera 1 for Case 3.



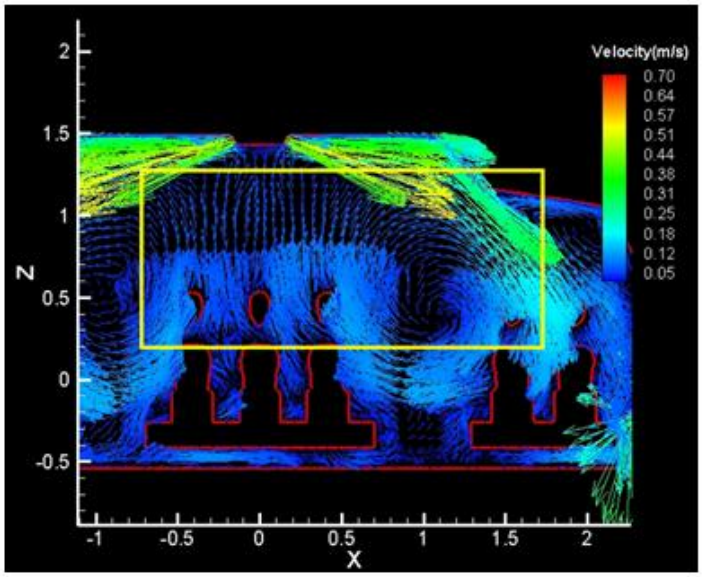
(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-18. Back projecting the detected 3D trajectories onto camera 2 for Case 3.

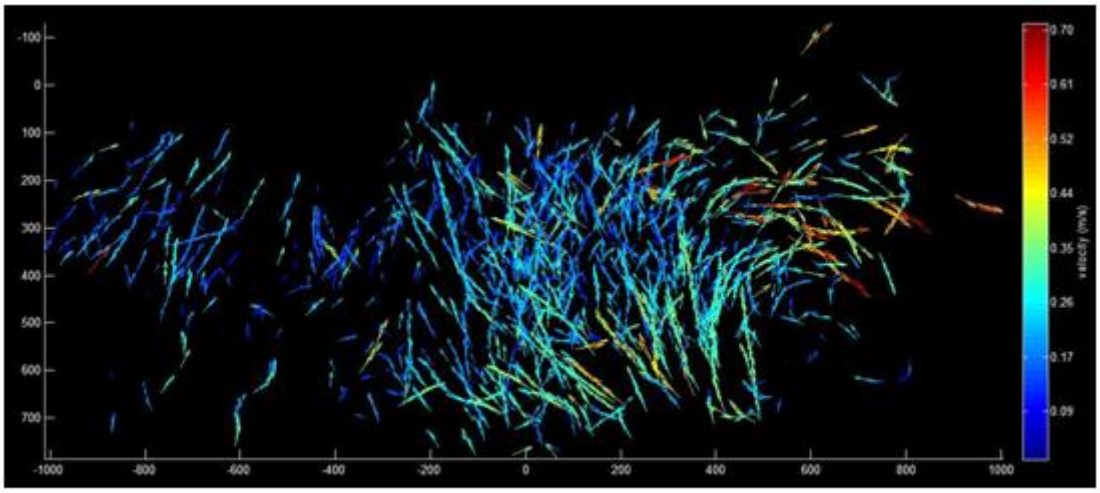


(a) Back projection on 2D trajectories (b) Back projection on background image

Figure 7-19. Back projecting the detected 3D trajectories onto camera 3 for Case 3.

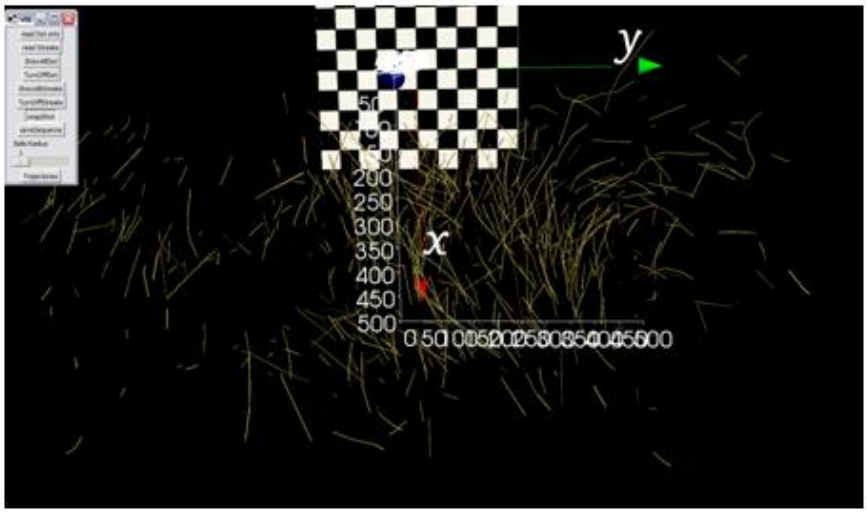


(a) Simulated velocity field at experimental detected region

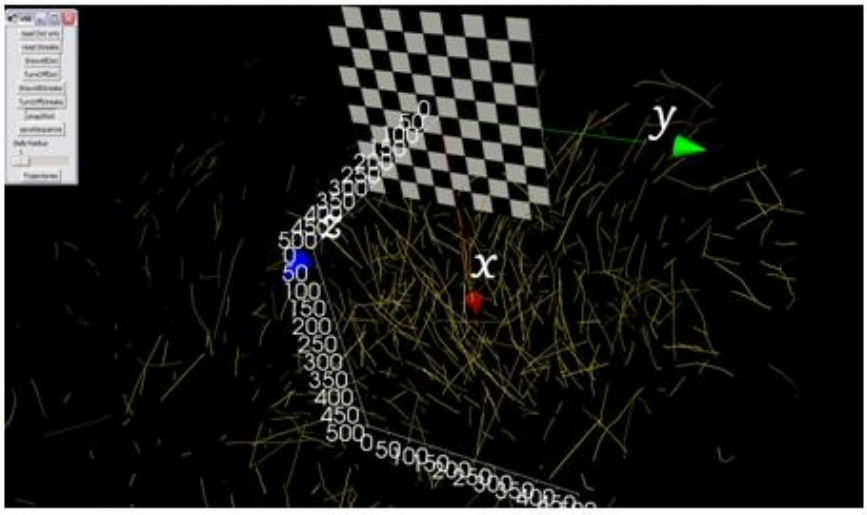


(b) Experimental detected velocity field

Figure 7-20. Compare simulated and experimental result at the sliced plane for Case 3.



(a) Observation of the path-lines from front camera



(b) Observation of the path-lines from the upper-right camera

Figure 7-21. Visualization of the 3D path line with VTK toolbox for Case 3.

From Figures 7-3 to 7-5, Figures 7-11 to 7-13, and Figures 7-17 to 7-19, it is observed that the projected bubble images agreed with the image processing result. Most of the projecting errors are less than 2 pixels. It means the system has achieved the desired accuracy under current camera configuration. Comparing the detected trajectories in three cases (Figures 7-7, 7-14 and 7-20), it is found that Case 1 has the highest bubble detecting rate. At 80% ventilation rate, it has the lowest air velocity field, which makes sense and is consistent with our expectation. According to previous discussions, lower velocity is an easier temporal tracking situation. But

more bubbles are detected in Case 3 than Case 2, although the latter case has a lower air velocity. This seems ridiculous, but in fact air velocity is not the only factor affecting the detecting rate. All other factors such as illumination, bubble density, camera parameters, etc., are key roles in the tracking process as well.

In regard to the velocity magnitude, the experimental detected bubble velocity is a little bit higher than the simulated result, especially for the high ventilation rate case (Case 3) as shown in Figure 7-20. The simulated result predicted that the high velocity field is only located at the diffuser region close to the ceiling, however, in experimental results a few high velocity bubbles are detected at the upper-right and lower-left region of the right vortex. Since no bubbles are detected at the lower-right region of the vortex due to the manikins' blocking, it is hard to conclude that the overall airflow field was under predicted by the simulated result or because of the geometry difference between the simulated and experimental manikins.

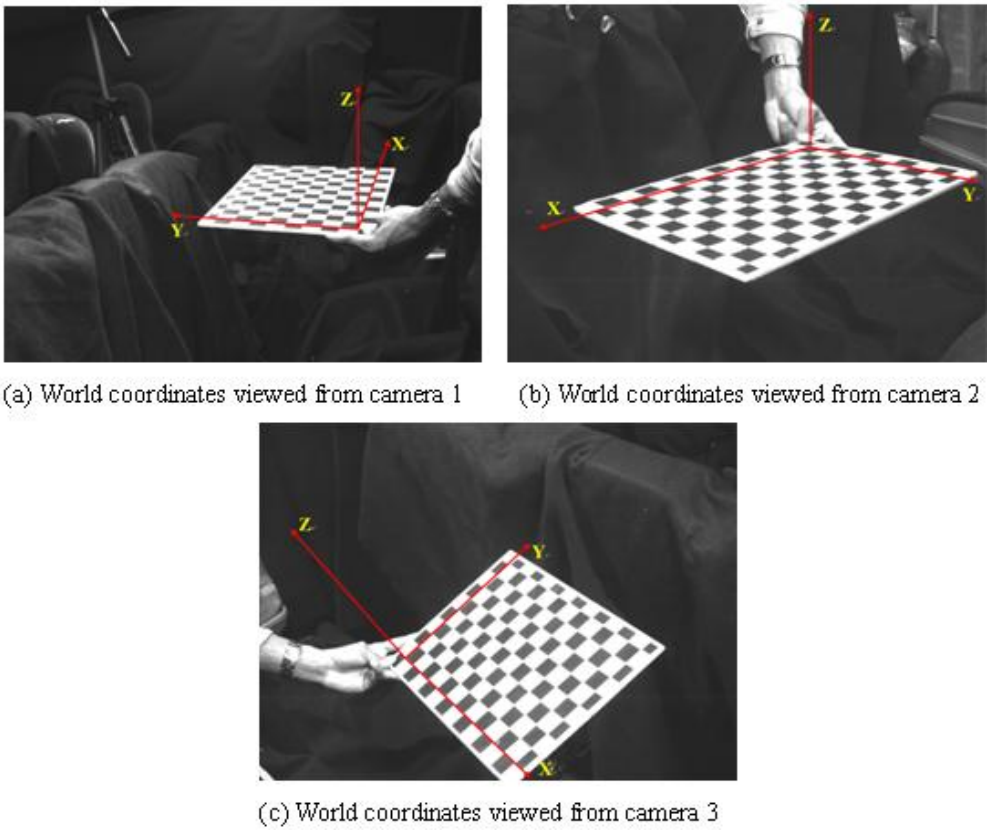
7.2 Experimental and numerical result at local level

The purpose of the local level experiment is to visualize the airflow pattern created by the passenger after his coughing or sneezing. In practice, the coughing or sneezing behavior is an unsteady jet flow phenomenon which happened within a few seconds. However, applying the HPTV system to study such a kind of airflow is extremely difficult because it is hard to produce enough seeding bubbles following the airflow within such a short period. Instead, a steady jet flow created by a small nuzzle installed at the manikin's mouth position was used to mimic the coughing process. We can take advantage of the steady-state assumption by combining the bubbles in all frames together so as to overcome the limitation of low bubble density. In other words, the shortage of the information in the spatial domain is compensated by those in the temporal domain. Through this approach, a higher bubble density could be obtained in local level cases and the uniform grid interpolation algorithm can be performed in the post-processing

procedure. In the next sections, we imposed a similar post-processing sequence as the global level cases. The result is discussed as follows.

7.2.1 Coordinate system of the experiment

The first thing of all is defining the world coordinate system. The board position of the last calibration image is selected as the X-Y plane, as shown in Figure 7-22. The Z axis is an upward direction. The target region is the volume below and above the calibration board. Three CCD cameras are installed at the side, front and top views of the volume respectively.



(a) World coordinates viewed from camera 1 (b) World coordinates viewed from camera 2

(c) World coordinates viewed from camera 3

Figure 7-22. Define the world coordinate system for local airflow measurement.

7.2.2 Case 4: Local level study without global ventilation

Case 4 served as a benchmark experiment to compare with later cases to evaluate the effect of global ventilation to local airflow. Measuring the local airflow field is quite challenging. The velocity range is much wider than previous cases (from 1 to 3) because the developing of the jet flow at this region. To obtain good measurement quality, both spatial hybrid and object hybrid approaches are demanded.

Figure 7-23 shows the raw images captured by the CCD cameras and the 2D trajectories associated with them after image processing. Both dot and streak shape bubbles are displayed as points in the trajectory images. Due to the high velocity airflow characters at the jet flow region, it is impossible to link the identical particles at adjacent frames.

Figure 7-24 performed a back projection of the detected particles. The left column images plotted the particles on the first frame images taken by all cameras. According to these images, about half of the streaks are detected and reconstructed by the algorithm. The right column images have plotted all of the detected 3D bubbles and streaks in all frames onto one image for each camera. The measurement volume has been divided into two regions: the jet flow region mainly consists of short streak-shape bubble images and the ambient region consists of dot-shape bubble images. Since the ventilation system is turned off in the benchmark experiment, the coughing jet impinged straight ahead onto the front seat. For those short streaks, colors have been used to visualize the velocity magnitude they represented. Red color streaks indicate the high velocity bubbles, while the blue color ones indicate the low velocity bubbles. The velocity decay character of the jet flow is clearly observed in back projection images. The velocity of bubbles in the ambient region is much smaller than those in the jet region. Rendering them according to velocity magnitude will result in all blue colors. Therefore, dot-shape bubbles are alternatively rendered according to their frame index. Blue color indicates the early frame while the red color indicates the late one.

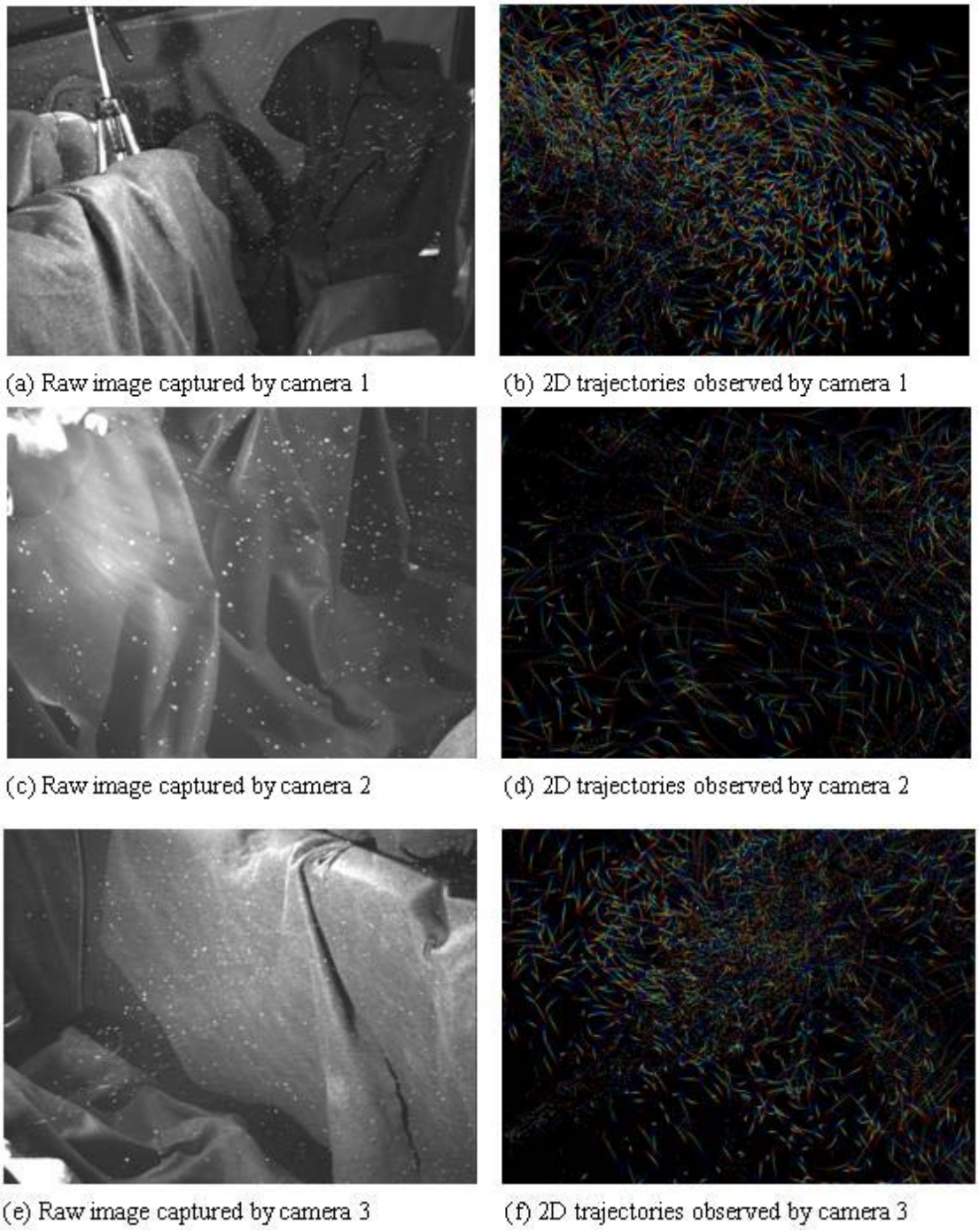
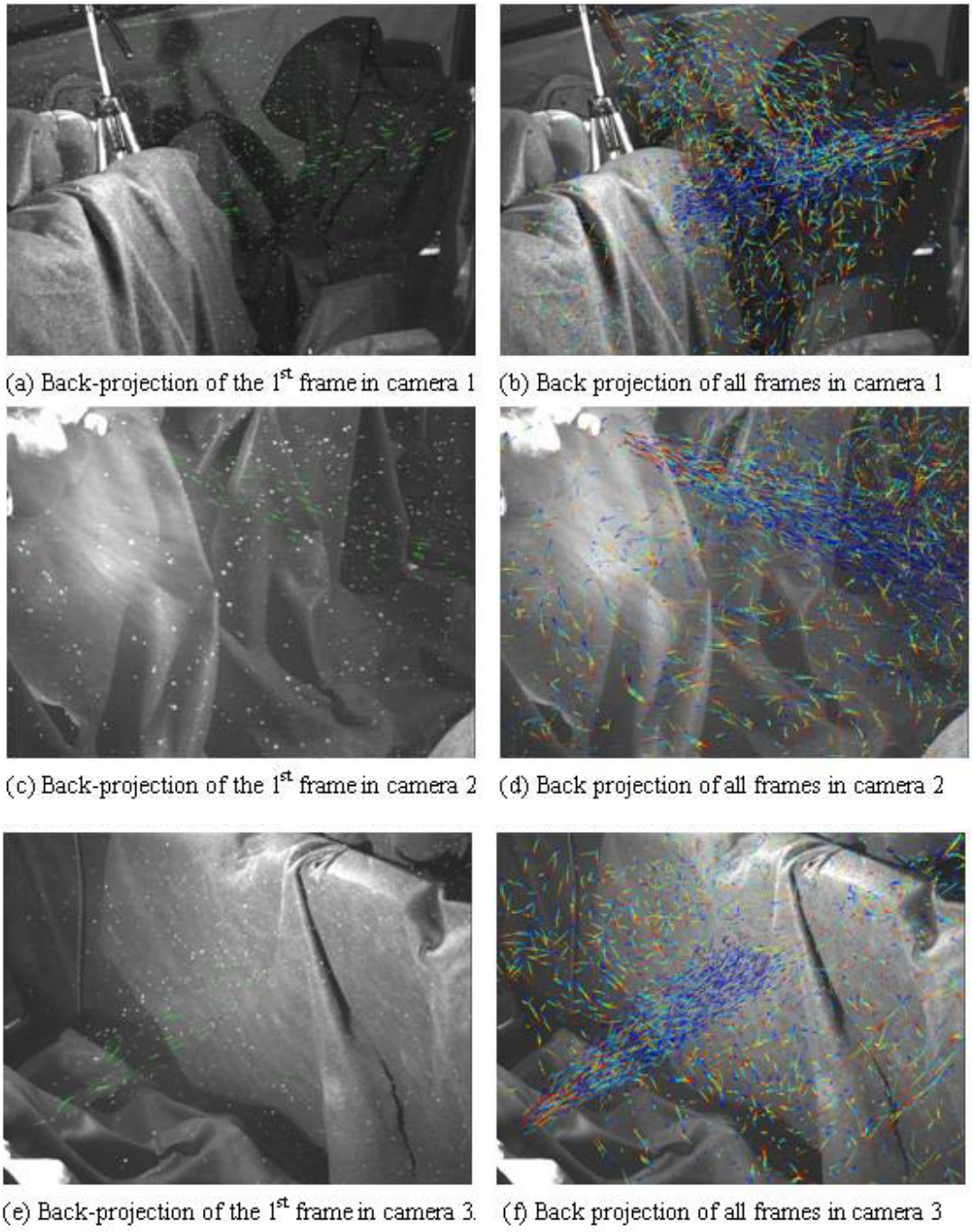


Figure 7-23. The raw image and 2D trajectories captured by the cameras (local environment).



(a) Back-projection of the 1st frame in camera 1 (b) Back projection of all frames in camera 1

(c) Back-projection of the 1st frame in camera 2 (d) Back projection of all frames in camera 2

(e) Back-projection of the 1st frame in camera 3. (f) Back projection of all frames in camera 3

Figure 7-24. The back projected 2D trajectories (local environment without ventilation).

The experimental result is compared with the simulated result after interpolated onto the uniform grids. A slice plane at the central location of the cabin (X=0m) was extracted as the target plane indicated in Figure 7-25.

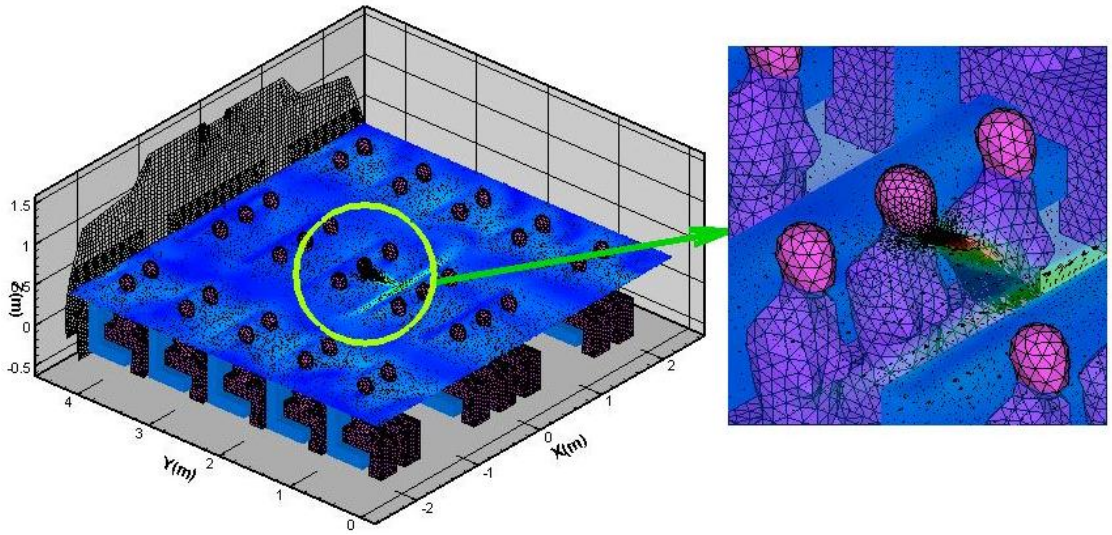
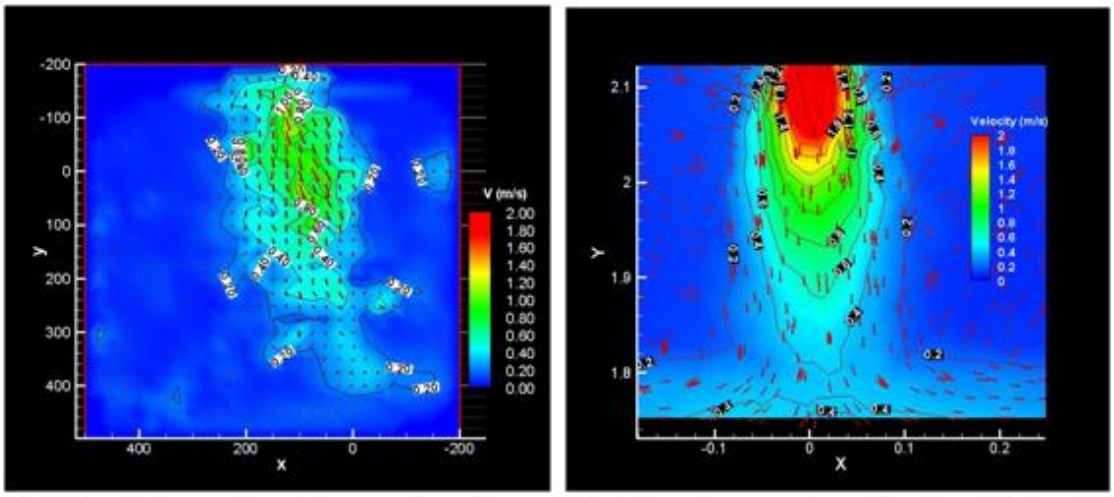
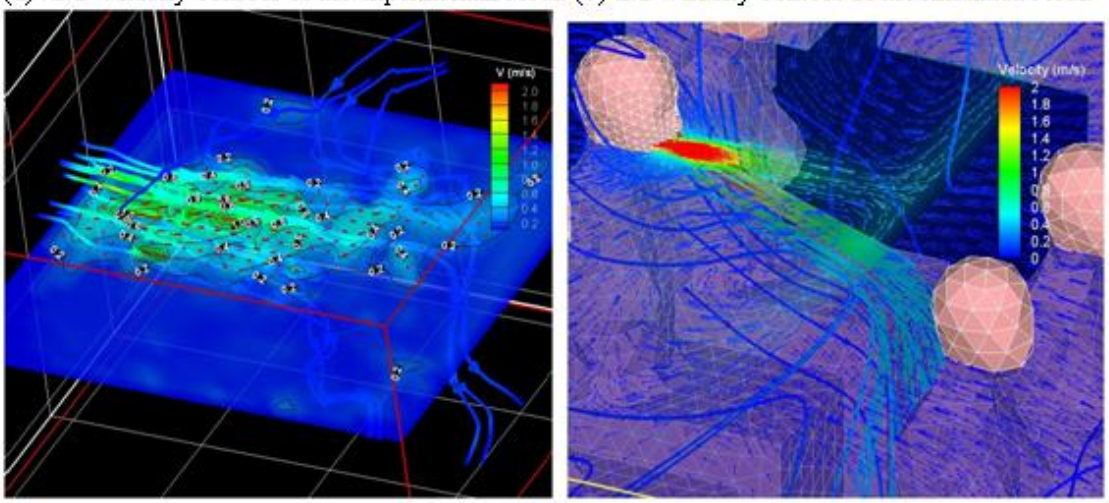


Figure 7-25. Sliced plane within simulated aircraft cabin at local environment.

Figure 7-26 compared the experimental and simulated velocity field at the sliced plane. It is observed from the visualized contour and vector maps that the experimental jet flow decayed quicker than the simulated result. This could be explained by two possible reasons. The boundary condition for the numerical simulation is not well defined. As discussed in Chapter 6, a constant and uniform distributed velocity field is assigned to the inlet of the jet. The velocity magnitude is defined according to the in situ measurement using a hot-wire anemometer. The sampling location is the center of the nozzle orifice. It is possible that this velocity profile overestimated the experimental data. The other possible reason is that the detected particle streaks are not dense enough to capture the whole profile of the jet flow, especially for those particles far away from the jet inlet. Because the initial velocity of the jet is relatively high, a lot of bubbles are broken before they propagated to faraway locations. Figure 7-26 (c), (d) compared the streamline of experimental data and the simulated data. According to the simulated data, the jet flow impinged on the front seat from a slightly downward direction which is consistent with the observation of the laboratory experiment. This trend is also captured by the particle tracking result as shown in Figure 7-26 (c).

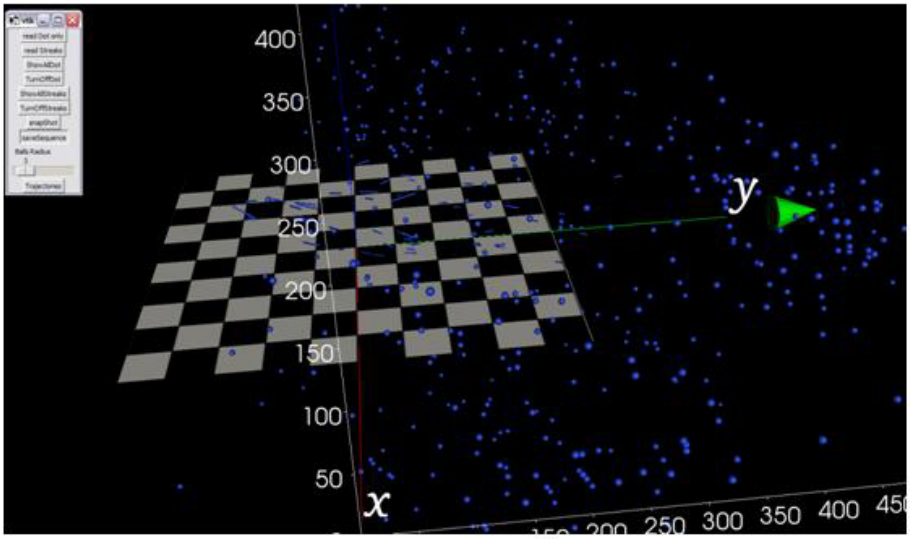


(a) 2-D velocity contour of the experimental result (b) 2-D velocity contour of the simulated result

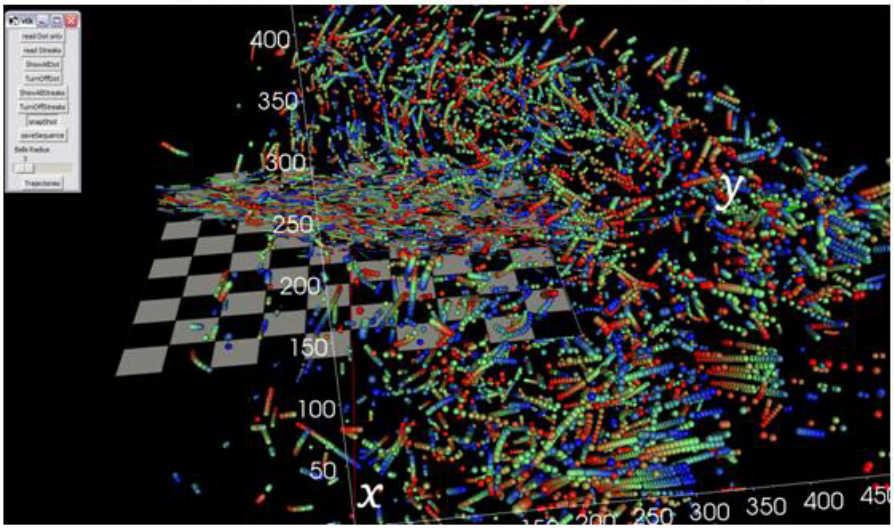


(c) Streamline generated from experimental data (d)Streamline generated from simulated data

Figure 7-26. Compare the experimental and simulated jet flow development.



(a) Visualize the hybrid particles at the 3rd frame using VTK toolbox.



(b) Visualize the hybrid particles of all frames using VTK toolbox

Figure 7-27. Particle visualization with VTK toolbox (without ventilation).

Figure 7-27 visualized the detected particles using the VTK toolbox we developed. Figure 7-27 (a) showed the streaks and dots particles detected at the 3rd frame, while Figure 7-27 (b) combined the particles images of all frames. The rendered color indicates the frame index of the particles: the blue ones indicate the earlier frames while the red ones reference to the latter frames. The moving direction of the low velocity particles (those particles beyond the jet flow region) is randomly distributed. Because there is no global ventilation applied, jet flow is not

bended or affected by the global airflow pattern. The images from different frames are combined together and produced a short movie to recur what happened during the whole recording period.

7.2.3 Case 5: Local level study with 80% ventilation rate

The previous procedures are applied to the following cases as well. For all of these local level experiments, an identical coordinates system is used. Figure 7-28 and Figure 7-29 show the detected 2D trajectories after image processing and the back projection result from 3D reconstructed data.

It is observed from Figure 7-28 that less dot shape particles are captured compared with the preview case. In Figure 7-28 (c), the camera view covers only part of the whole jet flow region. In the tracking algorithm, streak shape images from all three cameras are needed to reconstruct the 3D coordinates; therefore, those particles beyond the field view of camera 2 will not be recovered under the current algorithm. However, the dot shape particles reconstructed from the spatial hybrid matching algorithm can be tracked although only camera 1 and camera 3 captured them.

It is also noticed from Figure 7-29 that the global airflow pattern has significantly affected the local airflow pattern. The jet flow is bended to one side according to the experimental observation and the recorded images. The velocity decay is clearly indicated in the back projection images. The long red streaks are the particles with high velocity while the short blue streaks are low velocity ones.

Figure 7-30 compared the contour map, velocity profile and streamline of the experimental and simulated jet flow at the local environment with 80% ventilation rate. According to the result in Figure 7-30 (b), numerical simulation didn't capture the bended jet character. The CFD geometry was created by neglecting a lot of detail structure. The geometry model is symmetrical along the central X-plane and the inlet is exactly defined on that plane. Under this arrangement, it is impossible for CFD to predict the bending effect. However, the velocity magnitude profiles obtained through experiment and simulation are comparable as indicated in Figure 7-30 (c), (d). Their velocity distribution agreed with each other if neglecting

the bending effect. Figure 7-30 (e), (f) visualized the streamlines created based on the 3-D velocity field. The detected particles are also visualized with the VTK toolbox as shown in Figure 7-31.

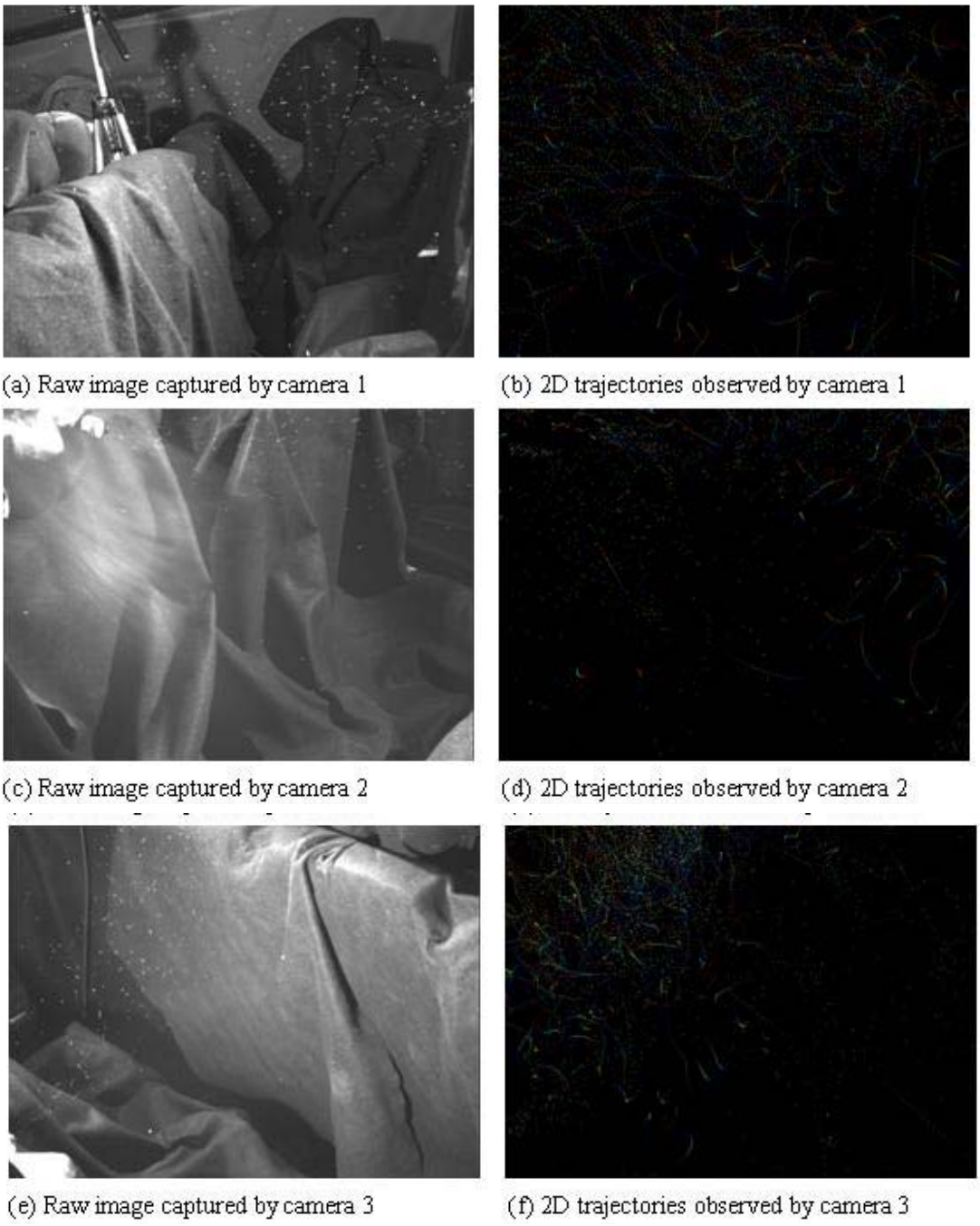


Figure 7-28. The raw image and 2D trajectories captured by the cameras (local environment with 80% ventilation rate).

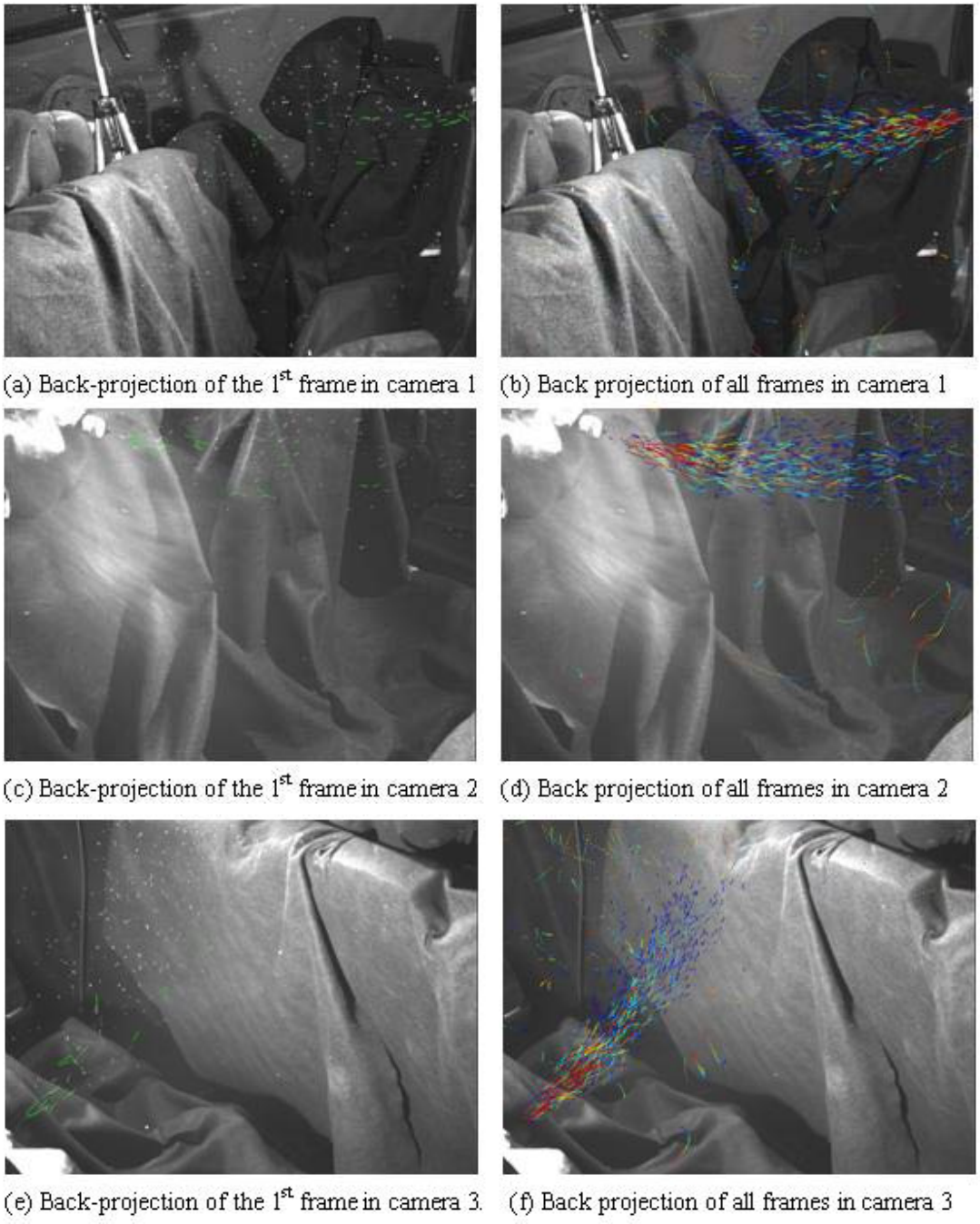
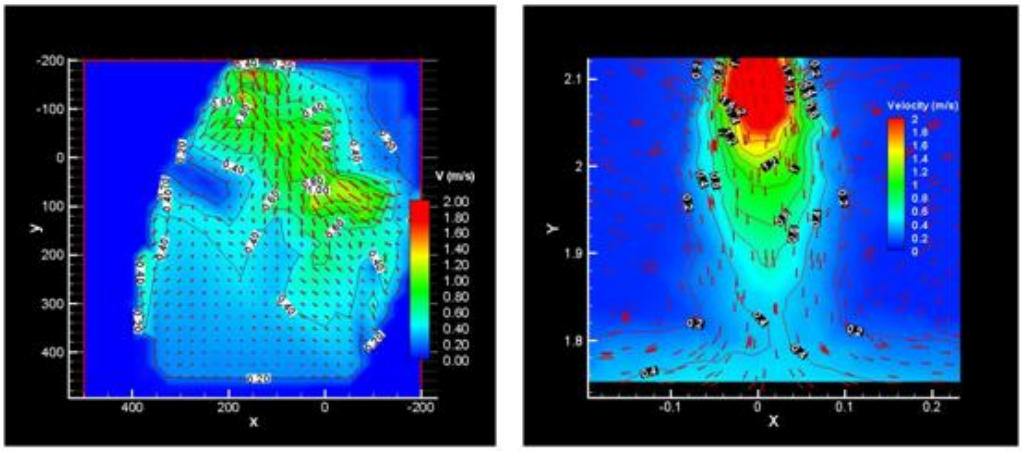
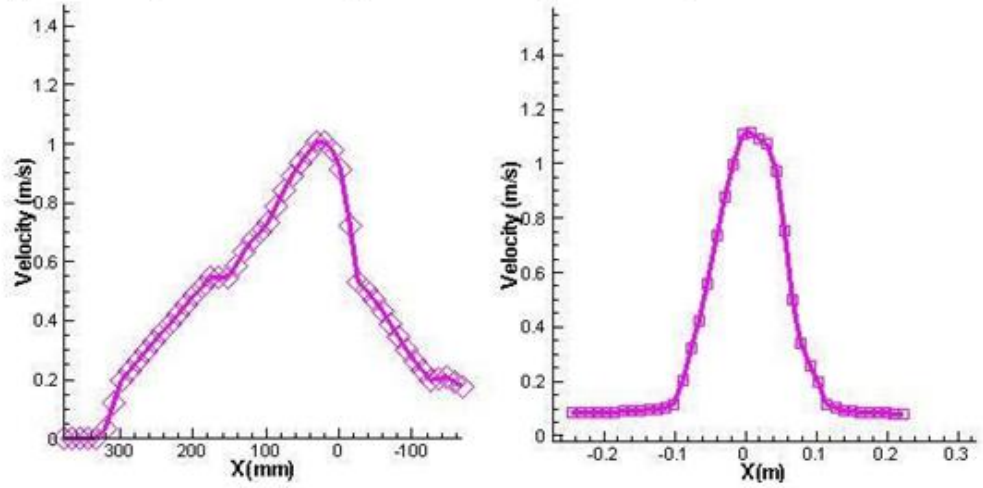


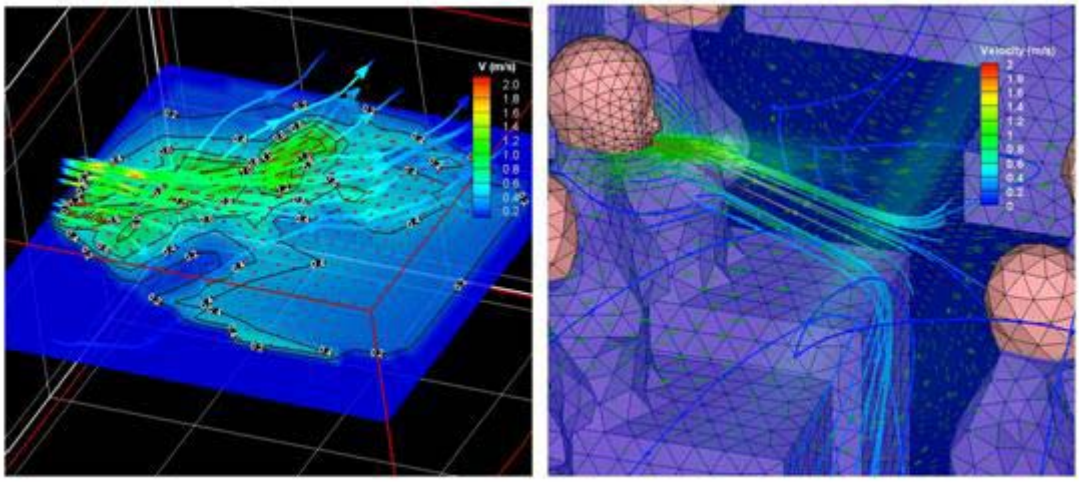
Figure 7-29. The back projected 2D trajectories (local environment with 80% ventilation rate).



(a) 2-D velocity contour of the experimental result (b) 2-D velocity contour of the simulated result

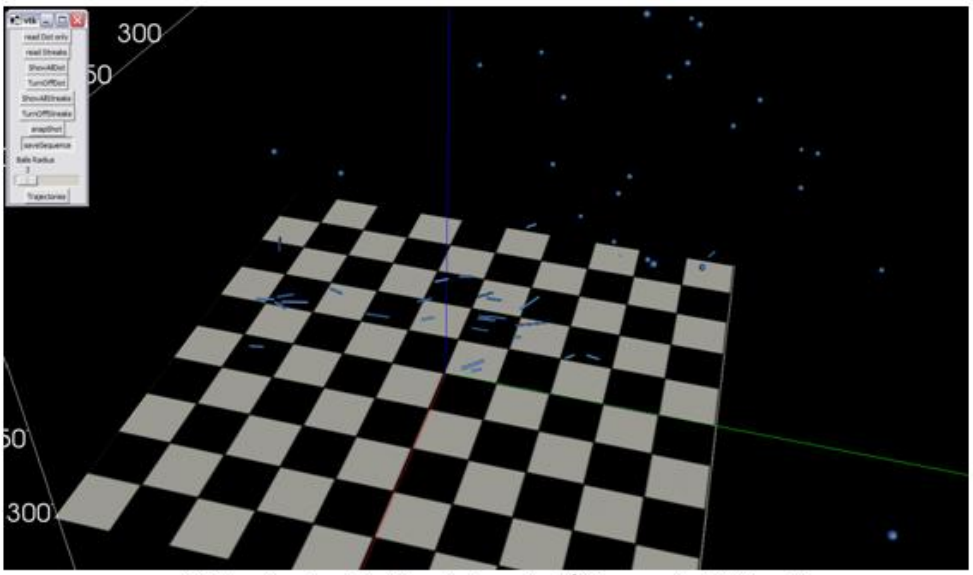


(c) Experimental velocity profile at Y=0mm (d) Simulated velocity profile at Y=2m

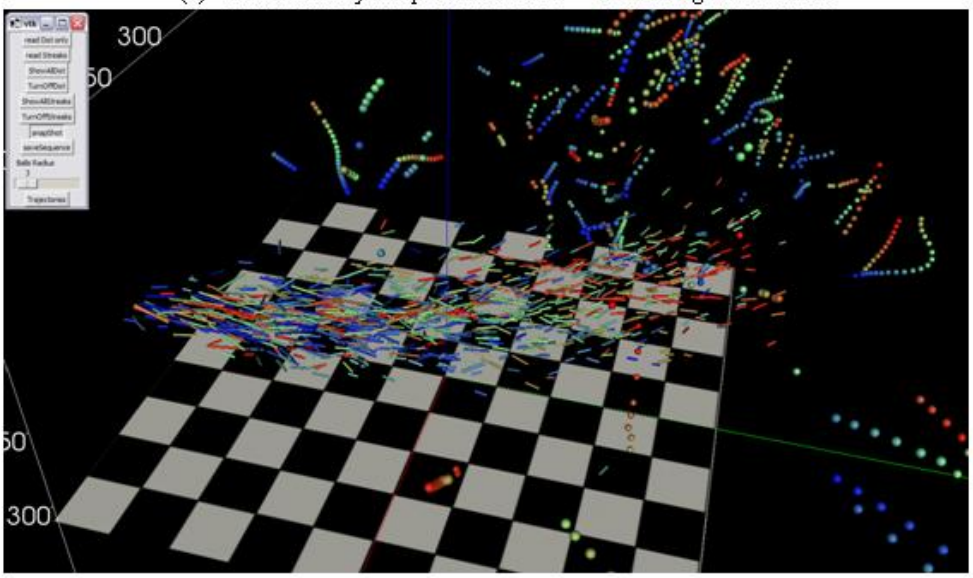


(e) Streamline generated from experimental data (f) Streamline generated from simulated data

Figure 7-30. Compare the experimental and simulated jet flow development with 80% ventilation rate.



(a) Visualize the hybrid particles at the 7th frame using VTK toolbox.



(b) Visualize the hybrid particles of all frames using VTK toolbox

Figure 7-31. Particle visualization with VTK toolbox (80% ventilation rate).

7.2.4 Case 6: Local level study with 100% ventilation rate

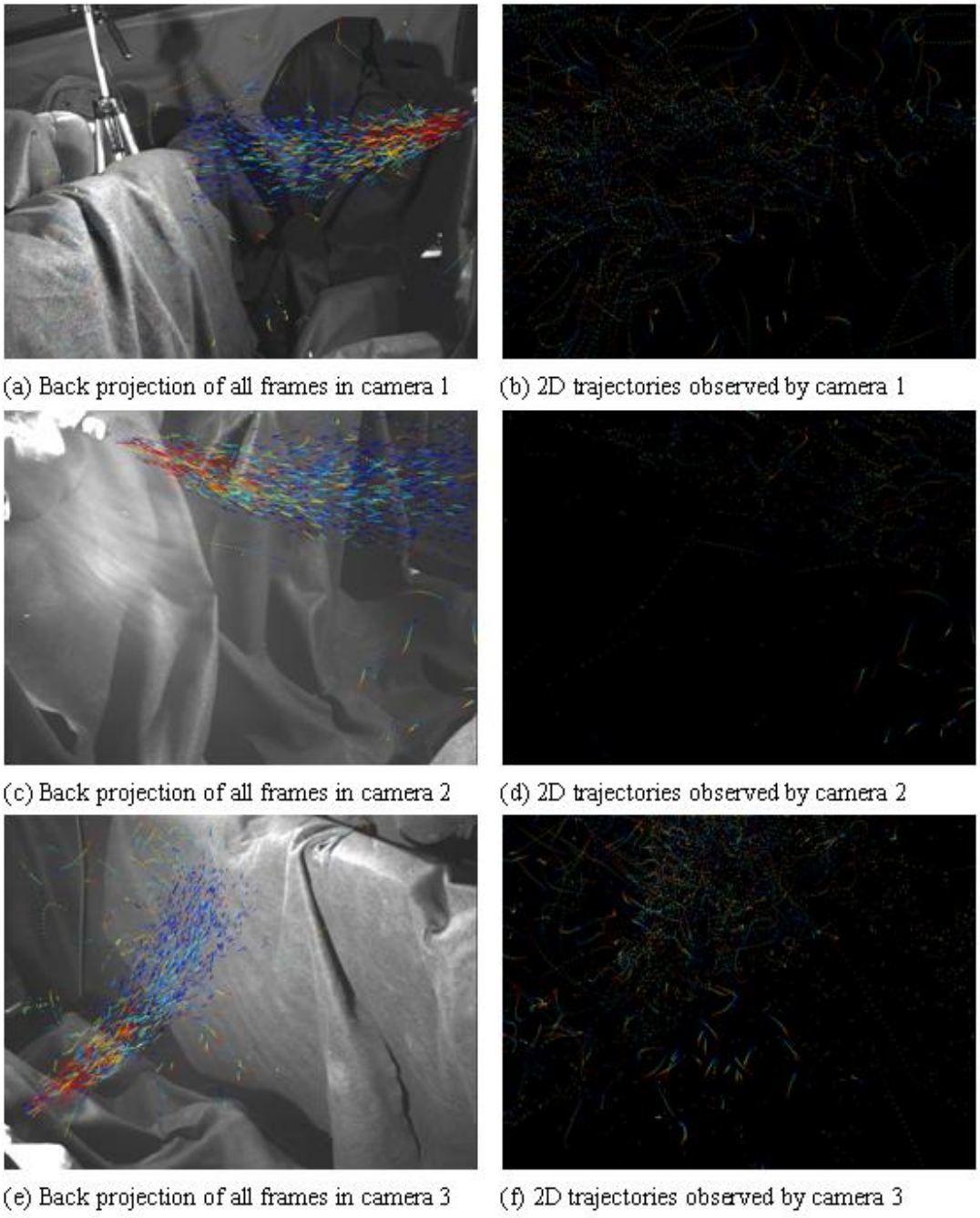
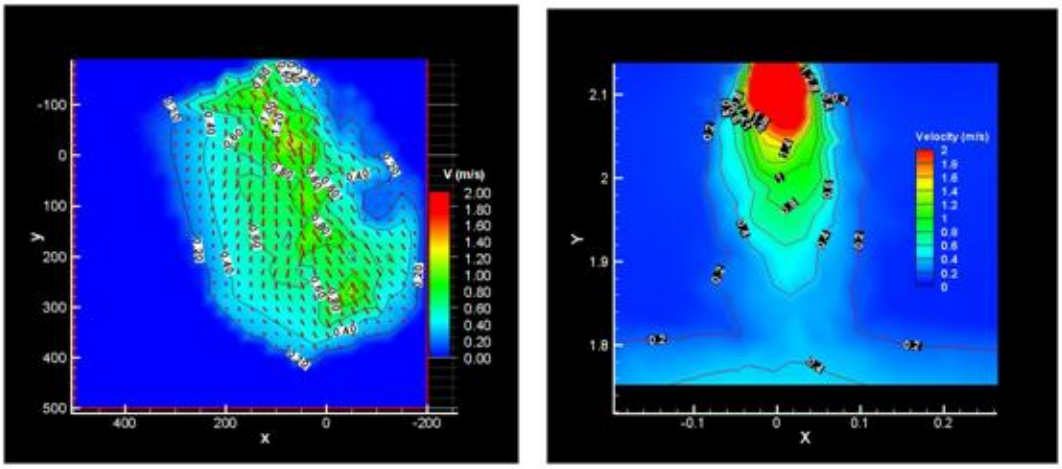
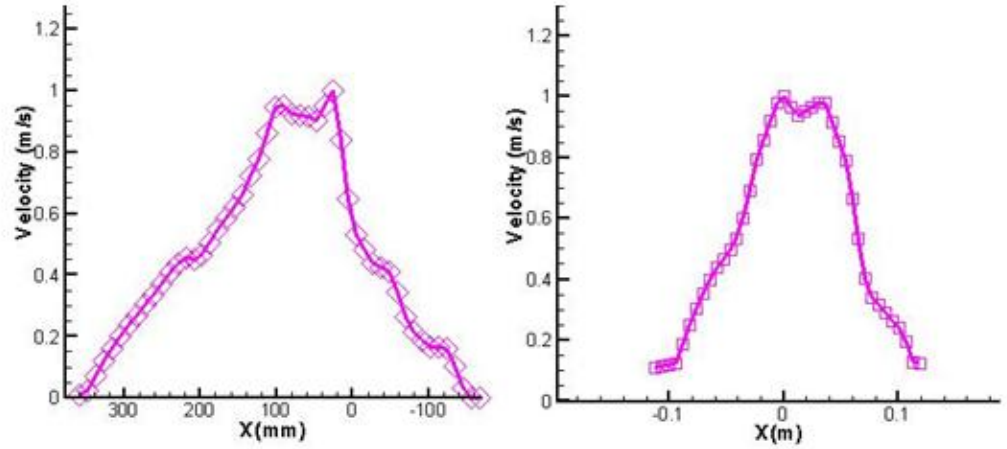


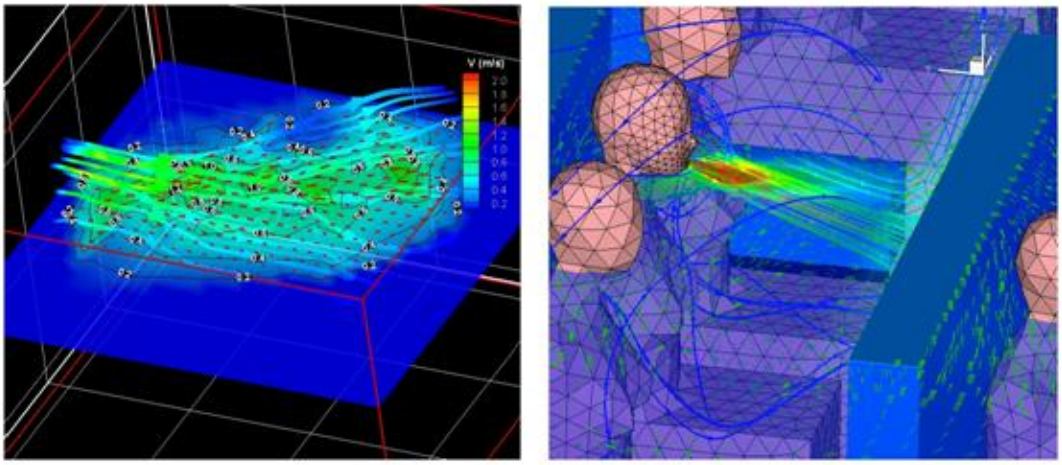
Figure 7-32. Compare 2D trajectories from back projection and image processing (local environment with 100% ventilation rate).



(a) 2-D velocity contour of the experimental result (b) 2-D velocity contour of the simulated result

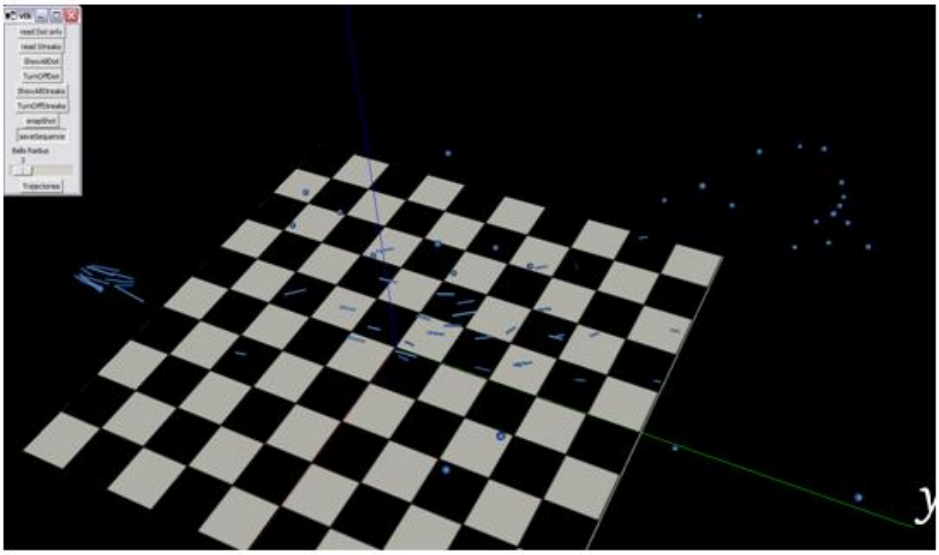


(c) Experimental velocity profile at Y=0mm (d) Simulated velocity profile at Y=2m

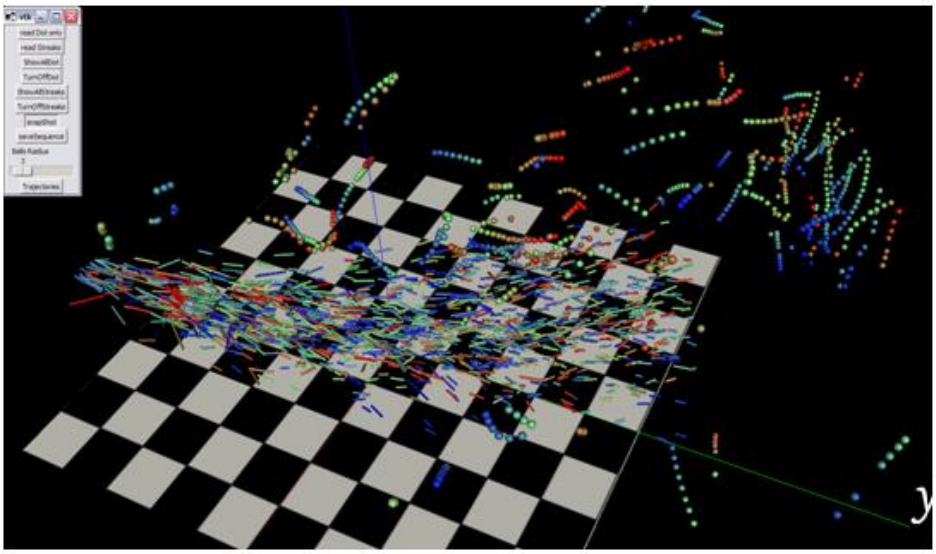


(e) Streamline generated from experimental data (f) Streamline generated from simulated data

Figure 7-33. Compare the experimental and simulated jet flow development with 100% ventilation rate.



(a) Visualize the hybrid particles at the 7th frame using VTK toolbox.



(b) Visualize the hybrid particles of all frames using VTK toolbox

Figure 7-34. Particle visualization with VTK toolbox (100% ventilation rate).

Figures 7-32 to 7.34 repeated the same visualization procedure as what has been done in Case 5. The results we obtained in these two cases are very similar. The difference of the flow direction between the measured and the predicted data is significant as shown in Figure 7-33 (e), (f). However, the velocity magnitude profile across the jet flow is comparable as indicated in Figure 7-33 (c), (d) except that the experimental jet flow spread wider than the simulated one.

7.2.5 Case 7: Local level study with 120% ventilation rate

Figures 7-35 and 7.37 show the result at 120% ventilation rate. The results are similar to previous cases.

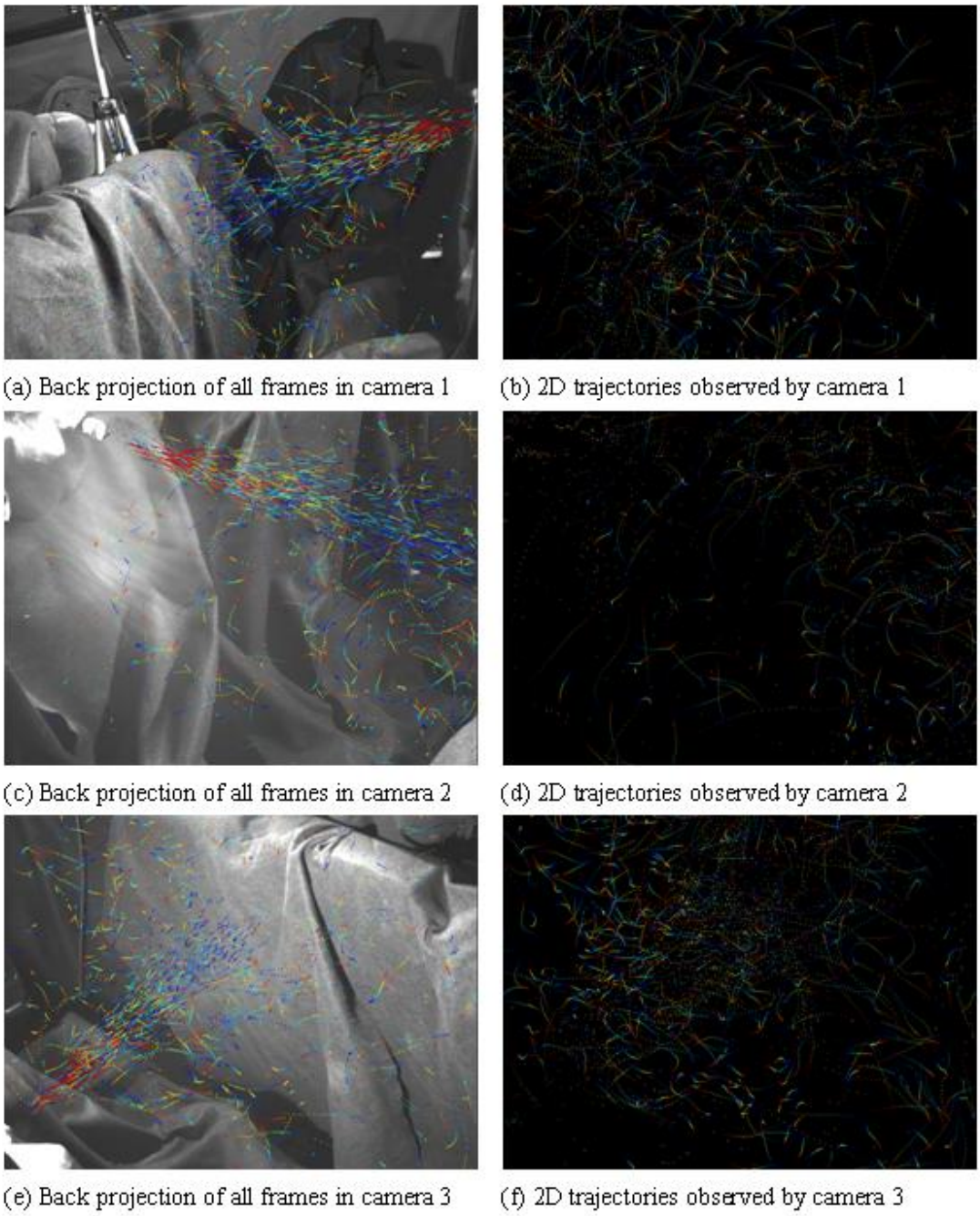
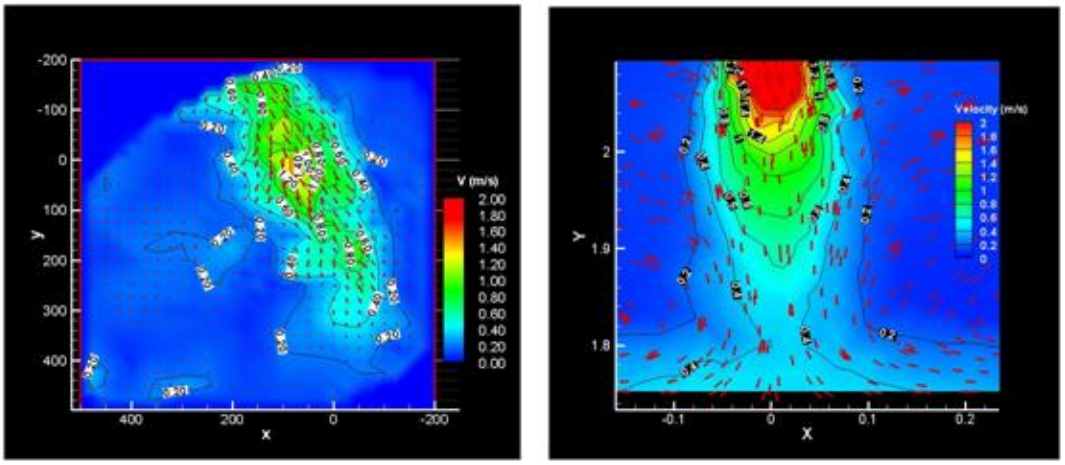
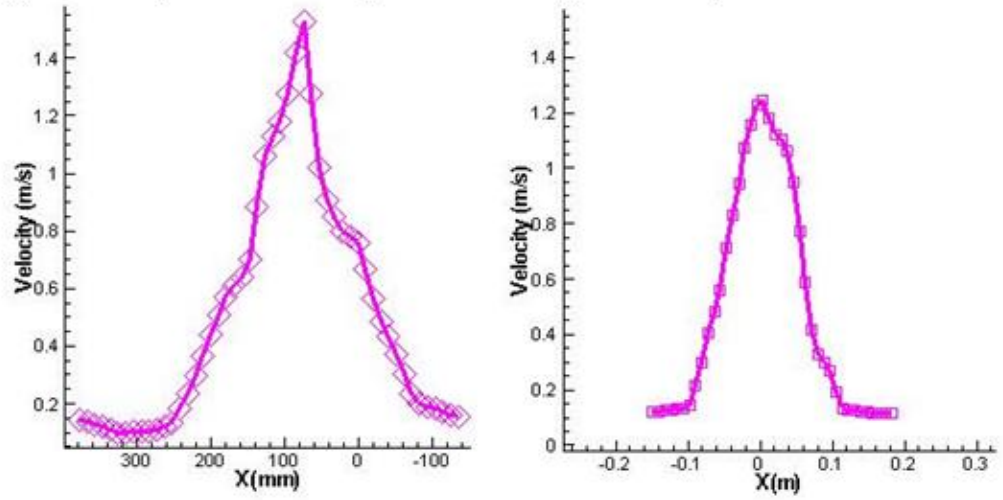


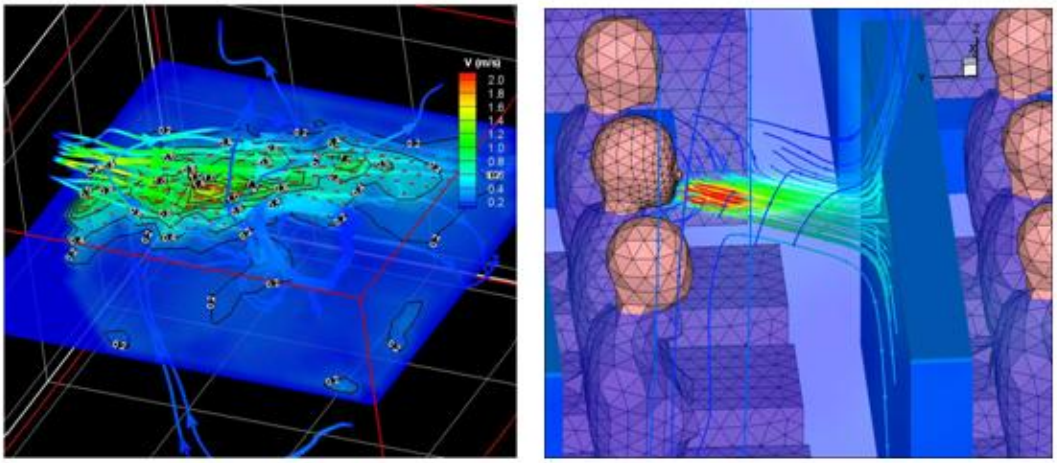
Figure 7-35. Compare 2D trajectories from back projection and image processing (local environment with 120% ventilation rate).



(a) 2-D velocity contour of the experimental result (b) 2-D velocity contour of the simulated result

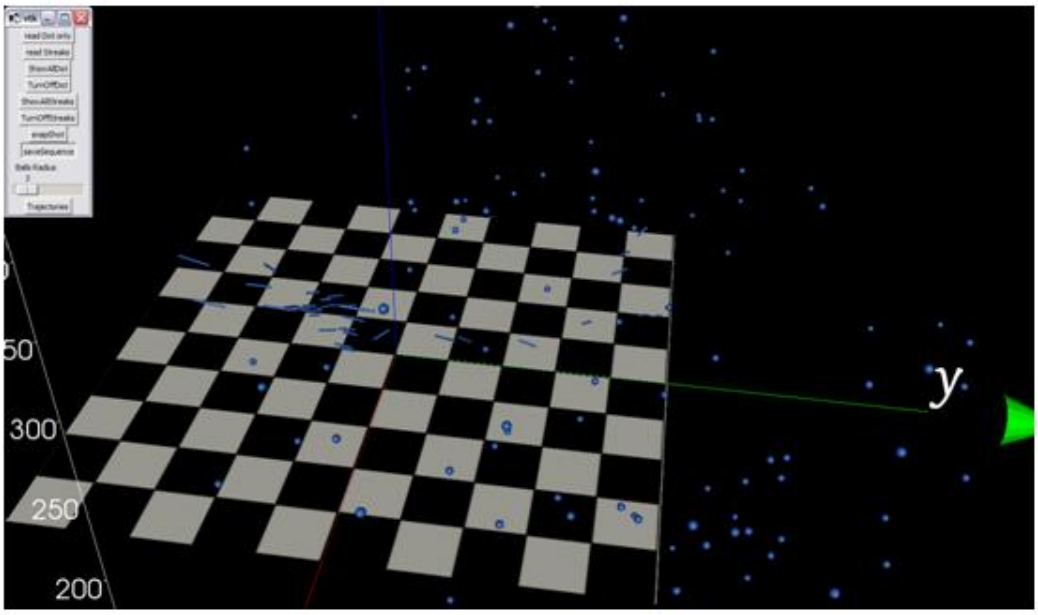


(c) Experimental velocity profile at Y=0mm (d) Simulated velocity profile at Y=2m

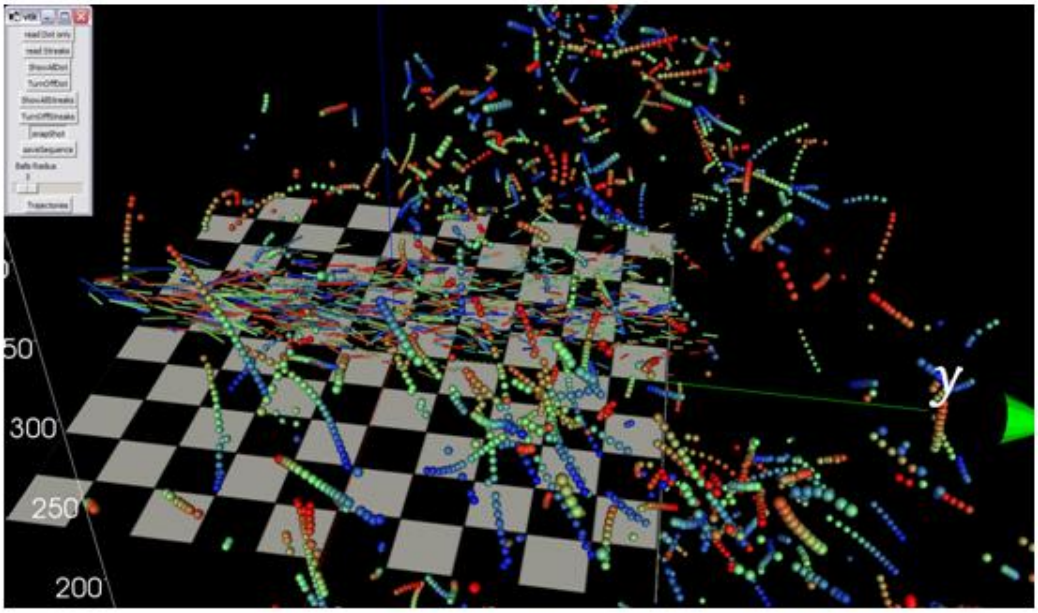


(e) Streamline generated from experimental data (f) Streamline generated from simulated data

Figure 7-36. Compare the experimental and simulated jet flow development with 120% ventilation rate.



(a) Visualize the hybrid particles at the 7th frame using VTK toolbox.



(b) Visualize the hybrid particles of all frames using VTK toolbox

Figure 7-37. Particle visualization with VTK toolbox (120% ventilation rate).

7.3 Summary

In this chapter, the HVPT system has been successfully applied to access the global and local airflow field within a full-scale aircraft cabin. A post-processing procedure was established and strictly followed through all experimental cases. It includes self-validation, uniform grid interpolation numerical result validation, and VTK-based 3D visualization.

Self-validation was performed through particle back-projection. We found that all of the projected trajectories could find their corresponding images in at least two cameras. Therefore the HPTV system has achieved the desired accuracy under current camera configuration.

At global level experiments, more than 1,000 3D trajectories had been detected for each case. We compared the trajectory map and velocity magnitude between the experimental and simulated result and found good agreement. Two large vortexes produced by the diffusers were detected in both methods. However, the particle density in the experiment was not high enough to interpolate a meaningful contour map.

At local level experiments, more than 2,000 3D streaks and 6,000 3D trajectories detected in 30 frames were combined together to obtain the steady velocity field. The particle density was high enough to interpolate the result onto an uniform grid, The mapped velocity contour and streamline were compared with the CFD simulation result and showed good agreement on velocity magnitued. But there was still significant difference on flow direction. This difference was explained by the imperfectly defined boundary conditions and geometry.

The particle trajectories of all cases were visualized with the VTK toolbox. With this powerful toolbox, the whole particle moving scenario was observed from arbitrary direction and distance. A short movie based on this observation was also produced.

CHAPTER 8 CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions

A wide range of indoor air quality problems, including headaches, chronic fatigue and drowsiness can often be accused by poor indoor air quality. To better understand the pollutant transport mechanism and further control the route of disease transmission, one of the priorities is to obtain the actual airflow pattern carrying the pollutant within the indoor environment. Current research reviewed the available experimental techniques including point-wise and whole-field measurement equipment. Among them, particle image velocimetry (PIV) and particle tracking velocimetry (PTV) are useful tools for indoor air quality applications because of their non-intrusive feature and multi-points accessing capability. However, PIV only has 2D tracking capability and relies on laser as the illumination source. PTV can track in three dimensions with high speed CCD cameras for high velocity flow measurement, in which case a laser light source is necessary. For the equipment using a laser as the illumination source, the detectable volume is usually small because of the light source limitation.

This research group has been developing large scale non-intrusive airflow measurement techniques using normal light since 1997. The earlier algorithm (VPSTV) used two cameras to capture the particle displacement during short exposure time. The problem of this algorithm is its low resolution in the depth direction and no temporal tracking capability. The latest PTV algorithm adopted three cameras. With a new 2D temporal-tracking algorithm based on regression, and a spatial-matching algorithm based on epipolar constraint, the PTV algorithm was successfully employed to detect the water flow within a small channel. The limitation of the current PTV algorithm is its low detecting rate and small detecting scale.

To overcome these limitations, a hybrid particle tracking velocimetry (HPTV) system was developed in current research. It combined and further developed the existing VPSTV and

PTV algorithms. This system consists of two components: object hybrid and spatial hybrid. The object hybrid integrated the PTV algorithm and a newly developed streak tracking algorithm. This combination largely extended the velocity range the system can detect. The spatial hybrid algorithm further extended the existing PTV algorithm. Through combing the two-way and three-way matching strategies, a much larger detecting volume were achieved.

The HPTV algorithm was validated and estimated with the synthetic images generated by Matlab. The errors related to each aspect of the algorithms, including particle center detecting, camera calibration and 3D reconstruction, were discussed respectively. The simulated results showed that the detecting errors associated with image processing and calibration algorithms were usually smaller than 3 pixels. These errors were inherited and transformed by the reconstruction algorithm and caused the detecting error in physical domain. There is no universal criterion to evaluate the reconstruction error for the whole system because it is location-dependent and case by case. However, theoretical analysis suggested this error is strongly related to the condition number of the reconstruction matrix, which is mainly controlled by the camera configuration. A perpendicular camera configuration will result in the smallest error, but the smallest detecting volume.

11 cases simulating particle movement along helix curves were conducted to validate the performance of the tracking algorithms. Three parameters including particle density, particle velocity, and exposure time were varying in these cases. The results suggested that the algorithm performance was affected by a parameter called “particle density rate”. A larger particle density rate indicates the widely separated particles. It will be easier to track such cases. The results also showed that the hybrid spatial matching algorithm had helped nearly doubled the .detecting rate at high particle density cases. For those cases with given exposure time and high particle velocity, the object hybrid tracking approach had been successfully applied. 19 out of 25 released particles were detected by the system,

Finally, the developed HPTV system was applied in the measurements of airflow pattern within a full scale aircraft cabin in both global and local levels at three different ventilation rates.

A practical measurement procedure was developed and strictly followed throughout the experiments. The measurement errors were analyzed based on particle simulation and statistical analysis. According to the simulation result, the average error in physical domain was 1.988mm for global level case and 0.783mm for local level case. For global level experiment, the error contributed by Z-direction was almost three times larger than those by the other two directions, which strongly indicated the importance of the camera configuration. The 3D trajectories were converted to velocity field and mapped onto uniform grids. Then the contour map, streamlines and particle movement movies were produced and compared with the CFD simulated results. They had shown good agreement.

These applications have demonstrated that the newly developed HPTV system is capable of capturing the instantaneous velocity field in large volumetric domain at relatively high frequency. It also has the potential to track pollutants dispersion within a confined environment after they were released from the source locations. With this system, it is possible to obtain reliable experimental data for numerical model validation. The objectives of this research have been achieved.

8.2 Recommendations

(1) Asynchronization among cameras will affect not only the spatial matching rate, but also the accuracy of the detected result, especially those high velocity particles. Although an algorithm resisting asynchronization has been developed, it still sustains substantial calibration errors. Hardware synchronization is strongly recommended whenever it is possible.

(2) The velocity is obtained through multiplying frame displacement and camera frame rate. For a given 3D reconstruction accuracy, a higher camera speed will produce larger detecting error. Therefore, the high speed CCD camera is not always a good choice for the PTV experiment. It should be determined by balancing the accuracy requirement and target velocity range.

(3) Camera setup will intensively affect the accuracy of the system. A trial-and-error procedure is recommended to optimize the camera configuration in order to achieve a large detectable volume with satisfied accuracy.

(4) For a large scale experiment with high trace particle density, error of a few pixels in the image processing phase could produce large error in the reconstruction process. Special attention should be paid to ensure experimental quality. (a) Selecting high quality image acquisition devices; (b) Using an illuminating system which can highlight the seeded particles while not glaring the background; and (c) Developing highly efficient and accurate image processing algorithm.

(5) The accuracy of the HPTV system is not some constant number. Before each experiment, it is the user's responsibility to evaluate the errors associated with their system configuration, and the acceptable error tolerance.

(6) The speed of the current algorithm is a problem with the existing computational capability. It takes up to two days to process the data of one case. It is expected that massively parallel computing techniques will benefit this algorithm. The on-going research on this topic using parallel computing on a Graphics Processing Unit (GPU) architecture will greatly enhance its speed in several multitudes.

REFERENCES

Aboosaidi, F., M. J. Warfield, and D. Choudhury. 1991. Computational fluid dynamics applications in airplane cabin ventilation system design. *Proceedings - Society of Automotive Engineers*, 249-258. Nagoya, Japan.: SAE.

Adamczyk, A. A., and L. Rimai. 1988a. Reconstruction of a 3-dimensional flow field from orthogonal views of seed track video images. *Experiments in Fluids* 6(6): 380-386.

Adamczyk, A. A., and L. Rimai. 1988b. Two-dimensional particle tracking velocimetry (PTV) technique and image-processing algorithms. *Experiments in Fluids* 6(6): 373-380.

Adrian, R. J. 1991. Particle-imaging techniques for experimental fluid mechanics. *Annual Review of Fluid Mechanics* 23(1): 261-304.

Adrian, R. J. 1984. Scattering particle characteristics and their effect on pulsed laser measurements of fluid flow: Speckle velocimetry vs particle image velocimetry. *Applied Optics* 23(11): 1690-1691.

ANSYS. 2007. FLUENT 6.3 Documents (software package). Canonsburg, PA.: ANSYS, Ltd.

Arroyo, M. P., and C. A. Greated. 1991. Stereoscopic particle image velocimetry. *Measurement Science and Technology* 2(12): 1181-1186.

Barnard, S. T., and M. A. Fischler. 1982. Computational stereo. *Computing Surveys* 14(4): 554-572.

Barnhart, D. H., W. D. Koek, T. Juchem, N. Hampp, J. M. Coupland, and N. A. Halliwell. 2004. Bacteriorhodopsin as a high-resolution, high-capacity buffer for digital holographic measurements. *Measurement Science and Technology* 15(4): 639-646.

Bosbach, J., J. Pennecot, C. Wagner, M. Raffel, T. Lerche, and S. Repp. 2006. Experimental and numerical simulations of turbulent ventilation in aircraft cabins. *Energy* 31(5): 694-705.

Brown, D. C. 1971. Close-range camera calibration. *Photogrammetric Engineering and Remote Sensing* 37(8): 855-866.

Carosone, F., A. Cenedese, and G. Querzoli. 1995. Recognition of partially overlapped particle images using the Kohonen neural network. *Experiments in Fluids* 19(4): 225-232.

Cenedese, A., and G. Querzoli. 1997. Lagrangian statistics and transilient matrix measurements by PTV in a convective boundary layer. *Measurement Science and Technology* 8(12): 1553-1561.

Chen, Q. 1995. Comparison of different K-EPSILON models for indoor air flow computations. *Numerical Heat Transfer, Part B: Fundamentals* 28(3): 353-369.

Cowen, E. A., and S. G. Monismith. 1997. A hybrid digital particle tracking velocimetry technique. *Experiments in Fluids* 22(3): 199-211.

Cummins, H. Z., N. Knable, and Y. Yeh. 1964. Observation of diffusion broadening of rayleigh scattered light. *Physical Review Letters* 12(6): 150-153.

Doh, D. H., T. G. Hwang, and T. Saga. 2004. 3D-PTV measurements of the wake of a sphere. *Measurement Science and Technology* 15(6): 1059-1066.

Doh, D. H., D. H. Kim, S. H. Choi, S. D. Hong, T. Saga, and T. Kobayashi. 2000. Single-frame (two-field image) 3-D PTV for high speed flows. *Experiments In Fluids* 29(supplement): 85-98.

Feng, Y., J. Goree, and B. Liu. 2007. Accurate particle position measurement from images. *Review of Scientific Instruments* 78(5): Article No. 053704.

Flynn, M. R., K. Ahn, and C. T. Miller. 1995. Three-dimensional finite-element simulation of a turbulent push-pull ventilation system. *Annals of Occupational Hygiene* 39(5): 573-589.

Gao, N. P., and J. L. Niu. 2004. CFD study on micro-environment around human body and personalized ventilation. *Building and Environment* 39(7): 795-805.

Gao, N. P., and J. L. Niu. 2008. Personal ventilation for commercial aircraft cabins. *Journal of aircraft* 45(2): 508-512.

Gebremedhin, K. G., and B. Wu. 2003. Characterization of flow field in a ventilated space and simulation of heat exchange between cows and their environment. *Journal of Thermal Biology* 28(4): 301-319.

Goldstein, R. J. 1996. *Fluid Mechanics Measurements*. 2nd ed. Washington, DC.: Taylor and Francis.

Grant, I., X. Pan, F. Romano, and X. Wang. 1998. Neural-network method applied to the stereo image correspondence problem in three-component particle image velocimetry. *Applied Optics* 37(17): 3656-3663.

Guezennec, Y. G., R. S. Brodkey, N. Trigui, and J. C. Kent. 1994. Algorithms for fully automated three-dimensional particle tracking velocimetry. *Experiments in Fluids* 17(4): 209-219.

Hassan, Y. A., T. K. Blanchat, C. H. Seeley Jr, and R. E. Canaan. 1992. Simultaneous velocity measurements of both components of a two-phase flow using particle image velocimetry. *International Journal of Multiphase Flow* 18(3): 371-395.

Hayashi, T., Y. Ishizu, S. Kato, and S. Murakami. 2002. CFD analysis on characteristics of contaminated indoor air ventilation and its application in the evaluation of the effects of contaminant inhalation by a human occupant. *Building and Environment* 37(3): 219-230.

He, G., X. Yang, and J. Srebric. 2005. Removal of contaminants released from room surfaces by displacement and mixing ventilation: modeling and validation. *Indoor Air* 15(5): 367-380.

Heikkila, J., and O. Silven. 1997. Four-step camera calibration procedure with implicit image correction. *Proceedings of the Institute of Electrical and Electronics Engineers Computer Society Conference on Computer Vision and Pattern Recognition*, 1106-1112. San Juan , Puerto Rico.: IEEE.

Hering, F., C. Leue, D. Wierzimok, and B. Jahne. 1997. Particle tracking velocimetry beneath water waves. Part I: visualization and tracking algorithms. *Experiments in Fluids* 23(6): 472-482.

Hocking, M. B. 1998. Indoor air quality: Recommendations relevant to aircraft passenger cabins. *American Industrial Hygiene Association Journal* 59(7): 446-454.

Ishikawa, M. A., Y. Murai, and F. Yamamoto. 2000. Numerical validation of velocity gradient tensor particle tracking velocimetry for highly deformed flow fields. *Measurement Science and Technology* 11(6): 677-684.

Jacobs, P., and W. F. De Gids. 2006. Individual and collective climate control in aircraft cabins. *International Journal of Vehicle Design* 42(1-2): 57-66.

Jiang, Y., D. Alexander, H. Jenkins, R. Arthur, and Q. Chen. 2003. Natural ventilation in buildings: Measurement in a wind tunnel and numerical simulation with large-eddy simulation. *Journal of Wind Engineering and Industrial Aerodynamics* 91(3): 331-353.

Jones, J. D. C., D. J. Anderson, and C. A. Greated. 1997. Fibre-optic beam delivery systems for particle image velocimetry. *Optics and Lasers in Engineering* 27(6): 657-674.

Keane, R. D., and R. J. Adrian. 1991a. Cross-correlation analysis of particle image fields for velocity measurement. *American Society of Mechanical Engineers, Fluids Engineering Division*, 1-8. Atlanta, GA: ASME.

Keane, R. D., and R. J. Adrian. 1991b. Optimization of particle image velocimeters. II. Multiple pulsed systems. *Measurement Science and Technology* 2(10): 963-974.

Keane, R. D., and R. J. Adrian. 1993. Theory and simulation of particle image velocimetry. *Proceedings of SPIE - The International Society for Optical Engineering*, 477-492. Bellingham, WA: ISOE.

Kessler, M., and D. Leith. 1991. Flow measurement and efficiency modeling of cyclones for particle collection. *Aerosol Science and Technology* 15(1): 8-18.

Khalighi, B., and Y. H. Lee. 1989. Particle tracking velocimetry - an automatic image-processing algorithm. *Applied Optics* 28(20): 4328-4332.

Kim, Y., K. Lee, and H. Cho. 2001. Experimental study of flow characteristics of a diffuser for under floor air-conditioning system. *ASHRAE Transactions* 107(1): 230-236.

Klontz, K. C., N. A. Hynes, R. A. Gunn, M. H. Wilder, M. W. Harmon, and A. P. Kendal. 1989. An outbreak of influenza a/taiwan/1/86 (H1N1) infections at a naval base and its association with airplane travel. *American Journal of Epidemiology* 129(2): 341-348.

Knowles, M. W. E., and F. S. Athelstan 1953. *Meteorological Instruments*. 3rd ed. Toronto, Ontario: University of Toronto Press.

Kreith, F., and Y. Goswami 2005. *The CRC Handbook of Mechanical Engineering*. 2nd ed. Boca Raton, FL: CRC Press.

Leder, K., and D. Newman. 2005. Respiratory infections during air travel. *Internal Medicine Journal* 35(1): 50-55.

Li, D. 2008. Numerical and experimental study of volumetric particle tracking velocimetry in a spatial and temporal domain. Ph.D. diss. Urbana, IL: University of Illinois at Urbana-Champaign, Department of Agricultural and Biological Engineering.

Li, D. N., Y. H. Zhang, Y. G. Sun, and W. Yan. 2008. A multi-frame particle tracking algorithm robust against input noise. *Measurement Science & Technology* 19(10).

Lin, C. H., R. H. Horstman, M. F. Ahlers, L. M. Sedgwick, K. H. Dunn, J. L. Topmiller, J. S. Bennett, and S. Wirogo. 2005. Numerical simulation of airflow and airborne pathogen transport in aircraft cabins - Part I: Numerical simulation of the flow field. *ASHRAE Transactions* 111(1): 755-763.

Ma, B. S., V. Ruwet, P. Corieri, R. Theunissen, M. Riethmuller, and C. Darquenne. 2009. CFD simulation and experimental validation of fluid flow and particle transport in a model of alveolated airways. *Journal of Aerosol Science* 40(5): 403-414.

Malik, N. A., T. Dracos, and D. A. Papantoniou. 1993. Particle tracking velocimetry in three-dimensional flows - Part II: Particle tracking. *Experiments in Fluids* 15(4-5): 279-294.

Mangili, A., and M. A. Gendreau. 2005. Transmission of infectious diseases during commercial air travel. *Lancet* 365(9463): 989-996.

Medioni, G., and R. Nevatia. 1985. Segment based stereo matching. *Computer Vision, Graphics, and Image Processing* 18: 31-32.

Melling, A. 1997. Tracer particles and seeding for particle image velocimetry. *Measurement Science & Technology* 8(12): 1406-1416.

Menon, R. K. 1998. Section 29.3: Laser anemometry. In *Measurements, Instrumentation and Sensors Handbook*, J. G. Webster, ed. Boca Raton, FL: CRC Press.

Meyer, J. F., and H. Komine. 1991. Doppler global velocimetry - a new way to look at velocity. *ASME 4th International Conference on Laser Anemometry*, 273-277. Cleveland, Ohio: ASME.

Mo, H., M. H. Hosni, and B. W. Jones. 2003. Application of particle image velocimetry for the measurement of the airflow characteristics in an aircraft cabin. *ASHRAE Transactions* 109(2): 101-110.

Mortensen, L. H., C. Rode, and R. Peuhkuri. 2008. Investigation of airflow patterns in a microclimate by particle image velocimetry (PIV). *Building and Environment* 43(11): 1929-1938.

Moser, M. R., T. R. Bender, H. S. Margolis, G. R. Noble, A. P. Kendal, and D. G. Ritter. 1979. An outbreak of influenza aboard a commercial airliner. *American Journal of Epidemiology* 110(1): 1-6.

Muller, R. H. G., T. Scherer, T. Rotger, O. Schaumann, and M. Markwart. 1997. Large body aircraft cabin A/C flow measurement by helium bubble tracking. *Journal of Flow Visualization and Image Processing* 4(3): 295-306.

Murakami, S. 2004. Analysis and design of micro-climate around the human body with respiration by CFD. *Indoor Air* 14: 144-156.

Nishino, K., H. Kato, and K. Torii. 2000. Stereo imaging for simultaneous measurement of size and velocity of particles in dispersed two-phase flow. *Measurement Science & Technology* 11(6): 633-645.

Olsen, S. J., H. Chang, T. Y. Cheung, A. F. Tang, T. L. Fisk, S. P. Ooi, H. Kuo, D. D. Jiang, K. Chen, J. Lando, K. Hsu, T. Chen, and S. F. Dowell. 2003. Transmission of the severe acute respiratory syndrome on aircraft. *New England Journal of Medicine* 349(25): 2416-2422.

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

Owe, J. O. 1997. Air transportation. In *Major Industries and Occupations*, D. Brune, G. Gerhardsson, G. W. Crockford & D. Norback, ed., Geneva, Switzerland: International Labor Office.

Palero, V., N. Andres, M. P. Arroyo, and M. Quintanilla. 2000. Holographic interferometry versus stereoscopic PIV for measuring out-of-plane velocity fields in confined flows. *Measurement Science and Technology* 11(6): 655-666.

Papadopoulos, K. H., A. T. Soilemes, C. G. Helmis, D. N. Asimakopoulos, M. Santamouris, A. Argiriou, and E. Dascalaki. 1996. A triple hot-wire system for indoor air flow measurements. *Journal of Solar Energy Engineering* 118(3): 168-176.

Pereira, F., H. Stuer, E. C. Graft, and M. Gharib. 2006. Two-frame 3D particle tracking. *Measurement Science and Technology* 17(7): 1680-1692.

Pickering, C. J. D., and N. A. Halliwell. 1985. Particle image velocimetry: Fringe visibility and pedestal removal. *Applied Optics* 24: 2474-2476.

Posner, J. D., C. R. Buchanan, and D. Dunn-Rankin. 2003. Measurement and prediction of indoor air flow in a model room. *Energy and Buildings* 35(5): 515-526.

Querzoli, G. 1996. A Lagrangian study of particle dispersion in the unstable boundary layer. *Atmospheric Environment* 30(16): 2821-2829.

Richmond, B. J., A. D. Eisner, L. A. Brixey, and R. W. Wiener. 2006. Transport of airborne particles within a room. *Indoor Air* 16(1): 48-55.

Ruhnau, P., C. Guetter, T. Putze, and C. Schnorr. 2005. A variational approach for particle tracking velocimetry. *Measurement Science and Technology* 16(7): 1449-1458.

Rydock, J. P., and O. Hermansen. 2002. A tracer method for evaluating recirculation of pollutant releases in buildings. *American Industrial Hygiene Association Journal* 63(2): 234-238.

Sandberg, M. 2007. Whole-field measuring methods in ventilated rooms. *HVAC and R Research* 13(6): 951-970.

Singh, A., M. H. Hosni, R. H. Horstman, J. Van Gilder, and R. May. 2002. Numerical simulation of airflow in an aircraft cabin section. *ASHRAE Transactions* 108(1): 1005-1013.

Sinha, S. K. 1988. Improving the accuracy and resolution of particle image or laser speckle velocimetry. *Experiments in Fluids* 6(1): 67-68.

Smith, M. 1998. Application of a planar doppler velocimetry system to a high reynolds number compressible jet. *AIAA Paper 98-0428*, Louis, MO: AIAA.

Sorensen, D. N., and P. V. Nielsen. 2003. Quality control of computational fluid dynamics in indoor environments. *Indoor Air* 13(1): 2-17.

Sun, H., L. Zhao, and Y. Zhang. 2007. Evaluation of RNG kappa-epsilon and LES non-isothermal models for indoor airflow using PIV measurement data. *Transactions of the ASABE* 50(2): 621-631.

Sun, Y. 2007. Volumetric particle streak-tracking velocimetry and its application in indoor airflow measurements. Ph.D. diss. Urbana, IL: University of Illinois at Urban-Champaign, Department of Agricultural and Biological Engineering

Sun, Y., and Y. Zhang. 2003. Development of a stereoscopic particle image velocimetry system for full-scale room airflow studies, Part II: Experimental setup. *ASHRAE Transactions* 109(2): 540-548.

Sun, Y., Y. Zhang, A. Wang, J. L. Topmiller, and J. S. Bennet. 2005. Experimental characterization of airflows in aircraft cabins, Part I: Experimental system and measurement procedure. *ASHRAE Transactions* 111(2): 45-52.

Sun, Y., Y. Zhang, L. Zhao, and X. Wang. 2004. An algorithm of stereoscopic particle image velocimetry for full-scale room airflow studies. *ASHRAE Transactions* 110(1): 75-80.

Suzuki, Y., and N. Kasagi. 2000. Turbulent air-flow measurement with the aid of 3-D particle tracking velocimetry in a curved square bend. *Flow Turbulence and Combustion* 63(1-4): 415-442.

Tavoularis, S. 2005. *Measurement in Fluid Mechanics*. Cambridge, England: Cambridge University Press.

Thomas, S. H., B. Robert, B. Homer, D. C. Barry, D. Colin, and L. Len. 1999. Assessment of technologies deployed to improve aviation security: first report. Washington, DC: National Materials Advisory Board.

Topp, C., P. V. Nielsen, and D. N. Sorensen. 2002. Application of computer simulated persons in indoor environmental modeling. *ASHRAE Transactions* 108(2): 1084-1089.

Trucco, and A. Verri 1998. *In Introductory Techniques for 3-D Computer Vision*. Upper Saddle River, NJ: Prentice Hall PTR.

Wang, A. 2006. Quantifying air distribution, ventilation effectiveness and airborne pollutant transport in an aircraft cabin mock-up. Ph.D. diss. Urbana, IL: University of Illinois at Urbana-Champaign, Department of Agricultural and Biological Engineering.

Yakhot, V., and S. A. Orszag. 1986. Renormalization group analysis of turbulence. *Physical Review Letters* 57(14): 1722-1724.

Yan, W., X. Yang, and M. Shan. 2009. How to simplify computer simulated persons (CSPs) for modeling personal microenvironments: comparison and case studies. *ASHRAE Transactions* 115(1): 473-483.

Yan, W., Y. Zhang, Y. Sun, and D. Li. 2009. Experimental and CFD study of unsteady airborne pollutant transport within an aircraft cabin mock-up. *Building and Environment* 44(1): 34-43.

Zhang, T., and Q. Y. Chen. 2007. Novel air distribution systems for commercial aircraft cabins. *Building and Environment* 42(4): 1675-1684.

Zhang, Y., Y. Sun, A. Wang, J. L. Topmiller, and J. S. Bennett. 2005. Experimental characterization of airflows in aircraft cabins, part II: Results and research recommendations. *ASHRAE Transactions* 111(2): 53-59.

Zhang, Z. 1999. Flexible camera calibration by viewing a plane from unknown orientations. *Proceedings of the Institute of Electrical and Electronics Engineers International Conference on Computer Vision*, 666-673, Kerkyra, Greece: IEEE.

Zhang, Z., X. Chen, S. Mazumdar, T. Zhang, and Q. Chen. 2009. Experimental and numerical investigation of airflow and contaminant transport in an airliner cabin mockup. *Building and Environment* 44(1): 85-94.

Zhao, L. 2000. Measurement and analysis of full-scale room air-flow using particle imagine velocimetry (PIV) techniques. Ph.D. diss. Urbana, IL: University of Illinois at Urbana-Champaign, Department of Agricultural and Biological Engineering.

Zhao, L., Y. Zhang, X. Wang, G. L. Riskowski, and L. L. Christianson. 1999. Measurement of effect of air exchange rate on airflow patterns in ventilated spaces using particle image velocimetry. *ASAE/CSAE Annual International Meeting*, Paper No.994156. Toronto, ON: ASAE/CSAE.

Zhou, X., Q. Ouyang, G. Lin, and Y. Zhu. 2006. Impact of dynamic airflow on human thermal response. *Indoor Air* 16(5): 348-355.

Zhu, S. W., S. Kato, S. Murakami, and T. Hayashi. 2005. Study on inhalation region by means of CFD analysis and experiment. *Building and Environment* 40(10): 1329-1336.

Zhu, S. W., S. Kato, and J. H. Yang. 2006. Study on transport characteristics of saliva droplets produced by coughing in a calm indoor environment. *Building and Environment* 41(12): 1691-1702.

APPENDIX A. PROCEDURES OF CAMERA CALIBRATION

The camera calibration toolbox used in current project is developed by the vision group in Caltech. It is used to extract both the intrinsic and extrinsic camera parameters of the HPTV system. A briefly calibration procedure is described as following:

(1) Add the location of calibration toolbox to the main Matlab path. The default folder name is **TOOLBOX_calib**.

(2) Run the main function **calib_gui**. Select the **Standard** running mode.

Figure A-1. The mode selection window.

(3) Assume that the calibration images are saved in four folders: (a) **cam1**; (b) **cam2**; (c) **cam3**; (d) **common**. The first three folders saved the recorded calibration images captured by three cameras respectively, and the last folder saved the common view pictures synchronously recorded by three cameras.

(4) Click **image names** and set the base name of the images. For example if the image name is "im1.bmp", input "im" as the base name. In our case, we don't have base name, therefore just press **enter** to continue. Then indicate the image format. Enter "**b**" as BMP format. All images will be shown in thumbnail format.

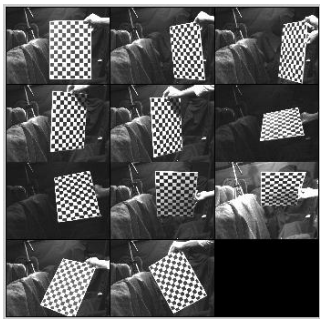


Figure A-2. All images shown in thumbnail format.

(5) Click the "**Extract grid corners**" button in the toolbox window. Press "**enter**" to select all images. Then use default values for the following questions. This leads to a corner extraction engine for counting the number of squares in the grid. The user should manually enter the four corners of the board, as shown in figure A.3.

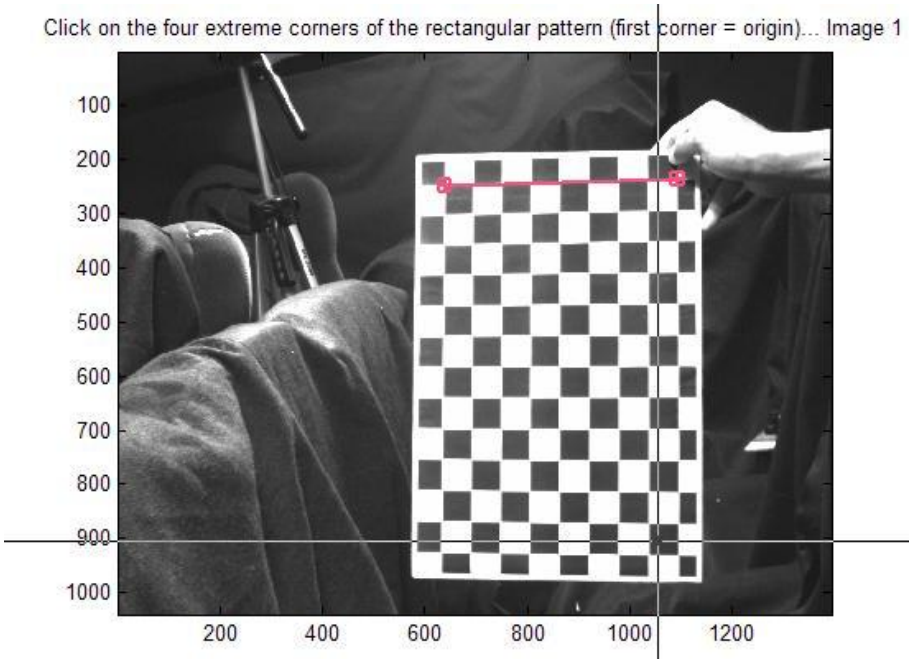


Figure A-3. Corners selecting window.

Click on the four extreme corners on the rectangular checkerboard pattern. The first clicked point is selected to be associated to the origin point of the reference frame attached to the grid. The other three points of the rectangular grid can be clicked in any order. Then enter the grid size of the squares. In current example, both dX and dY are equal to 30mm. Keep the other questions as default value.

Repeat this process for all calibration images.

(6) After the corner extraction, click on the "**Calibration**" button of the toolbox to run the main calibration procedure.

(7) Click the "**Recomp. corners**" button to re-compute the corners on all the images; and then click the "**Calibration**" button again to run the calibration procedure again. Keep all the default values during the whole procedure.

(8) Click on "**Comp.Extrinsic**" buttons on the toolbox. Copy the common view picture of camera one from common directory to current directory. Enter the full image name without extension. In current example, the image name is "cam1". Then enter the image format as "**b**' which indicates BMP format. Keep all other value as default numbers. A window similar to Figure A.3 will appear. The user needs to indicate the four corners of the grid board.

(9) Click on the "**File**" menu and select "**save work space as..**" command to save the calibration data in mat format.

(10) Apply the same procedure for the other two cameras.

APPENDIX B. PROCEDURES OF HYBRID PARTICLE TRACKING

The hybrid particle tracking procedures include three steps: (1) image processing; (2) 2D temporal tracking; (3) 3D spatial matching and streaks tracking. These procedures will be conducted with two toolboxes: ptv_gui v1.0 developed by Dongning and ptv2_gui developed in current project.

Let's use a local environment testing case as the example. The images of this case are saved in three folders: **calib**, **bg**, and **particle**. The **calib** folder recorded all the camera calibration images which can be processed according to the procedures in Appendix A. The **bg** folder recorded the background images taken by three cameras: A0_1~10, A1_1~10, and B0_1~10. The **particle** folder recorded the particle images taken by three cameras: A0_1~30, A1_1~ 30 and B0_1~30.

(1) Run the main function **ptv2_gui**. The main interface window is shown as following:

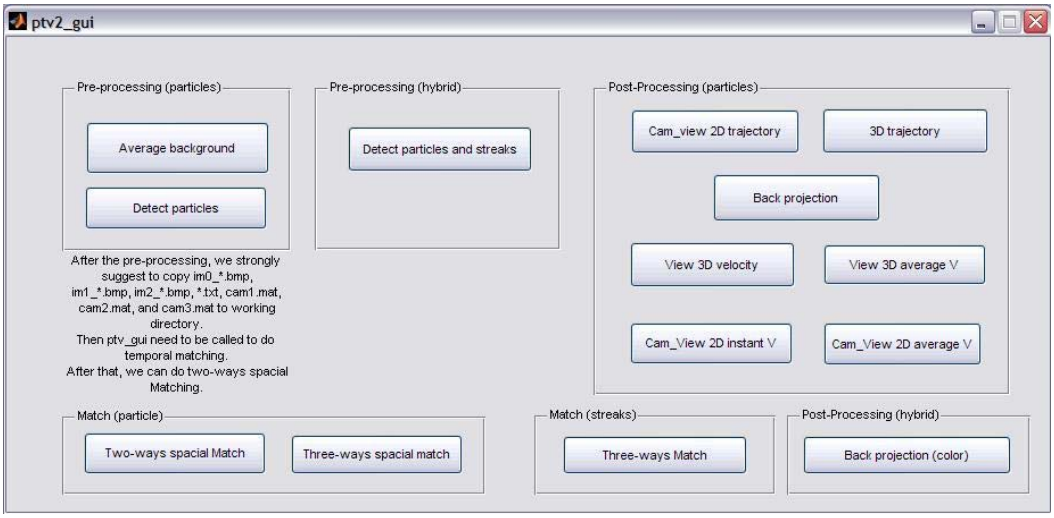


Figure B-1. The main GUI of ptv2_gui toolbox.

Image processing:

(2) Click "**average background**" button. Select the first image **A0_1.bmp** in the **bg** folder. Enter the total number of images taken by each camera (enter **10** in this case). The algorithm

will automatically generate the averaged background images named as A0_ave.bmp, A1_ave.bmp and B0_ave.bmp.

(3) Copy **A0_ave.bmp**, **A1_ave.bmp** and **B0_ave.bmp** to the **particle** folder.

(4) If the user only wanted to process dot shape particle tracking, click the "**Detect particles"** button. In current case, we want to track both dot and streak shape particles. Click "**Detect particles and streaks**" button. Select the first particle image "**A0_1.bmp**" in the **particle** folder. Enter the total number of image per camera (**30** for current case). Enter **0.6** as the bin level threshold. Three sorts of files will be generated: dot particle files, streak particle files and B/W particle images. They are named as im*_*.bmp, im*_*_streaks.txt, and im*_*.bmp respectively. The 2D particle trajectories images obtained by combining all thirty frames of each camera are also produced named as im0_ave.bmp, im1_ave.bmp and im2_ave.bmp.

2D temporal tracking:

(5) Run the main function **ptv_gui**. The main interface window is shown as following:

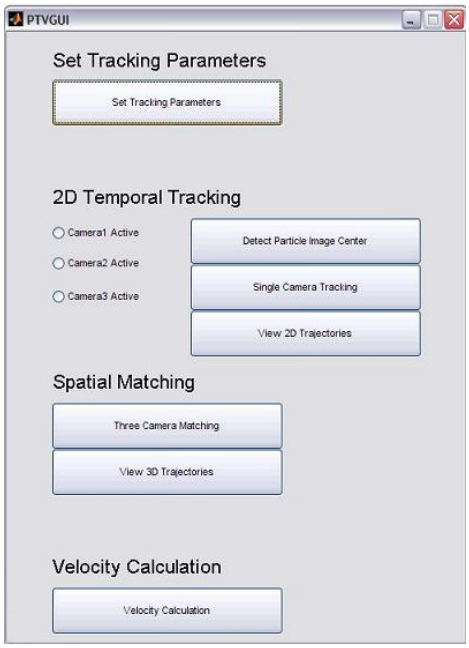


Figure B-2. The main GUI of ptv_gui toolbox.

(6) Click the active button of three cameras "**Camera1 Active**" "**Camera2 Active**" and "**Camera3 Active**". Then click "**Single Camera Tracking**". It may take hours or days to complete this 2D tracking process. Three data files will be generated at the end of this step: im0_linked_raw.txt, im1_linked_raw.txt, and im2_linked_raw.txt.

(7) Go back to ptv2_gui windows. Click the "**Cam_view 2D trajectory**" to visualize the 2D trajectories.

3D spatial matching and streaks tracking:

(8) Click the "**Two-ways spatial Match**" button to perform the 3D reconstruction for the dot shape particles. The user should indicate the calibration data files for three cameras. Then enter the threshold for spatial matching (**1.2** for current project). It may take up to one hour to complete the whole process. The output of this process is a 3D trajectories file names as "Traj_3D.txt"

(9) Then we need to do 3D streaks tracking. Click the "**Three-ways Match**" button in "Match(streaks)" frame. Similarly the location of the calibration files should be indicated. Then enter the threshold for matching. The 3D streaks will be constructed frame by frame and saved as streak*_3D_3ways.txt.

Post-processing:

Two kinds of post-processing commands have been developed: (1) commands for pure dot-shape particles; and (2) commands for hybrid particles.

The commands for pure dot-shape particles include 3D trajectories visualization ("3D trajectory"), 3D trajectories back projection ("Back projection"), instance velocity field visualization ("View 3D velocity"), average velocity field visualization ("View 3D average V"), instant velocity field back projection ("Cam_view 2D instant V"), and average velocity field back projection ("Cam_View 2D average V").

At current stage, back projection is the only post-processing command developed for hybrid particle tracking.

(10) Click the "**Back projection (color)**" button to process the back projection. Then select the calibration files as required. The user also needs to select the background images for the projection.

(11) Enter the "**view_3D_average_jet**" command manually will produce a data file named as "tec_streaks.txt" which can be read by Tecplot. Further visualization could be processed with this third party software.

APPENDIX C. VISUALIZATION OF THE 3D PARTICLES WITH THE TOOLBOX BASED ON VTK LIBRARY

The Visualization Toolkit (VTK) is an open-source, freely available software system for 3D computer graphics, image processing and visualization. VTK consists of a C++ class library and several interpreted interface layers including Tcl/Tk, Java, and Python. A small 3D particle visualization toolbox has been developed based on TCL language. This chapter will introduce how to use it.

(1) Open http://www.vtk.org/VTK/resources/software.html. Download the VTK source code and compile it on your own computer according to its instruction.

(2) Copy the **c.tcl** to the **bin** directory.

(3) Copy the data you want to visualize to the same folder. These data include: "Traj_3D.txt", "streak*_3D_3ways.txt", and "SetGetTime.txt"

(4) Enter "source c.tcl".

(5) The main interface shows as Figure C.1

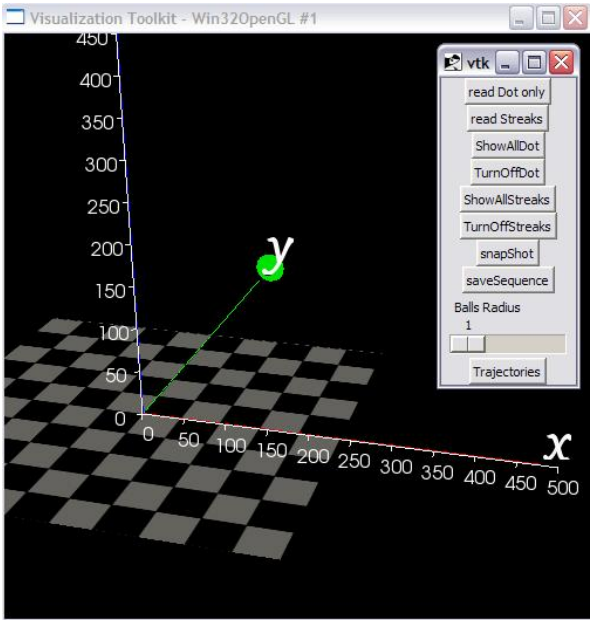


Figure C-1. The interface window of the visualization toolbox.

(6) Click the "**read Dot only**" button on the panel. The software will read the "Traj_3D.txt" file and visualize the particles at the first frame.

(7) The size of the particle could be controlled by dragging the "**Balls radius**" button.

(8) Click the "**read Streaks**" button. The software will read all of the streaks files "streak*_3D_3ways.txt"

(9) The objects could be rotated by pressing **left mouse button** on the image.

(10) Press the **right mouse button** on the image will zoom in/out the objects.

(11) Press the **middle mouse button** and drag on the image will pan move the objects.

(12) Click the "**ShowAllDot**" button will display dot-shape particles at all frames.

(13) Click the "**ShowAllStreaks**" button will display streaks at all frames.

(14) "**TurnOffDot**" and "**TurnOffStreaks**" will erase all dot-shape particles and streaks from the image.

(15) The current image can be saved by clicking "**snapShot**" button.

(16) The "**SaveSequence**" command will save all frames one by one to the default directory.

(17) The "**Trajectories**" command can generate smooth streamlines according to the 3D particles recorded in "Traj_3D.txt" file.

APPENDIX D. ANALYSIS TOOLS

Series of tools have been developed to estimate the performance of the HPTV algorithm. There is not interface build for them. They can only be called through command line.

Artificial image simulation tools:

Image simulation tools are developed to produce particle images for algorithm analysis. All these commands will need the camera calibration data. Assume that these files have been copied to current directory.

(1) "**simhelix**" function: This function produces images simulating particles moving along helix curves. Firstly, the user should input the total number of released particles. Then input the size of the images. The default value is [1040,1392]. It also requires the helix diameter, the movement distance per revolution, and the total number of revolutions to simulate. Notified that 24 frames will be generated in each revolution.

(2) "**simhelix_streak_exp60**" function. This function produces pure streak-shape images for particles moving along helix curves. The exposure time set in this function is 1/60s. The input values are identical to "simhelix" function.

(3) "**simhelix_dot_streak_exp60**" function. This function produce images with both dot-shape and streaks particles on them. The default frame rate is 30fps. The exposure time is 1/60s. The particle velocity is uniformly increased in this simulation. The input values are identical to "simhelix" function.

Common view analysis tools:

For given camera setup, it is important to know the detectable volume under its configuration. The common view analysis tools are developed to fulfill such requirement.

(1) "**com_view**" function. This function will read the camera calibration data saved in current directory. Using these camera parameters, it will construct the common view of all three cameras and those between each two cameras. The user should input the real-world

dimension. The default value is [-1000 1000 -1000 1000 -1000 1000]. The size of the image is also need to be specified. The default value is [1092, 1040]. The resultant data is a file named as “cam123_view.dat”. It can be read and further visualized by TecPlot.

Error analysis tools:

As described in section 5.3.2.3, the error analysis algorithm based on statistical analysis has been proposed. The function is developed following that procedures.

(1) “**errorsimulate**” function. After running errorsimulate, the user should input the dimension of the target volume. The default value is [-500 500 -500 500 -500 500]. Then input the total number of simulated particles. The default value is 2000. Finally define the size of image. In current case, we choose [1092, 1040]. The function will automatically call the reconstruction algorithm and compute the detected particle locations. These locations are compared with the exact value to evaluate the tracking errors. Error distributions will be plotted similar to Figure 5-14~17. In the same time, the common view analysis data is also generated and saved as “cam123_view.dat”.

AUTHOR'S BIOGRAPHY

Wei Yan was born in Wendeng County, Shandong Province of China. He finished his high school education with top ranking and showed strong interest in science and engineering. He won first prize on the Shandong physics competition and the excellent prize on the Shandong mathematics competition. In September 1994, Wei entered the Department of Thermal Engineering at Tongji University, Shanghai, China. Due to his excellent academic performance, he was awarded the people's Changqing Scholarship and the Siemens Fellowship during his undergraduate study. In 1998, he was admitted to graduate school in the same department with the national entry exam waived. After receiving his M.S. degree in 2001, Wei worked as a Mechanical Engineer in China International IPPR Engineering Corporation in Beijing for nearly three years. After that, he moved back to Shanghai and worked for Samsung Engineering. Wei got married in Shanghai at that time. In January 2005, Wei started his Ph.D. study at the Department of Civil, Architectural and Environmental Engineering, University of Miami, Coral Gables, Florida. In the following year, he transferred to the Department of Agricultural and Biological Engineering at the University of Illinois at Urbana-Champaign to continue his Ph.D. study. His research work was focused on fluid visualization, computational fluid dynamics and indoor air quality. Wei was awarded the Achievement Fellowship in 2008 for his excellent academic performance. He has published several papers in peer-reviewed journals and served as the peer-reviewer for ASHRAE Transactions and Transactions of the ASABE.