

Copyright Warning & Restrictions

The copyright law of the United States (Title 17, United States Code) governs the making of photocopies or other reproductions of copyrighted material.

Under certain conditions specified in the law, libraries and archives are authorized to furnish a photocopy or other reproduction. One of these specified conditions is that the photocopy or reproduction is not to be “used for any purpose other than private study, scholarship, or research.” If a user makes a request for, or later uses, a photocopy or reproduction for purposes in excess of “fair use” that user may be liable for copyright infringement,

This institution reserves the right to refuse to accept a copying order if, in its judgment, fulfillment of the order would involve violation of copyright law.

Please Note: The author retains the copyright while the New Jersey Institute of Technology reserves the right to distribute this thesis or dissertation

Printing note: If you do not wish to print this page, then select “Pages from: first page # to: last page #” on the print dialog screen

The Van Houten library has removed some of the personal information and all signatures from the approval page and biographical sketches of theses and dissertations in order to protect the identity of NJIT graduates and faculty.

ABSTRACT

Evaluation Of Collision Properties Of Spheres Using High-Speed Video Analysis

**by
Jian Yu**

Experimental evaluation of the collision properties of spheres is performed using video image analysis techniques. A high-speed Kodak EktaPro1000 video camera is utilized to record a collision sequence between two spheres at 1000 frames/sec, and then the images are analyzed to calculate three dimensional translation and rotation before and after the collision. These quantities are used to compute the collision properties for a pair of one inch nylon spheres, i.e. the coefficient of friction, and the coefficients of normal and tangential restitution. The focus of the thesis is on image analysis techniques that provide high accuracy results even though the image resolution is very low, i.e. 240x192 pixels. The procedure developed here can be extended to smaller size spheres and can also be applicable to other motion analysis experiments involving low resolution images.

**EVALUATION OF COLLISION PROPERTIES
OF SPHERES USING HIGH-SPEED VIDEO ANALYSIS**

by
Jian Yu

**A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Mechanical Engineering**

**Department of Mechanical and Industrial Engineering
October 1993**

APPROVAL PAGE

Evaluation Of Collision Properties Of Spheres Using High-Speed Video Analysis

by
Jian Yu

Dr. Rajesh N. Dave, Thesis Advisor 8/13/93
Associate Professor of Mechanical Engineering, NJIT date

Dr. Anthony Rosato, Committee Member 8/13/93
Assistant Professor of Mechanical Engineering, NJIT date

Dr. Ian S. Fischer, Committee Member 8/13/93
Associate Professor of Mechanical Engineering, NJIT date

BIOGRAPHICAL SKETCH

Author: Jian Yu

Degree: Master of Science in Mechanical Engineering

Date: October 1993

Date of Birth:

Place of Birth:

Undergraduate and Graduate Education:

- Master of Science in Mechanical Engineering,
New Jersey Institute of Technology, Newark, NJ, 1993
- Master of Engineering in Engineering Mechanics,
Tsinghua University, Beijing, P. R. of China, 1986
- Bachelor of Science in Engineering Mechanics,
Zhejiang University, Hangzhou, P. R. of China, 1982

Major: Mechanical Engineering

Presentations:

Yu, Jian, "Evaluation of Collision Parameters of Spheres through Use of Computer Vision." Presented at *ASME Region-II Graduate Student Technical Conference*, Polytechnic University, April, 1993.

Dave, Rajesh, Yu, Jian, and Rosato, Anthony "Measurement of Collisional Properties of Spheres Using High-Speed Video Analysis." submitted to *1993 ASME Winter Annual Meeting*, New Orleans, Louisiana, November 28 - December 3, 1993.

*This thesis is dedicated
to my wife and my parents*

ACKNOWLEDGMENT

I take this opportunity to express my deep gratitude to Dr. Rajesh N. Dave, Associate Professor, Mechanical Engineering Department of NJIT. for his encouragement and valuable guidance throughout the course of this work. His helpful hints, suggestions and patience were of immense help.

I am also very thankful to Dr. Anthony Rosato and Dr. Ian Fischer for reviewing my work and providing me with their useful suggestions.

I also thank my colleagues Kurra Bhaswan, John Caesar, Summit Sen, Yong Wang, Jerry Volcy, Avadhani Ashok and Tony La Rosa who have lent a helping hand at various times during the course of this thesis work.

TABLE OF CONTENTS

Chapter	Page
1 INTRODUCTION	1
1.1 Background	1
1.2 Overview of the Experiment	1
1.3 Statement of the Problem	2
1.4 Overview of the Remaining Chapters.....	3
2 LITERATURE REVIEW	4
2.1 Granular Flow Model	4
2.2 Image Processing	5
2.2.1 Edge Detection.....	5
2.2.2 Ellipse and Circle Detection	7
3 EXPERIMENT SETUP	9
3.1 Collision Mechanics	9
3.2 Experimental Setup	11
4 IMAGE PROCESSING	16
4.1 Laplacian of Gaussian Filter.....	16
4.2 Subpixel Zero-crossing	16
4.3 A Sequential Labelling	20
4.4 AFCS Algorithm	21
5 DETERMINATION OF TRANSLATIONAL AND ANGULAR VELOCITY VECTORS	24
5.1 Translational Velocity	24
5.2 Angular Velocity	26
5.2.1 Determination of the Angle and Axis	27
5.2.2 Determination of Rotation Matrix	30
5.2.3 Determination of the Angular Velocity	32
5.3 Collision Position	33
6 THE ACCURACY ANALYSIS OF THE EXPERIMENT	35
7 DETERMINATION OF THE COLLISION PARAMETERS AND RESULTS	39

Chapter	Page
7.1 Determination of the Collision Parameters	39
7.2 Results	41
8 CONCLUSION	45
APPENDIX A Specifications of the Kodak System	47
APPENDIX B Proximal Uniformity Function	50
APPENDIX C Procedure of Evaluating Collision Parameters	53
REFERENCES	54

LIST OF TABLES

Table		Page
3.1	The Coordinates of the Collision Sphere 1	15
3.2	The Coordinates of the Collision Sphere 2	15
3.3	Relative Normal and Surface Velocities	15
6.1	The Accuracy of Translational Test	36
6.2	The Accuracy of Rotational Test	37
7.1	The Properties of Collision Sphere	44

LIST OF FIGURES

Figure	Page
3.1a Curve for Rotation Restitution Coefficient	10
3.1b Rolling and Sliding Solution	10
3.1c Curve for Coefficient of Restitution	10
3.2a The First Experimental Apparatus	11
3.2b The Second Experimental Apparatus	12
3.2c Schematic Diagram of the Second Apparatus	12
3.3 Experimental Schematic Diagram	13
4.1a Comparison of Subpixel vs. Pixel in X Direction Translation	19
4.1b Comparison of Subpixel vs. Pixel in Y Direction Translation	19
4.1c Comparison of Subpixel vs. Pixel in X-Y Direction Translation	20
4.2a Comparison of AHT vs. AFCS in X Direction Translation	22
4.2b Comparison of AHT vs. AFCS in Y Direction Translation	22
4.2c Comparison of AHT vs. AFCS in X-Y Direction Translation	23
5.1 Schematic Diagram of Trajectory of the Sphere	25
6.1 Results of Translation Test	36
6.2 Results of Rotation Test	38
7.1 A Grey Level Image Before Sphere Collision	41
7.2 A Sequence of Grey Level Images for Collision Spheres	42
7.3 A Sequence of Edge Images for Collision Spheres	43
7.4 Experimental Results for the Collision Parameters for 1” Spheres	44

CHAPTER 1

INTRODUCTION

1.1 Background

In recent years a substantial increase of interest in the granular flow of materials has been seen. It can be applied to many industrial fields, such as in-plant and long-distance transport, manufacturing ceramics, casting of solid-fuel rocket propellant, pharmaceuticals, plastics, materials development, food, mineral processing operations and natural geological flows [1]. Due to a poor knowledge of the collisional behavior properties of granular material, it is very difficult for one to make scaling from laboratory bench-top prototype operations to large-scale commercial plants. It is like a “cut-and-try” art other than a reasonable design process [2]. The highly nonlinear nature of the granular flow poses enormous difficulties in developing constitutive models to predict behavior over a range of conditions. Considerable advances in the theoretical models of kinetics and dynamics about granular flow have been made and particle dynamic simulations can provide a way in which the theoretical models may be tested and validated in the absence of experimental data [3]. However, the results from these simulations strongly rely on the input particle material coefficients; sliding friction coefficient, normal restitution and rotational restitution coefficients. Clearly, problems will be encountered when experimental data is compared with theoretical models, if the material parameters governing particle collisions are uncertain. Therefore, it is necessary to devise a three-dimensional experimental technique to obtain these properties used by numerical simulations, theories and other experiments.

1.2 Overview of the Experiment

In the past, the experiments to measure the material collision parameters were two-dimensional in nature. In this thesis, a three-dimension experiment is performed using Kodak EktaPro 1000 high speed video imaging system. Innovative image processing

techniques are utilized to automatically compute the particle velocities before and after sphere collisions.

The experiment is devised by using two one-inch white nylon spheres colliding space. A Kodak EktaPro 1000 high speed video system is used to record the whole collision procedure. The video system is able to record at a speed up to 1000 frames per second and can be replayed immediately at a low speed. This capability allows one to visually check the quality of the collision image so that a series of proper frames can be chosen for further processing. In order to determine the orientation of the spheres, some random markers are drawn on sphere surfaces. The images recorded can be transferred from the EktaPro 1000 video tape to the data images in a computer. Then, the images are processed individually to find the edges and the centers of spheres and markers. The markers are later matched from frame to frame to track their trajectories. This information is used to determine the collision position and the pre- and post-collision translational and rotational velocity vectors.

The most challenging part of this experiment is to accurately evaluate the position and orientation of spheres with the limited resolution images. The subpixel technique [4] is employed for detecting edges to a high accuracy and the fuzzy c-shell algorithm [5] is utilized to accurately find the centers of spheres for determining the translation, and to find the locations of markers for computing the sphere orientations. This experiment is unique in that it measures the actual angular velocity of spheres and the axis of rotation during their motion using automated image analysis.

1.3 Statement of the Problem

The purpose of this thesis is to evaluate the collision parameters of sphere using the Kodak EktaPro 1000 high speed video imaging system. Subpixel technique is employed to obtain high accuracy edges. The edge information is used to find the sphere center and marker location using the fuzzy clustering method. Correspondence of the markers in a series of frames is achieved by the proximal uniformity constraint [6]. The least squared error

method is utilized to determine the matrix of rotation between frames through use of information of the marker locations. The rotation matrix is employed to obtain the rotation angle and axis at different time steps. The translation displacements are measured through a simple difference of two sphere centers. The results are used to compute the velocity vectors based on the polynomial functions and the location of a collision is obtained using the velocity vectors. Finally, the collision coefficients are calculated by fitting straight lines to all the experimental results, as will be explained further on.

1.4 Overview of the Remaining Chapters

The chapter 2 briefly surveys the previous work in the field of hard sphere inelastic collision models, relevant experiments, and image processing techniques for edge detection, Hough transforms and fuzzy clustering methods.

Chapter 3 describes the experimental setup. Chapter 4 discusses several techniques of image processing used in this thesis. Chapter 5 presents the algorithm to determine the translational and angular velocities of colliding spheres. Chapter 6 discusses the accuracy of the experiment. Chapter 7 presents the results of the work conducted. Chapter 8 concludes the work with a summary of the various techniques used.

Appendix A gives a brief description of the high speed Kodak EktaPro 1000 video system. Appendix B contains an algorithm for feature point correspondences. Appendix C describes the steps and programs used to evaluate collision parameters.

CHAPTER 2

LITERATURE REVIEW

2.1 Granular Flow Model

The granular flow modeling of phenomena involving particulates is relevant to many fields. Recent years have seen a tremendous increase of interest in the flows of granular materials whose rheology is dominated by the physical contact between particles and between particles and the containing walls. Due to its highly nonlinear nature, it is very difficult to develop constitutive models to predict behavior over a range of conditions. Kinetic theory models have taken advantage of the similarities between the shearing of a granular material and the dynamics of dense systems of colliding gas molecules [3]. Dynamic theory models and simulations have been done to test theoretical models. The results from these simulations rely on input parameters, such as the coefficients of friction, and the coefficients of normal and tangential restitution. However, experimental studies aimed at measurements of fundamental collision parameters are scarce [7] and are considerably limited by the difficulties involved in measurement of three-dimensional translational and angular velocities.

A two-dimensional experiment was performed on an air table maintained by a series of jets [8]. A disk shaped puck cut from a sphere was rebounded from a fixed block of a similar material to the disk. The experiment was in a dark room enabling the trajectory of the puck to be photographed under stroboscopic lighting. Although the results of this experiment were useful in substantiating a modified Hertzian theory, it is difficult to extrapolate the two-dimensional experimental results to real three-dimensional collisions. Another two-dimensional experiment was performed. In this two-dimensional inclined chute flow experiment, the particle angular and translational velocities between collisions were calculated using a combined procedure of manually digitizing frames with software [9].

Recently, a three-dimensional experiment was performed using 3 millimeters glass

spheres. Two spheres were in free drop and there were no spin before collision. The collision procedure was grabbed by a regular camera [10].

2.2 Image Processing

It is a very challenging subject in computer vision to perform motion analysis using a sequence of images and this has been an active research topic for more than a decade [3]. Because the images must be processed individually, and the information obtained from every image has to be collected to find the motion parameters, it would be more complex than processing a single image. There are many image processing methods that are involved with motion imagery. In this section, a brief survey follows on some methods employed in this thesis.

2.2.1 Edge Detection

Edges are curves in the image where rapid changes occur in brightness or in the spatial derivatives of brightness. Edge detection algorithms can tremendously reduce the image content and thus make post-processing of these images computationally less costly. Edge detection is necessary for object recognition, feature extraction, or for other image processing applications. There are two major classifications of edge detectors, i.e. first derivative properties or second derivative properties.

The first derivative edge operator in 2x2 region of pixels is defined by

$$f_x = \frac{1}{2\tau} [f(i+1, j+1) - f(i, j+1) + f(i+1, j) - f(i, j)]$$

$$f_y = \frac{1}{2\tau} [f(i+1, j+1) - f(i+1, j) + f(i, j+1) - f(i, j)]$$

where τ is the spacing between picture cell centers. This definition can be expanded to other regions. The first derivative edge operators are directional, producing results that depend on the orientation of the edge. The gradient operator, i.e. vector (f_x, f_y) , can maintain its magnitude and orientation. The edge usually occurs at the largest magnitude of

gradient image. A squared gradient operator gets high values at edges but it does not tell us anything about the direction of the edge. This information is in the gradient itself. The Sobel's operator is a widely used edge detection operator. The main idea is to estimate the gradient (f_x, f_y) employing the eight neighbor pixels with equal weight in a 3x3 region of pixels [10,11]. The corner pixels have a factor of 2 less than the four pixels in the main directions because they are further apart (with a factor $\sqrt{2}$) and their difference vectors make 45° with the two main directions (another factor $\sqrt{2}$). A digital gradient may be computed by convolving the two windows with an image, one window giving the x component f_x of the gradient and the other giving the y component f_y . The Sobel operator is a nonlinear computation of the edge magnitude at (i, j) in an image defined by $\sqrt{(f_x^2 + f_y^2)}$

and its angle is defined by $\tan^{-1} \left(\frac{f_x}{f_y} \right)$. Canny's edge operator follows three main perfor-

mance criteria for edge detection: good detection, good localization and a singular response to a single edge [12]. It supposes that edge detection is performed by convolving the noisy edge with a spatial function and by marking edges at the maxima in the output of this convolution. The first two criteria on their own are inadequate to produce a useful edge detector and the third criterion gives an operator that has a very low probability of giving more than one response to a single edge. An analytic form for the operator was found, which is a sum of four complex exponents and is approximated by the first derivative of a Gaussian. In one dimension, the maxima in the output of this first derivative operator correspond to the zero-crossings in the output of a second derivative operator.

In second derivative edge operators, edges are detected as the point where the second derivatives of the image cross zero, i.e. the sign is different between detected edge pixels. The Laplacian and quadratic variation are rotationally symmetric operators and the quadratic variation happens to be equal to the square of the Laplacian but only the Laplacian retains the sign of the brightness difference across the edge. This allows us to reconstruct the original image from the edge image [13]. Mexican hat operators are filters of the

form $\nabla^2 G$ where ∇^2 is the Laplacian and G is the two dimensional Gaussian distribution [14]. The main idea is to smooth the image with a Gaussian shaped filter and then find the edges using the laplacian in the smoothed image. The operator introduced by Marr & Hildreth is probably the most widely used operator in this class [15]. They propose using zero crossings of the operator $D^2G(x, y)$ on the given image, where $G(x, y)$ is a two-dimensional Gaussian distribution function and D^2 is the second derivative operator for detecting intensity changes in the image. The Gaussian operator is used to satisfy localization requirements in both the spatial and the spectral domains. For the sake of reducing computations, the operator D^2G is replaced by the rotation invariant operator $\nabla^2 G$ also called Laplacian of Gaussian or LoG. Zero crossings of $\nabla^2 G$ at different scales are obtained by using $\nabla^2 G$ with different widths. If the expected edges are closed, the detected edge using LoG produces a better result than using first derivative operators.

2.2.2 Ellipse and Circle Detection

There are several useful methods to detect quadratic surfaces or curves, such as least squared approach, Hough transforms [16-18] and fuzzy clustering technique [19-23]. These techniques are efficient but each has its own constraints. If edge data are available in the form of an object boundary itself, the least squared technique can give a good result. But if the objects are spread out over the image space, or there is noise that is sometimes unavoidable, the result is poor.

The Hough Transform, HT, was first introduced as a method of detecting complex patterns of points in binary image data [16]. The main idea behind HT methods is to involve a mapping from features in the image space, which consists of image pixels, to sets of points in the parameter space which consists of parameters describing the features of images. In these methods each image point is treated independently and therefore the method can be implemented using more than one processing unit. It can recognize partial

or slightly deformed shapes. The HT method is robust and random image points or noise points are unlikely to contribute much to the accumulator bin consisting of the processing results of all image points. This is the major advantage of HT. The major disadvantage, however, is that it requires a very large amount of computer storage and that it is very computationally intensive. Several methods have been proposed to overcome these disadvantages. For reducing the computation cost, the problem was broken down into multiple stages through the use of gradient information [17]. For decreasing the computer storage, adaptive Hough transform, AHT, was introduced, where a small accumulator array and its resolution and position are iteratively adjusted [18]. These two approaches can be used together thereby decreasing the storage and computation requirement compared with the conventional HT, but if it compared with adaptive fuzzy c-shell algorithm, AFCS [19], which will be introduced below, it still has a high cost in the computation and computer storage in the circle and ellipse detection.

There are several fuzzy clustering algorithms available to detect ellipses and circles in images. Most of these algorithms are based on objective function minimization to generate a partition data where the weighted sum of the distances of the feature vectors from cluster prototypes are minimized. The cluster prototype can be a point or a line or a plane, etc. [20]. The fuzzy memberships are utilized as weighting factors. The partitions are either hard or fuzzy. Thus if the prototype is a circle, the distances are measured from the circle and the algorithm tends to detect circles in the data. However, until recently, it has been difficult to detect clusters other than the linear types. The fuzzy c-shell algorithms created by Dave [19,21-23] can detect the clusters that are described by circles and ellipses. This opens a new class of algorithms where the prototype can be a non-linear curve or surface [24]. In chapter 4, we will experimentally evaluate the AHT and AFCS algorithms.

CHAPTER 3

EXPERIMENTAL SETUP

3.1 Collision Mechanics

The main objective of this thesis is to measure the properties of particles in collision, i.e., the coefficient of friction, and normal and tangential restitution coefficients. In order to compute these quantities, the pre- and post-kinematic trajectory of two spheres colliding in three dimensional space needs to be obtained. This data can be processed to compute the desired properties using a known collision operator [2]. A brief description of the relevant definitions and equations from [2] is presented below. The change in the relative velocity of two colliding spheres can be modeled using normal (e) and tangential (β) coefficients of restitution, such that $0 \leq e \leq 1$ and $-1 \leq \beta \leq 1$. In what follows, primed and unprimed symbols are the values immediately after and before a collision, respectively. These coefficients can be defined by the relations below, where the incident normal and angular velocities after a collision are altered such that

$$\hat{r}_{12} \cdot v'_{21} = -e(\hat{r}_{12} \cdot v_{21}) \quad \text{and} \quad \hat{r}_{12} \times g'_{12} = -\beta(\hat{r}_{12} \times g_{12})$$

where \hat{r}_{12} is a unit vector connecting the centers of particles 1 and 2, $g_{12} \equiv v_{21} - \frac{\sigma}{2}(\hat{r}_{12} \times \bar{\omega}_{12})$ is the total relative velocity of the contact point, $\bar{\omega}_{12} \equiv \bar{\omega}_1 + \bar{\omega}_2$, and $v_{21} = v_1 - v_2$. The β values of -1, 0 and +1 physically correspond to contacts which are perfectly smooth, rolling and perfectly rough, respectively. The model admits either a sliding contact, where it is assumed that the tangential force is at the friction limit during the contact, or a rolling contact. This is depicted in Figure 3.1a, where β_0 is a predetermined constant value of the rotational restitution coefficient. If the assumed sliding solution yields a β value greater than β_0 , then the rolling solution is operative.

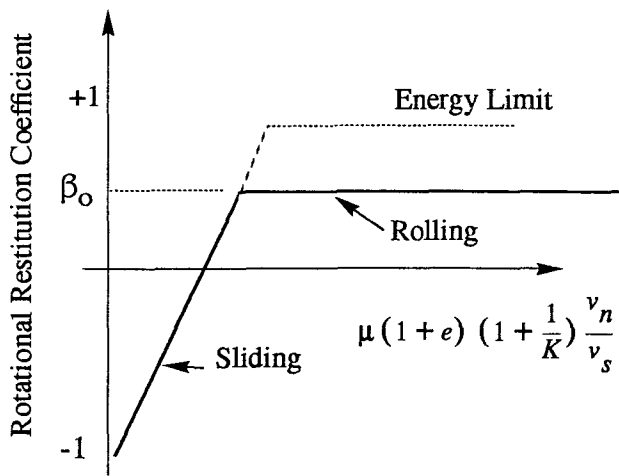


Figure 3.1a

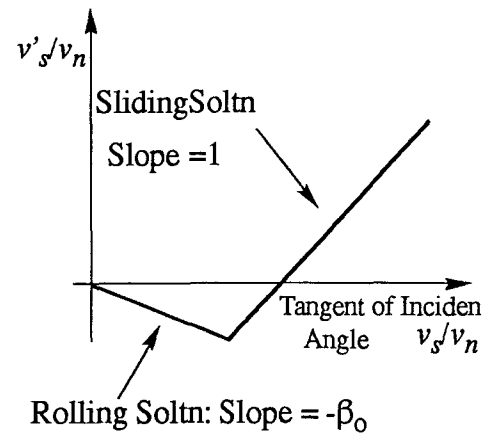


Figure 3.1b

In above figures, (v_s/v_n) is the ratios of the normal to tangential velocities before the collision. The rolling and sliding solutions in Figure 3.1b are given by the equations,

$$\frac{v'_s}{v'_n} = -\beta_0 \frac{v_s}{v_n} \text{ and } \frac{v'_s}{v'_n} = -\mu(1+e)\left(1 + \frac{1}{K}\right) + \frac{v_s}{v_n}, \text{ where } K \equiv \frac{4I_o}{m\sigma^2} = \frac{2}{5} \text{ for spheres}$$

of diameter σ . For spheres of equal mass and diameter, the values of v_s and v_n are com-

puted as $|v_s| = \left| v_t + \frac{\sigma}{2} \hat{r}_{12} \times (\bar{\omega}_{12}) \right| = v_s$ and $v_n = v_{12} \cdot \hat{r}_{12}$, where the tangential component of the velocity is given by $v_t = v_{12} - v_n \hat{r}_{12}$.

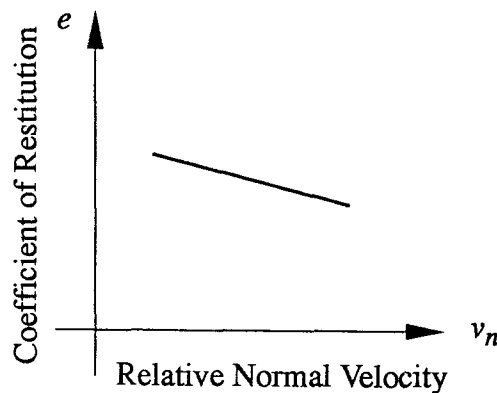


Figure 3.1c Coefficient of restitution with relative normal velocity

When the relative normal velocity increases, the coefficient of restitution decreases [30]. This is depicted in Figure 3.1c.

The steps required to compute the needed coefficients include computing the normal component of relative velocity v_n , and the tangential component of surface velocity v_s , both before and after the collision, and then fitting appropriate lines to many data points plotted in the form of Figure 3-1b.

3.2 Experimental setup

The main idea is to experimentally determine the rolling and sliding solution curves as depicted in Figure 3.1b. This is accomplished by computing values of v_s/v_n directly before and after the collisions. By making use of the equations which define these two solutions and the experimental data, it is possible to calculate the values of μ , β_0 and e . Determination of the necessary kinematic data, i.e., v_s/v_n and v'_s/v_n , is done with the Kodak system and subsequent motion analysis of the video images. The details of the method are described in chapter 4 and chapter 5.

Since collision parameter evaluation of spheres requires estimation of pre- and post-collisional position, velocity and angular velocity vectors of the spheres, an experiment apparatus is devised to collide two one-inch white nylon spheres in free space. In order to determine the orientation of spheres, random markers are drawn on their surfaces.

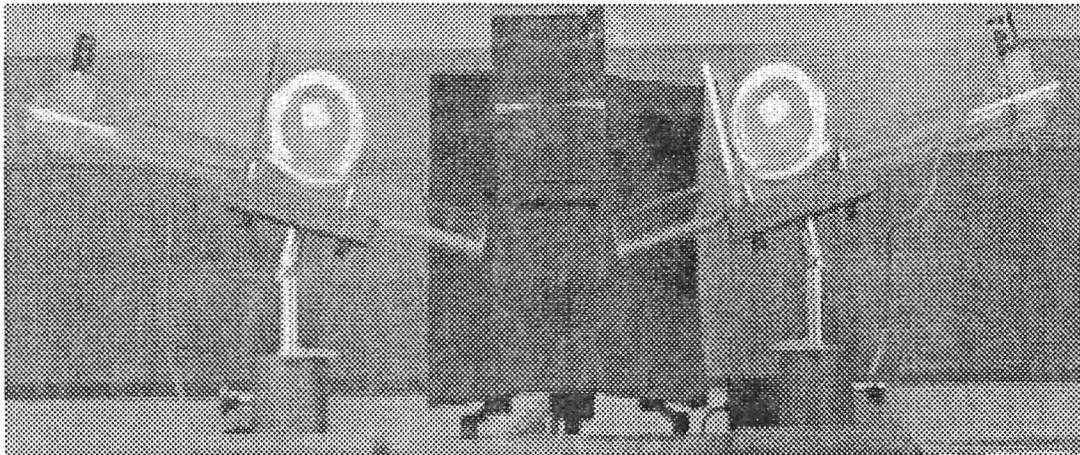


Figure 3.2a The first experimental apparatus

A picture of the first apparatus is shown in Figure 3.2a. Two identical plastic tubes are mounted on the stands which are installed on a base. The tubes themselves are able to

rotate about the stands and can be set the desired angle so that two spheres can collide at any desired angle. The spheres are held at the top of tubes by the solenoids which mounted on the end of the tube. The solenoids are connected to a triggering device controlled by a PC486. When the triggering device is activated the PC486, the solenoids are released and spheres roll down in the tubes under the action of gravity. When they exit the tubes, they are in free flight and collide in space. In this apparatus, since the spins are opposite to each other, the values of relative surface velocities are very small, so that this apparatus does not provide a wide range of values for pre-collision ratio v_s/v_n .

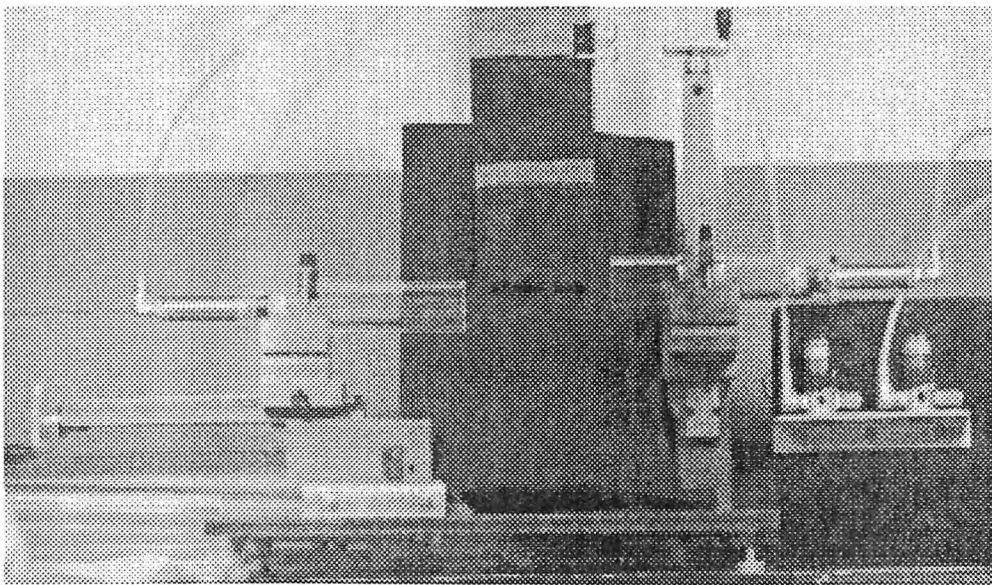


Figure 3.2b The second experimental apparatus

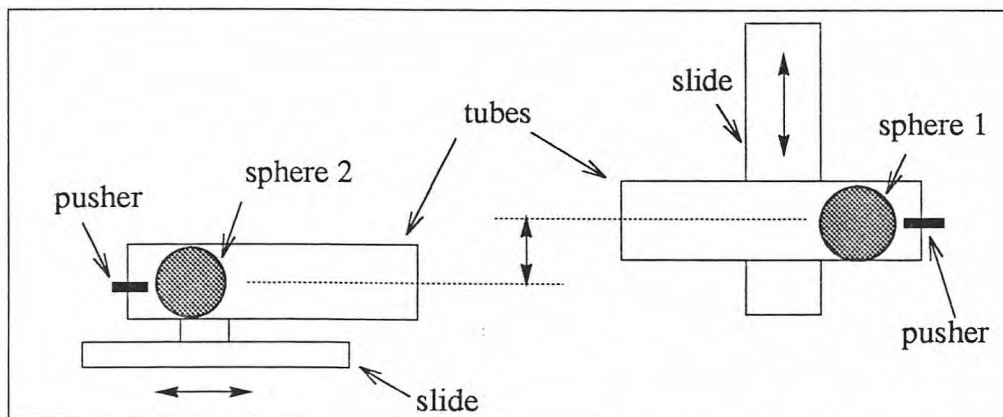


Figure 3.2c Schematic diagram for the second apparatus

In order to obtain the wide range of the values of v_s/v_n , a better experimental

apparatus is designed and built. In this apparatus, two adjustable micro-positioning slides are installed on a base. One is mounted horizontally, while the other is placed vertically. A plastic tube is then mounted on the each micro-positioning slide in the horizontal direction and in a vertical plane. This allows a range of vertical offsets between the center lines of tubes as well as horizontal displacement between the tube ends. Consequently, this flexibility allows a wide range of v_s/v_n values. The spheres are inserted into the tubes and firmly placed against pressure pushers mounted at the end of tubes. The pushers are controlled by a triggering system connected to the PC486. The PC486 can control the triggering system and Kodak Ektapro 1000 high speed video system using software previously developed [25]. When triggering system controlled by the PC486 turns on the pressure pushers, they push the two spheres at the same time and the spheres are shot out from the tubes. The two spheres are in free flight upon release from the tubes and collide in space. The whole collision procedure is recorded by Kodak Ektapro 1000 system. The pressure pushers are connected to the pressure regulator which adjusts the value of pressure in the pushers. This enables an adjustment of the impulse which the pushers impact to each sphere so that a range of collision velocities can be obtained. Figure 3.2b and 3.2c show the picture of the experimental device.

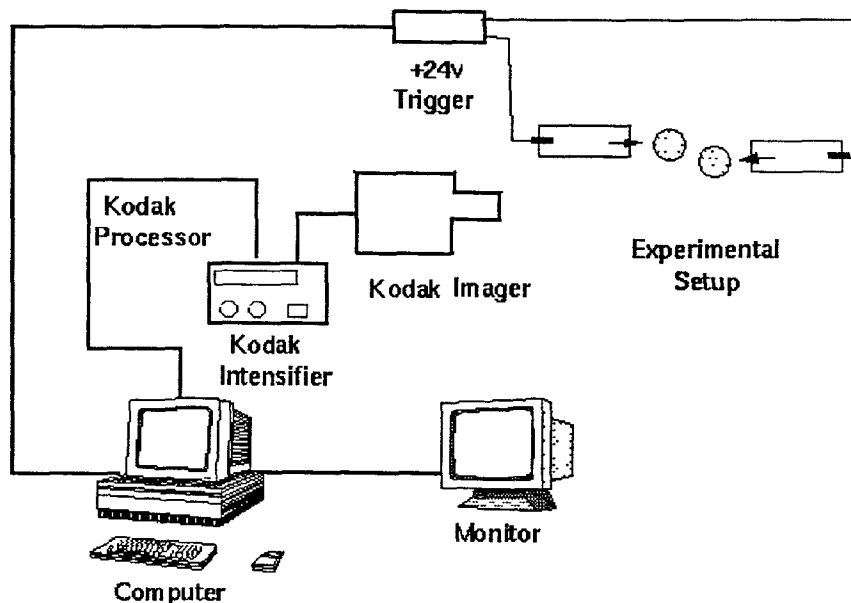


Figure 3.3 Schematic diagram

The Kodak Ektapro 1000 high speed video camera is placed in such a position that the collision area is within its field of view. A communication link exists between the PC486 and Kodak system. Images can be observed on the monitor. The recording speed is set at 1000 frames per second. The system has a playback capability of 30 frames per second and can be further slowed down by jogging the frames step by step. This allows us to observe the collision procedure at an extremely slow rate. The schematic diagram is shown in the Figure 3.3.

Analysis of collision parameters involves the estimation of translation and rotation of spheres. The translational motion is evaluated by observing the difference of the sphere centers between two interval frames, while the rotational motion is required to compute the rotation angles and rotation axis between the frames using the marker location information and then to decide the angular velocity vector.

After the collision of spheres has been recorded, it is replayed and a sequence of image frames is selected for image processing. Three pre-collision frames and three post-collision image frames are selected and the interval between frames is 2 milli-second. These images are transferred from the video tape to the PC486 hard disk. Then, the images are processed individually to find the edges, the two-dimensional centers of the spheres and markers. The markers are later matched from frame to frame to track their trajectories. The information is used to determine the collision position and pre- and post-collision velocity vectors.

Since the image obtained is two-dimensional, a mirror is placed directly above the collision space at an angle of 45 degrees from the horizontal and made to face to the camera to obtain the top view of the spheres. Thus in each image frame, the two orthogonal views are obtained, i.e., the front view and the top view. The location and orientation of spheres in three dimension is obtained through processing images of the two orthogonal views. However, since the tubes are mounted in a vertical plane in the second apparatus, the whole sphere collision procedure is also in the same vertical plane. Therefore, we only need to

process the front view image to obtain the location information about spheres because the position of spheres in the out-plane direction only has a small difference during the collision. In order to validate this, a test is done. The coordinates of collision spheres are shown in the Table 3.1 and 3.2. In this test we suppose the coordinates in the out-of-plane direction are constant in each image frame. The relative normal and surface velocities, and the relative errors are shown in Table 3.3. It can be seen, the involved relative error of relative normal and surface velocities is less than 0.5%. Therefore to simplify the experiment procedure, we only process front view images in the second apparatus experiments.

Table 3.1 Coordinates of sphere 1 (in sphere diameters)

	6 ms before collision	3 ms before collision	just collision	3 ms after collision	6 ms after collision
X direction	0	0.044	0.199	-0.025	-0.156
Y direction	0	-0.274	-0.557	-0.769	-0.978
Z direction	0	0.021	0.028	0.053	0.059

Table 3.2 Coordinates sphere 2 (in sphere diameters)

	6 ms before collision	3 ms before collision	just collision	3 ms after collision	6 ms after collision
X direction	0	0.042	0.073	0.257	0.467
Y direction	0	0.262	0.504	0.699	0.874
Z direction	0	0.001	0.016	0.020	0.015

Table 3.3 Relative normal and surface velocities (ins/second)

	v_n	v'_n	v_s	v'_s
Z = Actual value	-68.330	48.052	170.586	149.526
Z = Constant value	-68.588	47.824	170.679	149.608
Relative error %	-0.38	0.47	0.05	0.05

CHAPTER 4

IMAGE PROCESSING

In order to get the location of spheres and markers in each image frame, several image processing techniques are used, i.e., Laplacian of Gaussian operator, sub-pixel accuracy zero-crossing to get the edge of spheres and markers, labelling the edges to obtain the classifying edge points and calculation of centers using an adaptive fuzzy cluster c-shell method.

4.1 Laplacian of Gaussian Filter

Because the edge of spheres and markers is closed curves and the subpixel accuracy zero-crossing is applied, it is better to use the Laplacian of Gaussian filter to obtain useful results. The Laplacian of Gaussian, LoG, where a standard deviation of the Gaussian equals 2 is used for pre-processing to get zero-crossing image, which is equivalent to truncate grey level images at a size of 17 by 17 pixels centered by each precessing point. Then, by detecting zero-crossing in the image, we can get the edge of spheres and markers in each image frame.

4.2 Subpixel Zero-crossing

The Kodak Ektapro imager has a frame resolution of 240x192 pixels. The diameter of sphere used in the experiment is equivalent to about 40 pixels in the image. Based on the calibration of the position accuracy [3], using Canny's edge detection operator and shell clustering to find the center of a sphere, the accuracy of position measurement is one quarter of a pixel, which is equivalent to an accuracy of 1/160 of a sphere diameters. This accuracy for the position is acceptable but because the time interval between image frames is very short, i.e. two milliseconds, the accuracy for the velocities based on the time interval and the difference of center positions between images is not good enough. In order to

get more precise position data and thus obtain better accuracy of velocities, a subpixel technique is applied [4].

The idea of the subpixel method is to make an interpolation in the neighborhood of zero-crossing in the LoG image. It is equivalent to further dividing a pixel into n pixels at a dimension in the zero-crossing areas and thus the edges obtained from zero-crossing in the $n \times n$ times the original zero-crossing areas has the accuracy of $1/n$ pixels. In this method, it is assumed that the LoG image can be well fitted by a Chebyshev discrete orthogonal polynomial function in the neighborhood of a zero-crossing. Through sampling the function on a regular grid chosen, subpixel values are obtained. Since the accuracy is a pixel in the regular zero-crossing, thus, a 3×3 neighborhood grid can include the actual point of zero-crossing and has more accuracy. The variables of the polynomial function are image row and column coordinates. The 3×3 grid Chebyshev two-dimensional discrete orthogonal polynomial set $\{P\}$ is

$$P_0 = 1, P_1 = r, P_2 = c, P_3 = r^2 - \frac{2}{3}, P_4 = rc, P_5 = c^2 - \frac{2}{3}$$

and

$$P_6 = rP_3, P_7 = cP_5, P_8 = P_3P_5$$

where r and c are the coordinates in the row and column directions, respectively, which the origin of the coordinate is at the center of 3×3 local image.

Let I_{LoG} be the LoG image. For each single precision zero-crossing at location (r, c) , the interpolation function in the 3×3 grid is given by

$$I_{LoG}(r_s, c_s) = \sum_{n=0}^8 a_n P_n(r_s, c_s)$$

where the nine constant coefficients are given by

$$a_n = \frac{\sum_r \sum_c P_n(r,c) I_{LoG}(r,c)}{\sum_r \sum_c P_n^2(r,c)}, \quad n = 0, 1, \dots, 8 \text{ and } r, c = -1, 0, 1.$$

and the r_s and c_s describe subpixel coordinates in the row and column direction.

The algorithm is:

- find the zero-crossing position in the pixel image.
- calculate the nine constant coefficients.
- compute the subpixel values in a desired number of sampling.
- find the zero-crossing position in the subpixel image.

According to Huerta and Medioni [4], this method has following advantages:

- (i) Subpixel sometimes allows to remove the ambiguity met by an edge linking procedure.
- (ii) Extracting zero-crossings with $1/n$ pixel precision using a space constant in LoG filter give very similar results to that extracting zero-crossings with pixel precision using n times of the space constant, from an n times resolution image.

In order to test the accuracy of the pixel and subpixel, an experiment is conducted. The sphere is mounted on a two-axes micro-positioning stage with 0.01 mm accuracy, Then the micro-stage motion is calibrated to the image motion. On this calibration, the sphere is moved equivalent to 2 pixels in both directions, in eight steps, each step being $1/4$ pixels. The edge detection of LoG and zero-crossing in the pixel and subpixel, having a $1/4$ pixel accuracy, is done respectively. The edge data in the pixel and subpixel is analyzed using adaptive fuzzy cluster c-shell method to find the center of sphere. The results of this experiment are plotted in Figure 4-1, where the points indicate the actual measurements in the pixel and subpixel, while the line indicates actual values. As can be seen, the results of the subpixel are much better than those of the pixel and the actual accuracy of position measured from the subpixel is much down to $1/4$ pixels.

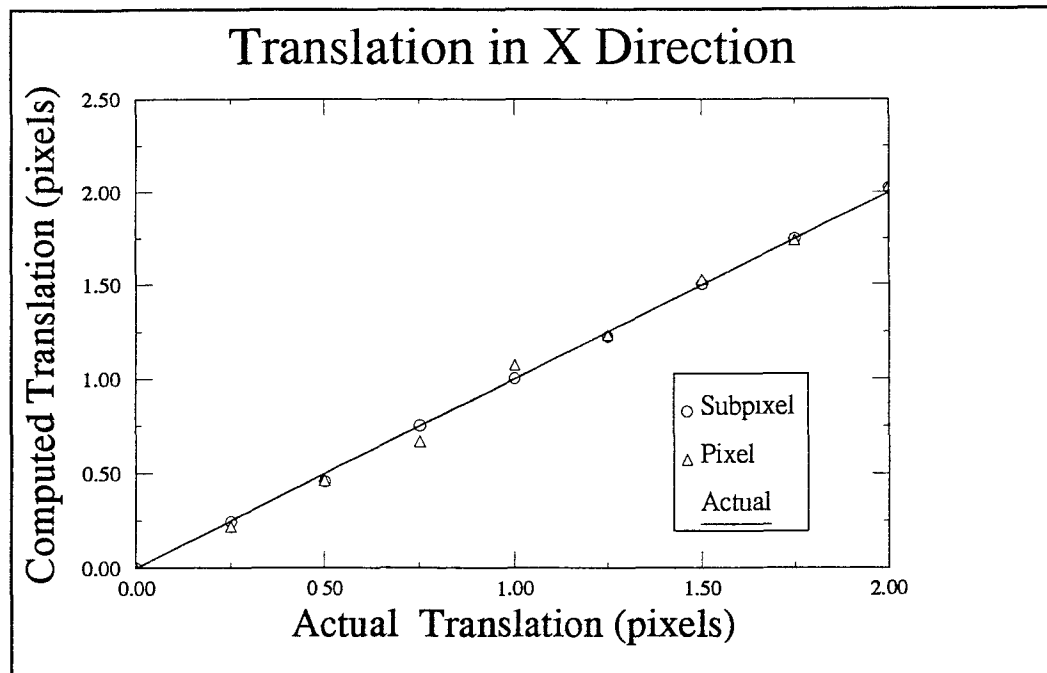


Figure 4.1a Comparison of subpixel vs. pixel in x direction translation

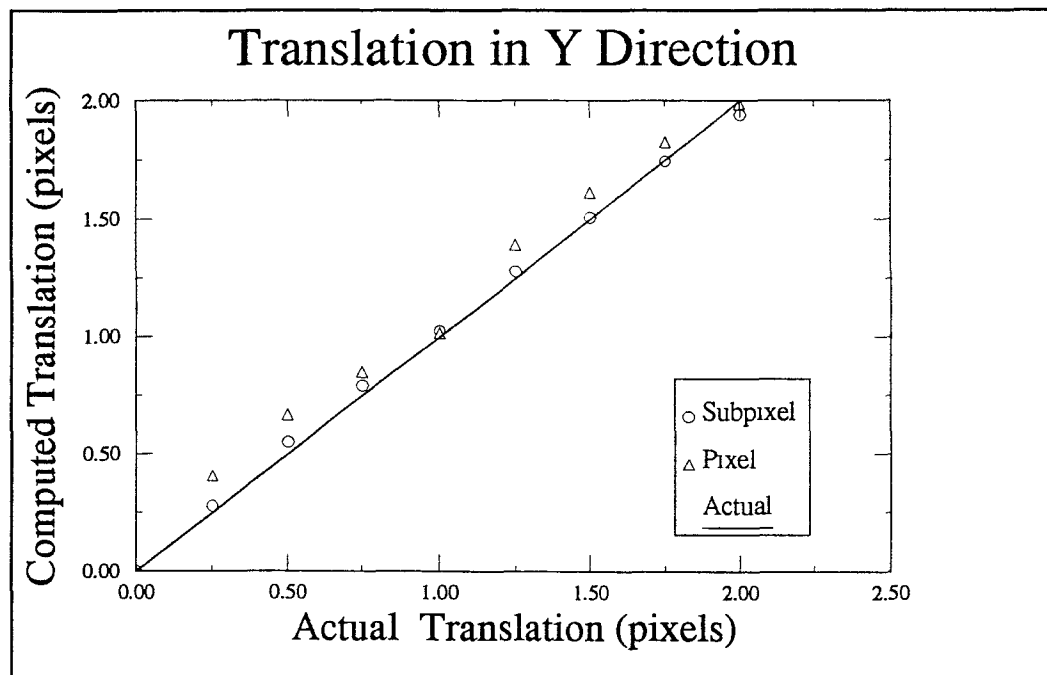


Figure 4-1b Comparison of subpixel vs. pixel in y direction translation

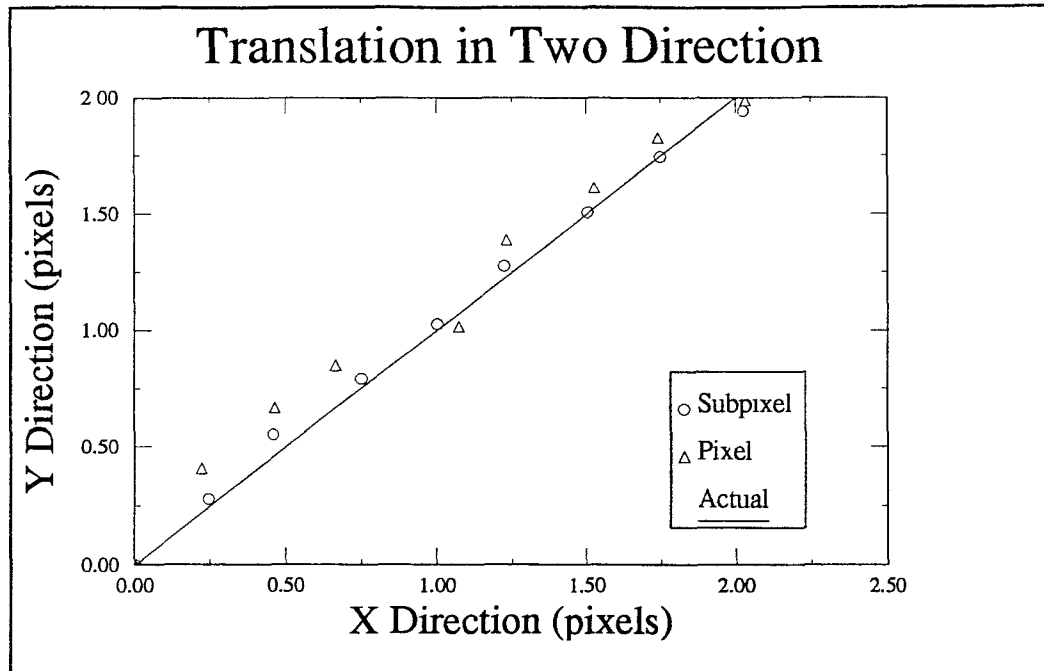


Figure 4.1c Comparison of subpixel vs. pixel in x-y direction translation

4.3 A Sequential Labelling

After the edge of spheres and markers is obtained from zero-crossing, the points of the edges are classified to determine which points belong to which set of the edge, i.e. the spheres' or the markers'. This procedure is prepared for the next processing to find the center of the spheres and markers in each image frame. The idea of labelling is to examine the pixel where its left and top side neighbors have been examined. If this pixel is not zero value and at least one of the neighbors has been labelled, one of the labels is copied. If no neighbors have been labelled, a new label is selected for this pixel. The image is scanned in this way row by row, top to bottom and left to right. The sequential labelling algorithm is as follows [13]:

Suppose a pixel A is examined, its neighbor B, C and D have been examined.

D C

B A

if (A=0) move on and examine the next pixel

```

else  if (D >0) A=D
      else  if (B>0)
            A=B
            if (C>0 and C!= B) B=C
      else if (C>0) A=C
      else A = new label.
end if

```

4.4 AFCS Algorithm

After the labelled edge image is obtained, The AFCS algorithm is applied to find the center of spheres and markers in each of the labelled edges in each image frame. Because the edge of a sphere in the image is in the form of a circle, the circle detection algorithm gives satisfactory results, while the circular markers on the surface of a sphere usually have the edges of ellipses in the image, the ellipse detection must be used. Here we also use AHT algorithm to find the sphere center and compare the two methods using the same approach as discussed in section 4.2. The results are plotted in Figure 4-2. As can be seen, the AFCS algorithm is much better than AHT algorithm in finding the location of the sphere. Besides this advantage, the AFCS algorithm has some other advantages over the AHT algorithm. The AFCS algorithm needs less computer memory, has less computing cost and does not require edge gradient information.

The sphere center obtained from AFCS in the front view image is two-dimensional. The top view image is obtained from the mirror which is placed right behind the collision space at an angle of 45 degree to the place of the camera. From this top view, the third coordinates of the sphere centers are obtained in each frame. The evaluating of the three-dimension coordinate of markers is discussed in the next chapter.

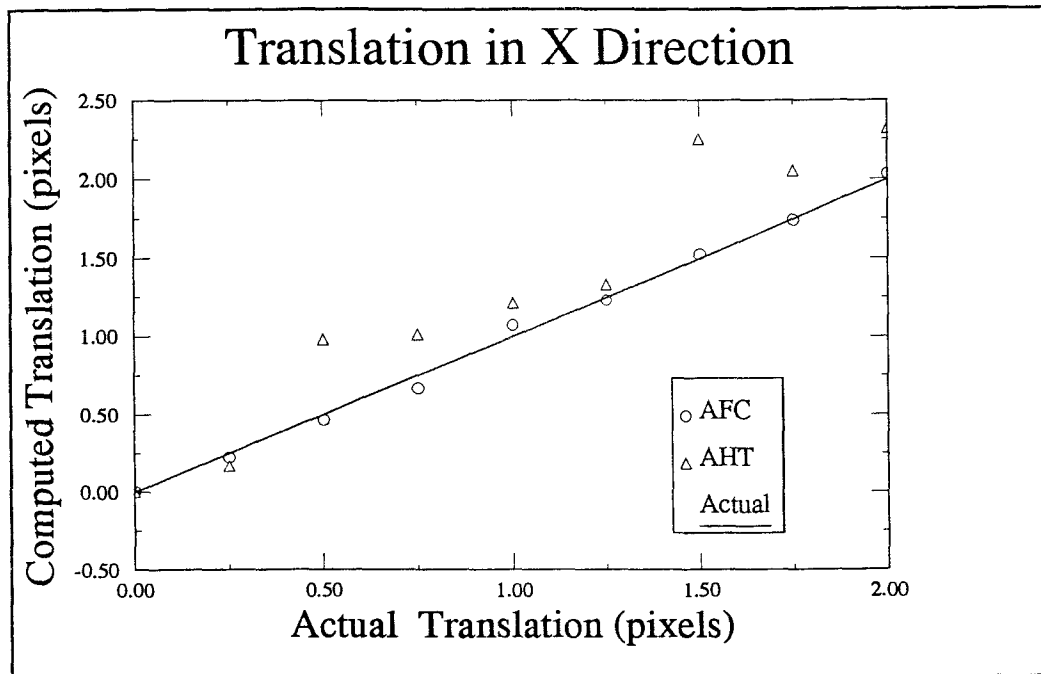


Figure 4.2a Comparison of AHT vs. AFCS in x-direction

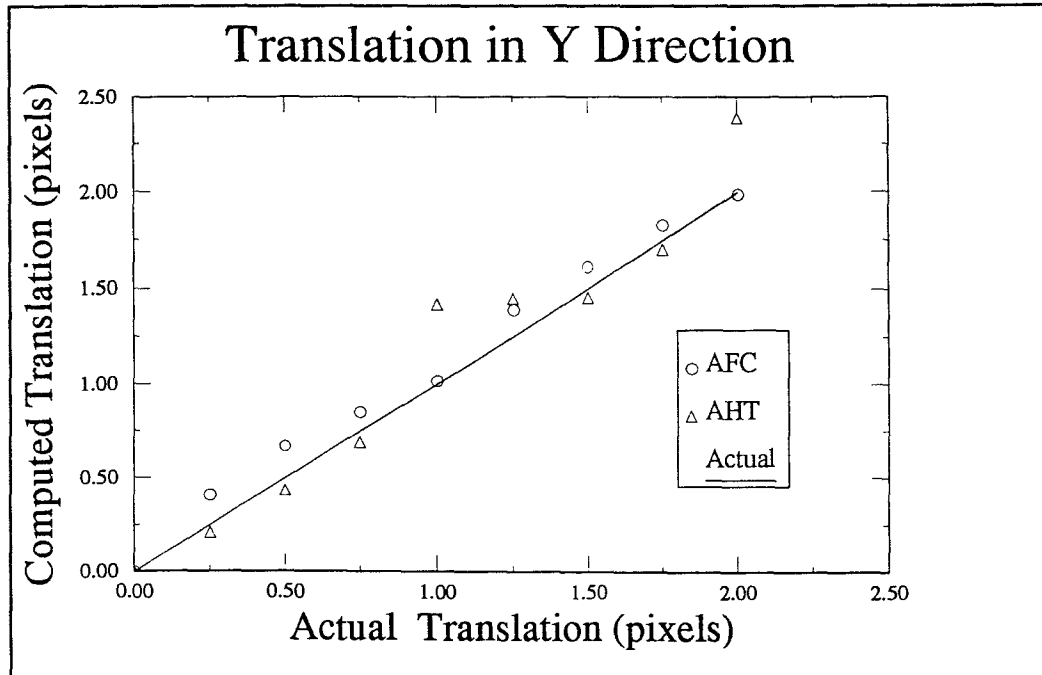


Figure 4.2b Comparison of AHT vs. AFCS in y-direction

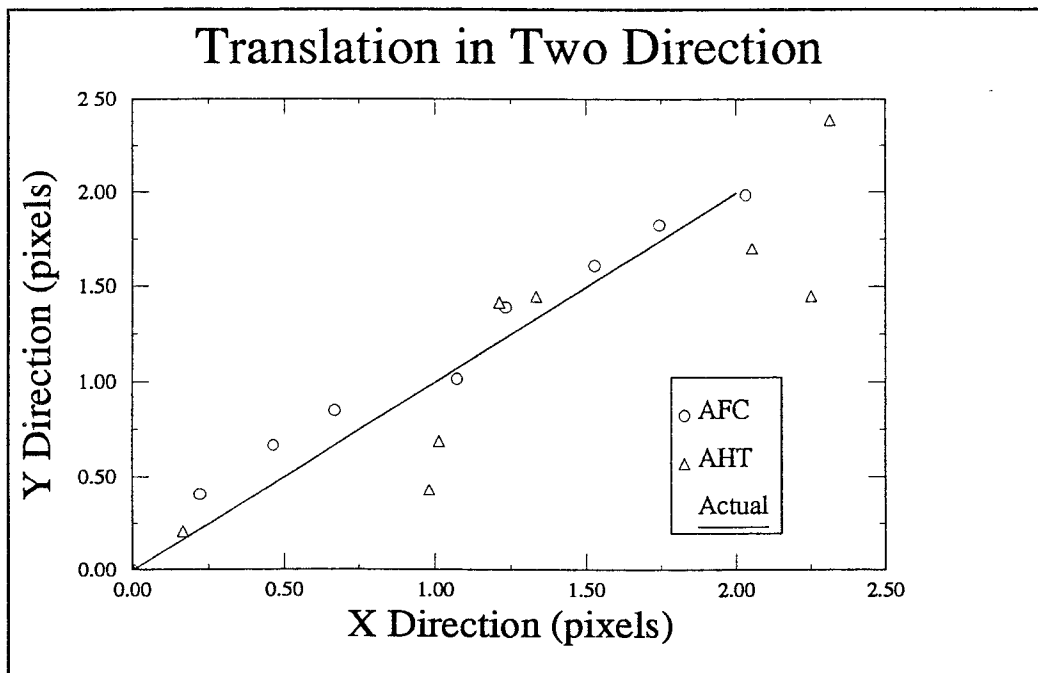


Figure 4.2c Comparison of AHT vs. AFCS in x-y direction

After the location of the spheres and markers is obtained in each frame for an experiment, the markers are matched from frame to frame to track their trajectories. Correspondence of the markers in the series of frames is achieved using the proximal uniformity constraint [6]. The details of the algorithm can be found in the Appendix B and in reference [26]. In this method, the initial correspondence must be given in the first two frames and then the corresponding markers are tracked using the previous marker location information. In most cases, the algorithm is found to work well but when the markers are too close to each other, it fails to make a correct correspondence.

CHAPTER 5

DETERMINATION OF TRANSLATIONAL AND ANGULAR VELOCITY VECTORS

In order to obtain the collision parameters of material, the translation and rotation velocity have to be computed. The translational velocity can be determined by the difference of sphere centers between frames while the rotation velocity is required to compute the rotation angles and rotation axis between the frames first using the marker position information and then to decide the angular velocity vector.

The image frames coordinates are generated by the original in the left top corner of the image. The X-Y plane is parallel to the camera, X in the vertical direction and Y in a horizontal direction. The other horizontal axis, Z, is determined by the right hand rule, to the place of the camera. The image coordinate unit is the pixel. It can be scaled to the inch unit by counting the number of pixels for a sphere diameter ($d=1''$). It can be obtained at the same time the locations of the sphere centers are computed.

5.1 Translational Velocity

First let us consider the horizontal velocities. Due to the existence of air drag on the moving spheres, the components of velocity in horizontal direction are not constants even though the changes are not great at a very short time intervals. In order to accurately determine the velocity suppose, the air reaction forces are constant, but different in each horizontal direction before or after the collision period. Thus there exist different constant acceleration in the two horizontal directions, x and z which we call a_x and a_z . The vertical velocity is determined by considering the acceleration due to gravity which is also reduced due to air drag. This is also assumed to be constant, a_z .

Suppose the frame farthest from the collision position in the three pre- or post-collision frames is called the first frame, while the frame nearest the collision position is

called the third frame. Fig. 5.1 depicts a trajectory of the sphere before and after collision. The dark filled-in circle in the figure represents the other sphere. Sphere positions in three pre- and post-collision subsequent frames are known. Then we determine the collision velocities of spheres.

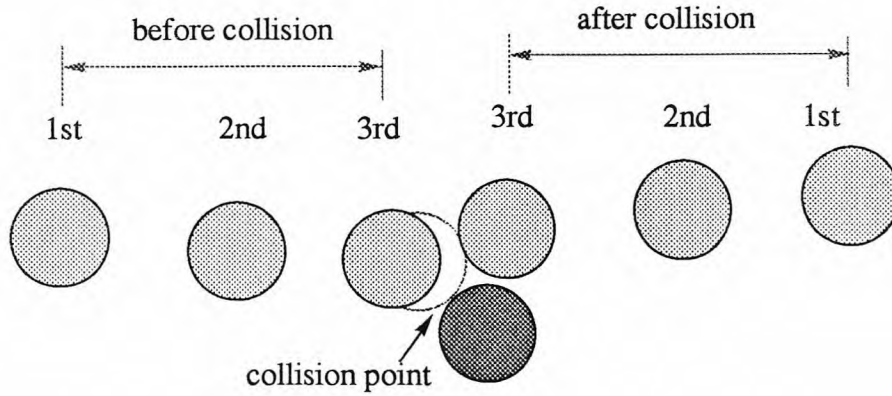


Figure 5.1 Schematic diagram of moving sphere

Based on the constant acceleration of the moving spheres, the moved distance of spheres, S , is described by

$$S = V_0 t + \frac{1}{2} a t^2 \quad (5-1a)$$

where the V_0 is initial velocity, a is the constant acceleration and t is time.

Upon rewriting Eq. (5-1a), it is found that,

$$S_{(1-2)i} = V_{1i} \Delta t_{1-2} + \frac{1}{2} a_i \Delta t_{1-2}^2 \quad (5-1b)$$

$$S_{(1-3)i} = V_{1i} \Delta t_{1-3} + \frac{1}{2} a_i \Delta t_{1-3}^2 \quad (5-1c)$$

where $S_{(1-2)i}$ and $S_{(1-3)i}$ are difference of the sphere positions between the first frame and the second frame, and between the first frame and the third frame, respectively. Here Δt_{1-2} and Δt_{1-3} are the corresponding time intervals between frames. V_{1i} is velocity at the first frame, a_i is the constant accelerations and $i = x, y, z$.

From above two equations, V_{1i} and a_i can be obtained

$$a_i = \frac{2(S_{(2-3)i} \Delta t_{1-2} - S_{(1-2)i} \Delta t_{2-3})}{\Delta t_{1-2} \Delta t_{2-3} (\Delta t_{1-2} + \Delta t_{2-3})} \quad (5-2a)$$

$$V_{1i} = \frac{S_{(1-2)i} \Delta t_{2-3} (2\Delta t_{1-2} + \Delta t_{2-3}) - S_{(2-3)i} \Delta t_{1-2}^2}{\Delta t_{1-2} \Delta t_{2-3} (\Delta t_{1-2} + \Delta t_{2-3})} \quad (5-2b)$$

where $S_{(2-3)i}$ is the difference of sphere positions between the second frame and the third frame.

Based on the assumption of the constant accelerations, the velocity at the third frame is

$$V_{3i} = V_{1i} + a_i (\Delta t_{1-2} + \Delta t_{2-3}) \quad (5-3)$$

So, from Eqs. (5-2), we can get

$$V_{3i} = \frac{S_{(2-3)i} \Delta t_{1-2} (2\Delta t_{2-3} + \Delta t_{1-2}) - S_{(1-2)i} \Delta t_{2-3}^2}{\Delta t_{1-2} \Delta t_{2-3} (\Delta t_{1-2} + \Delta t_{2-3})} \quad (5-4)$$

The components of the collision velocities can be determined by the above equations.

5.2 Angular Velocity

The rotational motion analysis involves evaluation of the rotation angle and the rotation axis of each sphere between subsequent frames in order to estimate the angular velocities.

To accomplish this, first, relative coordinates, the origin of each sphere center and the axes parallel to the image coordinate's, are required. The center positions of the markers in the correspondent sphere is described by the relative coordinates whereby the translation displacements are removed and the only rotation contribution is considered for the markers. Since the center of markers in the image is two-dimension, (X_i, Y_i) , the off image plane coordinate Z_i is determined by equation

$$Z_i = \sqrt{r^2 - X_i^2 - Y_i^2} \quad (5-5)$$

where r is radius of the sphere.

5.2.1 Determination of the Angle and Axis

In order to determine the angle and the axis of rotation, a 3x3 rotation matrix $[R]$ must be evaluated. Suppose a coordinate frame F rotates through an angle of θ to a new coordinate frame F' in which the z' axis is the same as the rotation axis having a unit vector k' . The rotation matrix between the two coordinate frames is given by,

$$\mathbf{Rot}(k', \theta) = \mathbf{R} = \begin{bmatrix} N_x & O_x & A_x \\ N_y & O_y & A_y \\ N_z & O_z & A_z \end{bmatrix} \quad (5-6)$$

The columns of \mathbf{R} are unit vectors in frame F' , i.e. N , O and A , expressed in the frame F . If a vector is described in the frame F as \mathbf{P} and \mathbf{P}' in the frame F' , then

$$\mathbf{P} = \mathbf{R}\mathbf{P}' \quad (5-7)$$

In order to construct the matrix \mathbf{R} , we will imagine that rotating around the vector k' is equivalent to rotating around z' axis of frame F'

$$\mathbf{Rot}(k', \theta) = \mathbf{Rot}(z', \theta) \quad (5-8)$$

So, Rotating \mathbf{P} around k' is equivalent to rotating \mathbf{P}' around z'

$$\mathbf{Rot}(k', \theta)\mathbf{P} = \mathbf{R}\mathbf{Rot}(z', \theta)\mathbf{P}' \quad (5-9)$$

Substitute equation (5-7) to equation (5-9)

$$\mathbf{Rot}(k', \theta)\mathbf{R}\mathbf{P}' = \mathbf{R}\mathbf{Rot}(z', \theta)\mathbf{P}' \quad (5-10)$$

Since \mathbf{R} exists invert matrix, Thus we get

$$\mathbf{Rot}(k', \theta) = \mathbf{R}\mathbf{Rot}(z', \theta)\mathbf{R}^{-1} \quad (5-11)$$

Since $\mathbf{Rot}(z', \beta)$ is rotation around z' about angle θ , we can get

$$\mathbf{Rot}(z', \theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (5-12)$$

By expanding the equation (5-11) and Simplifying the matrix according to the following relationship

(1) The dot product between any tow vectors, N , O and A , is zero for the vectors are

orthogonal.

(ii) The magnitude of any vectors N , O and A is equal 1 for they are the unit vector.

(iii) The z' unit vector is able to be got from cross product of vectors N and O .

Because the rotation axis is arbitrary, Therefore the matrix R is equal to $Rot(k', \theta)$ and the following equation is obtained

$$RRot(z', \theta)R^{-1} = \begin{bmatrix} N_x & O_x & A_x \\ N_y & O_y & A_y \\ N_z & O_z & A_z \end{bmatrix} = \begin{bmatrix} K_x K_x Vers\theta + \cos\theta & K_y K_x Vers\theta - K_z \sin\theta & K_z K_x Vers\theta + K_y \sin\theta \\ K_y K_x Vers\theta + K_z \sin\theta & K_y K_y Vers\theta + \cos\theta & K_y K_z Vers\theta - K_x \sin\theta \\ K_z K_x Vers\theta - K_y \sin\theta & K_y K_z Vers\theta + K_x \sin\theta & K_z K_z Vers\theta + \cos\theta \end{bmatrix} \quad (5-13)$$

where $Vers\theta = 1 - \cos\theta$.

This is rotation matrix. If we know all the nine components, we can get the rotation angle and axis of the sphere. Summing diagonal term of matrix in Eq. (5-13),

$$N_x + O_y + A_z = (K_x^2 + K_y^2 + K_z^2) Vers\theta + 3\cos\theta \quad (5-14)$$

Since $K_x^2 + K_y^2 + K_z^2 = 1$, therefore we get

$$\cos\theta = \frac{N_x + O_y + A_z - 1}{2} \quad (5-15)$$

Substrate symmetrically placed off-diagonal terms of matrix in Eq. (5-13)

$$O_z - A_y = 2K_x \sin\theta \quad (5-15a)$$

$$A_x - N_z = 2K_y \sin\theta \quad (5-15b)$$

$$N_y - O_x = 2K_z \sin\theta \quad (5-16c)$$

We may define the rotation to be positive about the vector K such that $0 \leq \theta \leq 180$. By squaring both sides of above equations. and combining the Eq. (5-15),

we get

$$\tan\theta = \frac{\sqrt{(O_z - A_y)^2 + (A_x - N_z)^2 + (N_y - O_x)^2}}{N_x + O_y + A_z - 1} \quad (5-17)$$

Thus we can calculate the angle of θ using above equation. From Eqs. (5-17) we get

$$K_x = \frac{O_z - A_y}{2 \sin \theta} \quad (5-18a)$$

$$K_y = \frac{A_x - N_z}{2 \sin \theta} \quad (5-18b)$$

$$K_z = \frac{N_y - O_x}{2 \sin \theta} \quad (5-18c)$$

As can be seen when the angle is very small, owing to the small magnitude of both numerator and denominator in above equations, the axis of rotation is not well defined. The vector K should be renormalized to ensure that $|K| = 1$. It can be observed again that there exists a critical case, at $\theta = 180^\circ$, the denominator in the above equations is of the form 0, so that no information about K vector can be got. In order to avoid it, equating the diagonal elements of the equation we can derive

$$K_x = \text{sgn}(O_z - A_y) \sqrt{\frac{N_x - \cos \theta}{1 - \cos \theta}} \quad (5-19a)$$

$$K_y = \text{sgn}(A_x - N_z) \sqrt{\frac{O_y - \cos \theta}{1 - \cos \theta}} \quad (5-19b)$$

$$K_z = \text{sgn}(N_y - O_x) \sqrt{\frac{A_z - \cos \theta}{1 - \cos \theta}} \quad (5-19c)$$

where $\text{sgn}(e) = \begin{cases} 1, e > 0 \\ -1, e < 0 \end{cases}$

In order to more accurately determine the components of the unit vector K , only the largest element of K is achieved from Eqs. (5-19). Others are obtained by summing pairs of off-diagonal elements of Eqs. (5-13). There are three cases to get the components. Case 1: if K_x is the largest in the Eqs. (5-19), then

$$K_y = \frac{N_y + O_x}{2K_x \text{Vers}\theta} \quad (5-20a)$$

$$K_z = \frac{A_x + N_z}{2K_x \text{Vers}\theta} \quad (5-20b)$$

Case 2: if K_y is the largest in the Eqs. (5-19), then

$$K_x = \frac{N_y + O_x}{2K_y \text{Vers}\theta} \quad (5-21a)$$

$$K_z = \frac{O_z + A_y}{2K_y \text{Vers}\theta} \quad (5-21b)$$

Case 3: if K_z is the largest in the Eqs. (5-19), then

$$K_x = \frac{(A_x + N_z)}{2K_z \text{Vers}\theta} \quad (5-22a)$$

$$K_y = \frac{O_z + A_y}{2K_z \text{Vers}\theta} \quad (5-22b)$$

From the equations above we can get the rotation axis unit vector \mathbf{K} . If the angle of rotation is in the first quarter i.e. $\theta < 90^\circ$, the Eqs. (5-18) are recommended. Else the equations from (5-19) to (5-22) are utilized.

5.2.2 Determination of Rotation Matrix

As can be seen in Eq. (5-6), the angle and axis of rotation are the function of the nine elements in the rotation matrix \mathbf{R} . If these elements are determined, we can evaluate the angle and the rotation axis.

Suppose there is a marker at position \mathbf{P}' on a sphere. When the sphere is rotated a angle, the new position of the marker, at \mathbf{P} , is constrained by $\mathbf{P} = \mathbf{R}\mathbf{P}'$. Since the corre-

spondent marker position is represented by the three-dimension coordinate, it produces three equations. If three markers is selected, they give nine linear equations and the matrix representing the nine equations is unsingular because the markers which are randomly marked do not lie on a line. So, this system has a unique solution. Theoretically, at least three points are required to obtain the nine rotation elements. However, in practice, because there exist errors in the evaluation of the coordinates. This approach will not give a good result. We propose the least-squared error approach and use at least five points to solve this problem.

Assume that P'_i and P_i represent the relative coordinates of the same marker on a sphere obtained from sequence of frames. These corresponded coordinates satisfy

$$P_i = RP'_i \quad i = 1, \dots, n \quad (5-23)$$

Because of the errors of estimation of the coordinates, there exist errors in the above equations. They are represented as

$$Er_i = RP'_i - P_i \quad i = 1, \dots, n \quad (5-24)$$

The total squared error is

$$G(P') = \sum_i Er_i^T Er_i \quad (5-25)$$

It can be observed, the total squared error is the function of nine rotation elements. differentiating the above equation to minimize the total squared error

$$\frac{\partial G}{\partial V} = 0 \quad (5-26)$$

where $V = N_p, O_p, A_p, i = x, y, z$

we can get the linear equations to solve the nine elements

$$B'D_x = C_x \quad (5-27)$$

$$B'D_y = C_y \quad (5-28)$$

$$B'D_z = C_z \quad (5-29)$$

where

$$B' = \begin{bmatrix} \sum_i X_i^2 & \sum_i X_i Y_i & \sum_i X_i Z_i \\ \sum_i X_i Y_i & \sum_i Y_i^2 & \sum_i Z_i Y_i \\ \sum_i X_i Z_i & \sum_i Z_i Y_i & \sum_i Z_i^2 \end{bmatrix} \quad (5-30)$$

$$C_w = \left[\sum_i X_i W_i \quad \sum_i Y_i W_i \quad \sum_i Z_i W_i \right]^T \quad (5-31a)$$

$$D_w = [N_w, O_w, A_w]^T \quad (5-31b)$$

where $W = X, Y, Z$.

In the actual experiments, we can get the five to eight markers in an image and obtain the rotation matrix using equations from (5-27) to (5-31).

5.2.3 Determination of the Angular Velocity

After we obtain the angle and axis of the sphere, the angular velocity is able to be evaluated. Suppose the sphere has rotated an angle of θ_{1-2} between frame1 and frame2 around an axis passing through the center of the sphere having direction cosines as K_{x1} , K_{y1} and K_{z1} and an angle of θ_{2-3} between frame2 and frame3 around an axis having direction cosines as K_{x2} , K_{y2} and K_{z2} . The components of angular velocity vector in the first or third frame are

estimated using the same way as the translation velocity. We get

$$\Omega_{1i} = \frac{\theta_{1i} \Delta t_{2-3} (2\Delta t_{1-2} + \Delta t_{2-3}) - \theta_{2i} \Delta t_{1-2}^2}{\Delta t_{1-2} \Delta t_{2-3} (\Delta t_{1-2} + \Delta t_{2-3})} \quad (5-32)$$

$$\Omega_{3i} = \frac{\theta_{2i} \Delta t_{1-2} (2\Delta t_{2-3} + \Delta t_{1-2}) - \theta_{1i} \Delta t_{2-3}^2}{\Delta t_{1-2} \Delta t_{2-3} (\Delta t_{1-2} + \Delta t_{2-3})} \quad (5-33)$$

where $\theta_{ji} = K_{ij}\theta_j$, $i = x, y, z, j = 1, 2$, respectively.

5.3 Collision Position

The collision position is able to be calculated using the sphere velocities and positions in the nearest collision frame. The distance of the sphere from this frame to the collision position can be determined by Eq. (5-1a). Since the time period from the nearest collision frame to the collision is very small, the contribution of acceleration can be ignore in the Eq. (5-1a). Suppose that the coordinates of the sphere A in the nearest collision frame are (X, Y, Z) and the velocity components are (V_x, V_y, V_z) while the sphere B's are (X', Y', Z') and (V'_x, V'_y, V'_z) , respectively. Then after the spheres flight a distance in a time period, t , they collide in space. The differences between the colliding sphere centers are given by

$$\Delta W_c = \Delta W + \Delta V_w t \quad (5-34)$$

where $W = X, Y, Z$, $\Delta W_c = W'_c - W_c$, $\Delta W_c = W' - W$, $\Delta V_w = V'_w - V_w$ and the subscript “c” describes the collision position. When the two spheres are just collision, the difference of their centers is equal to the sphere diameter. So, we get

$$D^2 = \sum_w (\Delta W + \Delta V_w t)^2 \quad (5-35)$$

Since the diameter of the two spheres is 1-inch we expend the above equation and obtain

$$at^2 + 2bt + c = 0 \quad (5-36)$$

where $a = \sum_w \Delta V_w^2$, $b = \sum_w \Delta W \Delta V_w$ and $c = \sum_w \Delta W^2$. The time t can be solved from the above equation

$$t = \frac{-b \pm \sqrt{b^2 - ac}}{a} \quad (5-37)$$

If $b^2 - ac > 0$, the equation has two solutions.

The image frame we selected is the nearest collision frame, recorded at a 1000 frames per second rate. The spheres have not collided yet in the pre-collision frame while have collided in the post-collision frame and the time begins from this frame. Thus the time t can be decided by the following relations.

- In the pre-collision frame, the time $t > 0$, otherwise $t < 0$.
- The magnitude of $t < 1$ milli-second.

The difference of the two sphere centers in the collision position is decided by Eq. (5-34) and the collision normal unit vector is calculated using these components.

CHAPTER 6

THE ACCURACY ANALYSIS OF THE EXPERIMENT

In this chapter, we will examine the accuracy of the whole technique proposed in the experiment. There are some aspects involving the accuracy of the experiments, such as the limited resolution of the camera and the error in evaluating the centers of the spheres and markers that give rise to errors in the velocities. In Chapter 4, we have discussed the subpixel technique to solve the issue of the limited resolution of the camera, the fuzzy c-shell technique to get the more accurate centers of the spheres and markers. In addition, experiments have been conducted to test the translational accuracy by using these techniques. In what follows, additional experiments are conducted to examine the accuracy of the overall experimental procedure.

To test the accuracy of the translation, we simply dropped a steel sphere and recorded its motion to compute the gravitational acceleration. The sphere is held by a magnetic trigger which is fixed in a frame above the floor. The trigger is controlled by a PC486. When the PC486 turns the trigger switch off, the sphere drops down, and at the same time the camera catches the images of the sphere dropping process while the video system stores the images on a tape and the rate of the camera to grab the image is 250 frames per second. The images are then transferred to the PC486. The motion recorded on the imager is calibrated to the real motion by using the knowledge of the pixel size of the sphere. There are over 10 frame images processed for each sphere drop experiment. The interval chosen between image frames processed is 8 milli-second. Subpixel and fuzzy c-shell techniques are employed to detect the edge and to find the center of the sphere in each image. There are approximately 33 pixels in the diameter of the sphere in the processed image. Since, in the collision experiment, more than 40 pixels are used, greater accuracy should be achieved. After the center coordinates of the sphere for all images of an experiment are obtained, regression analysis is used to fit the curve using quantic functions

from which the gravity acceleration value is obtained. A typical experimental result is plotted in Figure 6.1.

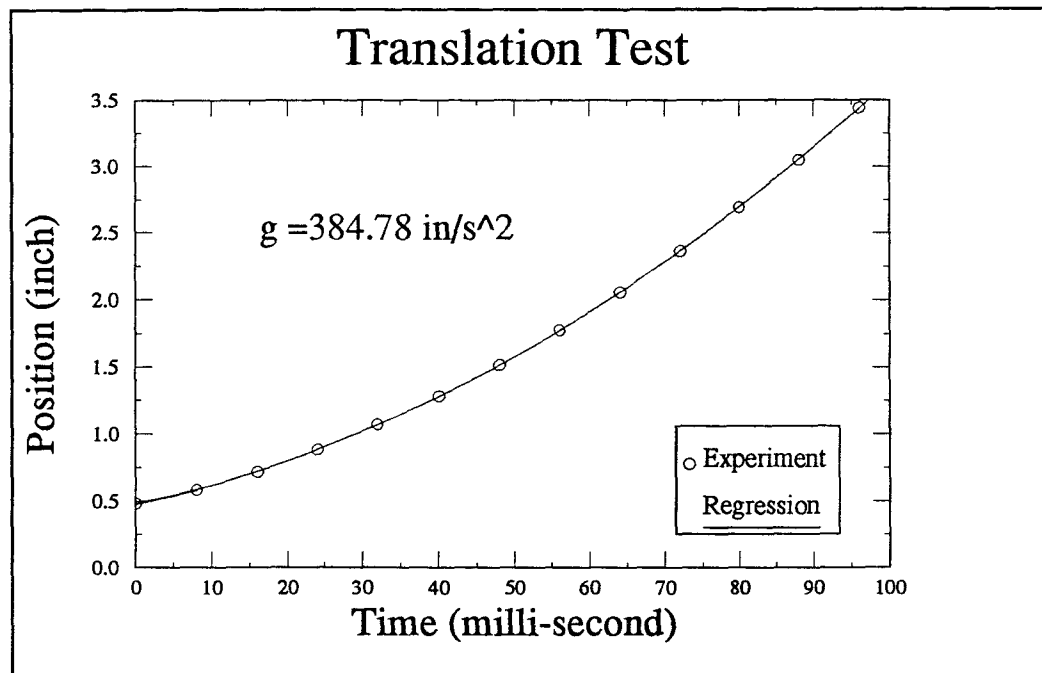


Figure 6.1 Results of translation test

For the statistic analysis ten such experiments were performed to verify repeatability of the results. Table 6.1 lists the computed values of g . As can be observed, the accuracy of translation measurement is very high even though the number of pixels in the sphere diameter in this experiment is less than the actual collision experiment.

Table 6.1 Summary of Results for Gravity Acceleration (inch/sec^2)

384.8	383.9	384.6	384.5	382.9	385.4	383.8	383.7	385.7	385.4
Average Gravity Acceleration = 384.47									
Standard Deviation = 0.89									
Normalized Error [†] = 0.5%									

†. Normalized by $g = 386.4 \text{ inch}/\text{sec}^2$

In order to test the accuracy of measuring rotations, a similar experiment is also conducted. In this experiment, a circular disk on which there are five dark markers is mounted on a motor shaft to test the rotation accuracy of the Kodak Ektapro 1000 system and whole processing method. The motor is controlled by a distributor and its rotation rate is displayed on a meter with an accuracy of 0.5 rpm. The camera grabs the images of rotating circular disk, the video system stores them on the tape and later, images are transferred to the PC486. The rate of rotation for the motor is 997 rpm and the camera catches the images at 1000 frames per second. There are 15 image frames processed and the minimum interval between image frames is 1 milli-second. Therefore, the minimum angle of rotation between frames is 5.982 degrees. The accuracy of different rotation angles is checked. For each rotation angle, data from many sequential frames are used. For instance, 14 calculations are done between image frames for the 5.982 degree test: frame1-frame2, frame2-frame3, etc. The least squares method is employed to compute the disk rotation angle between frames by making use of all five feature points. The Table 6.2 summarizes the angle results for the first 45 degrees, including the relative error and standard deviation. Comparison with the computed rotation is done by fitting a linear regression line to the actual rotation (based on experimental values read on meter). This is shown on Figure 6.2, where the points indicate the computed data, while the line indicates the actual value of rotation.

Table 6.2 Summary of Results for Angles of Rotation (degree)

Actual Angle [†]	5.982	11.964	17.946	23.928	29.910	35.892	41.874
Repetitions	14	13	12	11	10	9	8
Computed Angle	5.993	11.974	17.973	23.961	29.965	35.938	41.864
Rel. Error% [‡]	0.18	0.08	0.15	0.14	0.18	0.13	-0.02
Deviation	0.221	0.262	0.272	0.202	0.232	0.245	0.247

†. All angles are measured in degrees

‡. Relative error% = (Actual - Calculated)/Actual * 100

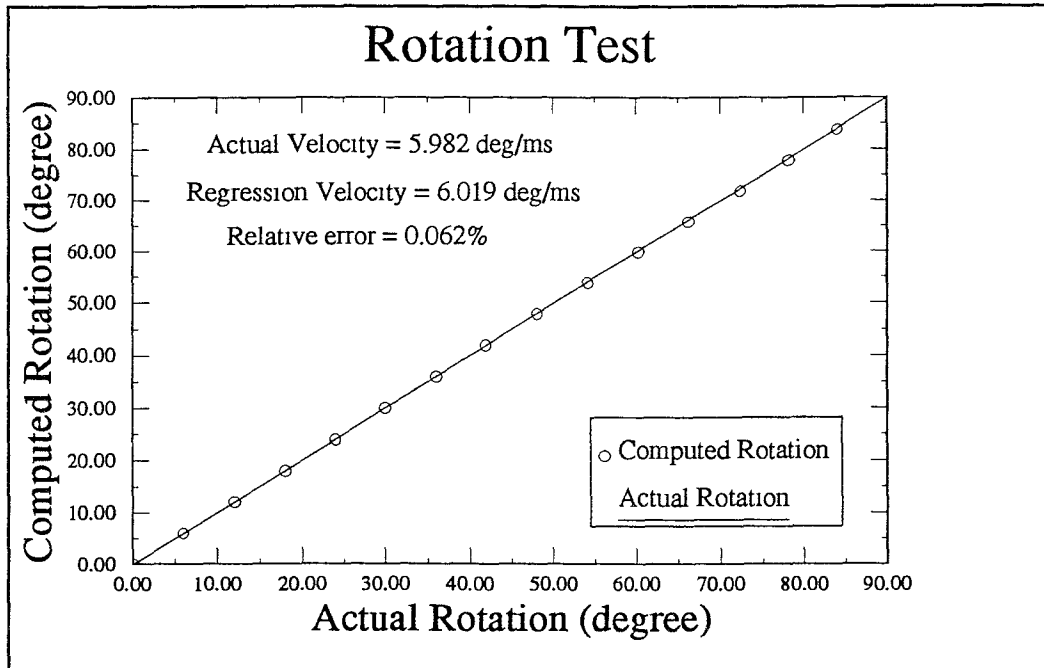


Figure 6.2 Results of rotation test

Based on these tests, it is concluded that the accuracy of the overall procedure is very high for use in the actual experiments.

CHAPTER 7

DETERMINATION OF THE COLLISION PARAMETERS AND RESULTS

7.1 Determination of Collision Parameters

After we obtain the collision position of the spheres and the velocity vector expressed in image coordinates, we can calculate the normal direction relative velocity and the tangential direction relative velocity based on frictional hard-sphere collision operator [2].

Consider a collision between two frictional, inelastic spheres with centers located at r_a, r_b , traveling with velocities V_a, V_b before collision, V'_a, V'_b after collision and having rotational velocities Ω_a, Ω_b .

$$\hat{r}_{ab} = \frac{r_b - r_a}{|r_b - r_a|} = \text{unit vector from the sphere } a \text{ to the sphere } b \text{ (at contact),}$$

$$V_{ab} = V_b - V_a = \text{relative velocity,}$$

$$V_t = \hat{r}_{ab} \times (V_{ab} \times \hat{r}_{ab}) = \text{tangential direction relative velocity,}$$

$$V_s = V_t + \frac{D}{2} (\hat{r}_{ab} \times \Omega_a + \hat{r}_{ab} \times \Omega_b) = \text{relative surface velocity (tangential direction),}$$

$$\hat{k}_s = \frac{V_s}{|V_s|} = \text{unit vector in direction of incident surface velocity.}$$

For the spheres before collision:

$$v_n = V_{ab} \cdot \hat{r}_{ab} = \text{normal component of relative velocity,}$$

$$v_s = V_s \cdot \hat{k}_s = \text{tangential component of surface relative velocity.}$$

For the spheres after collision:

$$v'_n = V'_{ab} \cdot \hat{r}_{ab} = \text{normal component of relative velocity,}$$

$$v'_s = V'_s \cdot \hat{k}_s = \text{tangential component of surface relative velocity.}$$

Application of the operator also gives rise to two straight lines if plotted as $-v'_s/v_n$, vs. the tangent of the effective incident angle, v_s/v_n . The sliding solution line is decided by the equation,

$$-\frac{v'_s}{v_n} = -\mu (e + 1) \left(1 + \frac{1}{K}\right) + \left(\frac{v_s}{v_n}\right) \quad (7-1a)$$

and the equation for the rolling solution line is,

$$-\frac{v'_s}{v_n} = -\beta_0 \left(\frac{v_s}{v_n}\right) \quad (7-1b)$$

where $K = 2/5$ for the uniform density spheres, β_0 is the rotational restitution coefficient, also defined as $\beta_0 = -\frac{v'_s}{v_s}$, μ is the coefficient of sliding friction and e is the restitution coefficient, also defined as $e = -\frac{v'_n}{v_n}$.

For the convenience, all the vectors are calculated in the form of components on the coordinate axes. When the normal component of relative velocity and the tangential component of surface relative velocity in pre- and post-collision are obtained, the velocities are employed to determine the collision parameters. Since there unavoidably exists the error for the evaluation of the velocities, in order to obtain more accurate results about collision parameters, a linear regression analysis is applied on the data and then this is fit to the model given by Eqs. (7-1).

7.2 Results

In this section, the results for the experiments of two spheres colliding in space are presented. The images chosen in the experiment are processed and very good edge images are obtained using the subpixel zero-crossing technique. The AFCS algorithm created by Dr. Rajesh Dave is utilized and this yielded excellent results for the location of the sphere and marker centers. The figures, tables and graphs in the section present a good explanation of automatic evaluation of collision coefficients by the image processing technique.

Figure 7-1 shows an image before collision where the top two spheres are the top view image obtained from the mirror, the bottom two spheres are the front view image and there are some markers on the spheres to determine the orientation of the spheres.

Figure 7-2 shows a sequence of three pre-collision image frames and three post-collision frames selected in an experiment.

Figure 7-3 shows the sphere and marker edge images and their locations where plus symbols indicate marker locations detected by AFCS algorithm.

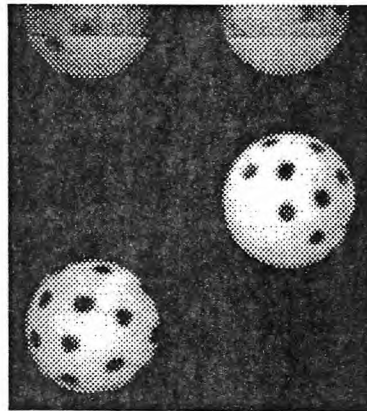
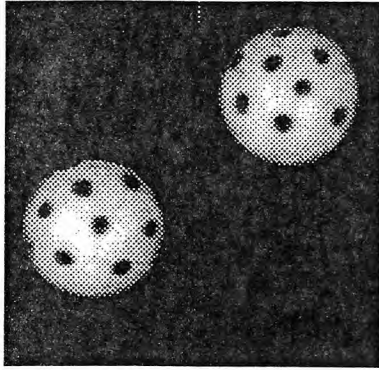
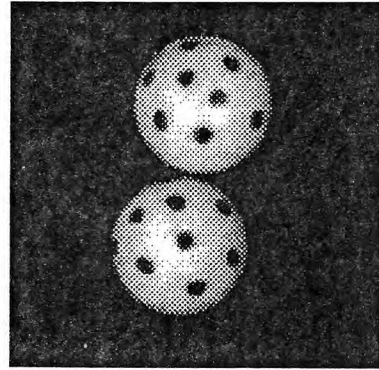


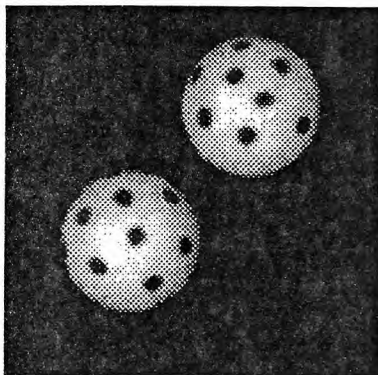
Figure 7.1 A grey level image before sphere collision



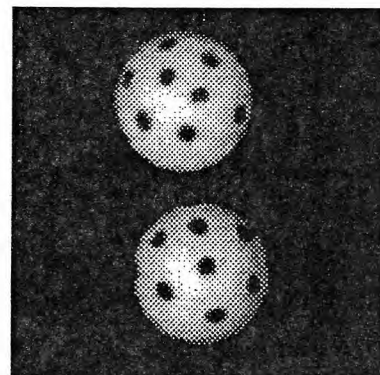
No. 1



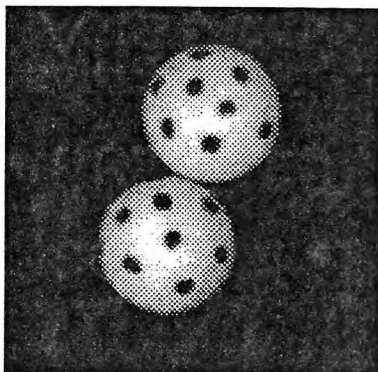
No. 4



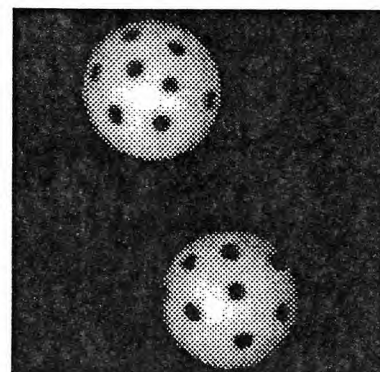
No. 2



No. 5

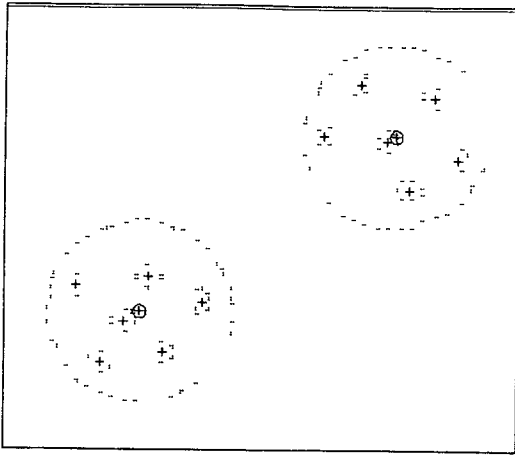


No. 3

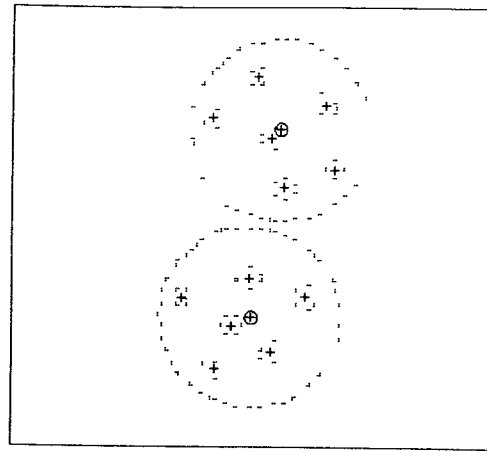


No. 6

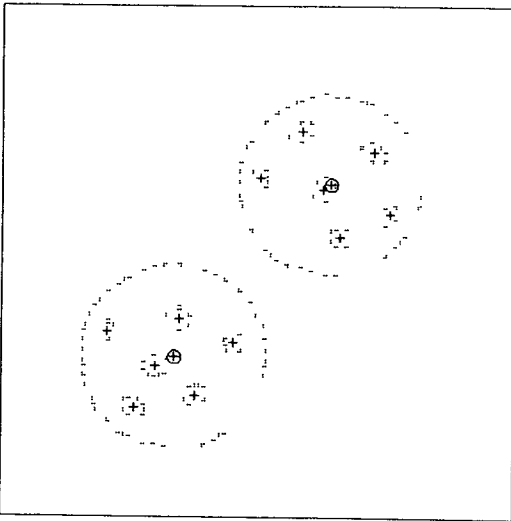
Figure 7.2 A sequence of grey level images.



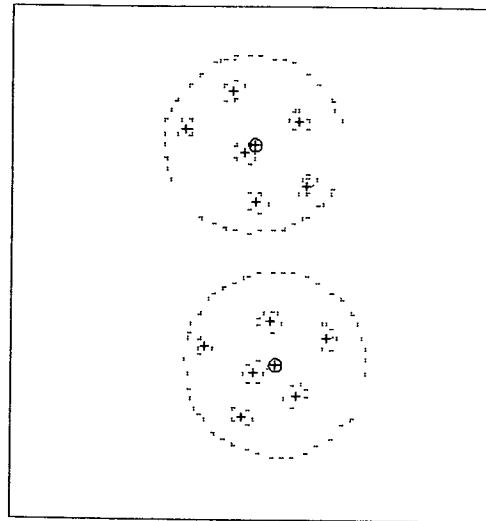
No. 1



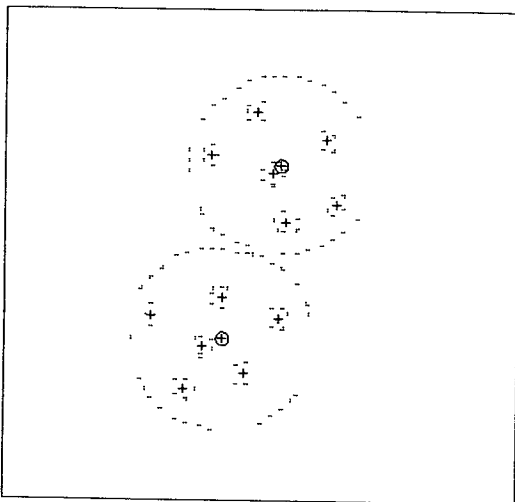
No. 4



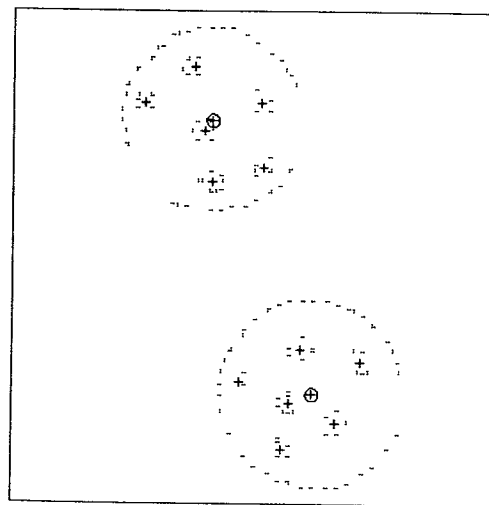
No. 2



No. 5



No. 3



No. 6

Figure 7.3 A sequence of edge images.

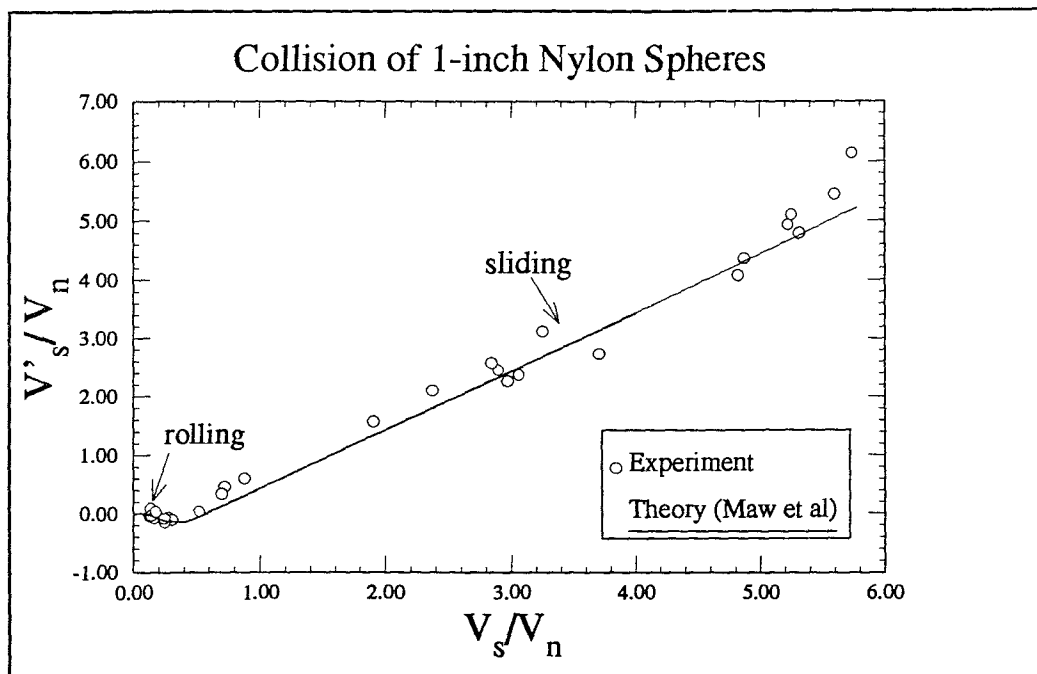


Figure 7.4 Experimental results for the collision parameters

Figure 7-4 shows the results for evaluation of collision coefficients for all experiments where the sliding solution line and the rolling solution line are simulation results based on Maw et al theory [28] while the circle points are the experimental results. The parameters used in the simulation [28] are shown in Table 7.1.

Table 7.1 Properties of Spheres

Density (g/cm^3)	Sphere Diameter (cm)	Coefficient of Friction	Poisson's ratio	Young's Modulus 10^{10} dynes/ cm^2
1.15	2.54	0.08	0.25	2.3-3.8

From this figure and using Eqs. (7-1), two best fit lines are obtained, and the collision parameters are computed. For one inch nylon spheres, the values obtained are $e = 0.74$, $\beta = 0.69$ and $\mu = 0.08$ respectively.

CHAPTER 8

CONCLUSION

The Kodak EktaPro1000 high speed image system has been effectively used to get clear images of the two colliding spheres.

The subpixel zero-crossing technique has proven to be successful for detecting edges in the limited image resolution. This technique based on the interpolation indeed gives subpixel precision. It is very useful to get high accuracy edges especially in the low resolution images.

The algorithm based on the adaptive fuzzy c-shell theorem has proven to be remarkably successful in detecting shapes characterized by sphere and ellipse. It has been shown to have some advantages over the AHT method. By comparing with AFCS, the AHT method is slower in computation, is less accurate in finding the location of spheres and ellipses, requires more memory and sometimes needs edge gradient information.

The method to calculate the velocities of the two colliding spheres has shown to be effective. It has considered the influence of the air reaction force so that it may obtain the more accurate velocities at the colliding position and get better precision of the colliding position computed by the velocities.

The collision coefficients are calculated by linear regression about all experiments and compare with theoretical predictions of Maw et al. In the rolling solution, the results are fit very good while in the sliding solution, data shows a little scatter when the ratio of v_s/v_n is greater than 5. This is produced by glancing of the colliding spheres so that the value of the relative normal velocity is too small.

The program to match the markers among the frames [26] is achieved using proximal uniformity constraint. In most of the cases, it works well but due to some limitation of the algorithm, occasionally it cannot get a correct result, especially when the markers are too close each other.

In summary, all the methods implemented here have performed very well and allowed us to correctly evaluate the coefficients of collision.

The experimental procedure developed here can be applied for different sphere sizes as well as collision between a sphere and a flat surface. The experimental apparatus needs to be modified when spheres of much smaller dimensions, e.g. 1 to 3 mm are used. The motion analysis technique, however, remains the same if a magnifying lens for the camera is used. Small differences in the value of some of these parameters produce significant difference in the results of simulations [27], therefore better experimental evaluation of such parameters is necessary in order to make dependable comparisons between the experiments and theory or simulations. The technique developed here should prove to be very useful in a variety of experiments in the granular flow.

APPENDIX A

SPECIFICATIONS OF THE KODAK SYSTEM

PROCESSOR

Controls

Menu-driven Keypad:	LCD display provides user access to all system functions. Includes six dedicated functions keys and ten-multi-function keys.
Power Switch:	Easily accessible.
Eject Switch:	Ejects tape cassette.

Operating Features:

Recording Technique:	Linear FM.
Recording Medium:	1/2" high density tape.
Tape Handling:	Cassette (700ft.)
Frame Rates:	Records at 30, 60, 125, 250, 500, 1000 full frames/sec. Up to 6000 pictures/sec.
Frame Formats:	1, 2, 3, 4 or 6 pictures/frame
Recording Time:	A minimum of 16 minutes at 30 fps and a minimum of 30 seconds at 1000 fps.
Normal Playback:	30 frames per second.
Single Step:	Displays one frame at a time, forward or reverse.
Jog:	Displays successive frames, forward or reverse, at a slow, continuous rate.
Fast Forward/Rewind:	Moves tape at 300 ips forward or reverse. This rate is faster than the highest recording speed.

BOT/EOT: Optically senses the beginning of tape and the end of tape to prevent overruns.

Search: Moves the tape to a given video frame.

Heads

Record & Playback: Two Microgap heads, each providing 19 channels-- 16 video, 2 timing, and 1 unsupported.

Erase: Permanent magnet.

Video Output

Compatible with: NTSC or PAL

Gamma Correction: Variable from 0.1 to 1.0

Grey Scale: 256 levels.

Size: 17"x22"x12 1/4".

Weight: Approximately 80 lbs.

Power: 110/220 VAC, 60/50 Hz, 8 amps/4 amps.

IMAGER

Control Keys: Live, Record & Stop

I/O Jacks: Video, Audio, & Remote Trigger.

Sensor: 192x240 pixel NMOS array.

Lens Mount: C-Mount, with electronic remote control capability for zoom, focus and exposure.

Tripod Mount: 1/4-20 and 3/8-16 with standard ANSI hole

	pattern.
Cables:	15 ft. standard.
Size:	Approximately 9"x4"x5"
Weight:	Approximately 5 lbs.
Power:	Derived from processor.

KEYPAD

Dedicated-function Keys

Live:	Displays live image on viewfinder and monitor.
Record:	Starts recording.
Stop:	Stops recording or playback and freezes the last image in frame store.
Replay:	Moves tape to first frame of most recent recording session and plays back at 30 fps.
Play:	Plays a recording in any selected playback mode.
Help:	Provides short cut paths through menu tree.

System Software Menu

System Setup:	Controls Imager selection, overlay format, position and size, frame rate and division factor, automatic lens functions and session numbers.
Move Tape:	Controls playback mode and event markers.
Video Display:	Enables reticle, gamma adjustment, interlaced video and saved image.
Environment:	Controls time and date.

APPENDIX B

PROXIMAL UNIFORMITY FUNCTION

The aim is to obtain a one to one correspondence F^k between points of the k th frame and the $(k + 1)$ th frame. It is assumed that objects in space move a small distance in a small amount of time and their motion is smooth or uniform. Therefore the location of a point from one frame to the next will be in the proximity of the previous location and the objects are assumed to follow a proximal uniform path. A *proximal uniformity function* d , was proposed which obeys the following criteria.

- In the two successive frames selected the speed of the objects does not change much.
- Direction in the two successive frames do not change much.
- There is very small displacement between any two successive frames.

The proximal uniformity function is defined as follows:

$$\delta (X_p^{k-1}, X_q^k, X_r^{k+1}) = \frac{\| \overline{X_p^{k-1}, X_q^k} - \overline{X_q^k, X_r^{k+1}} \|}{\sum_{x=1}^m \sum_{z=1}^m \| \overline{X_x^{k-1}, X_{\Phi^{k-1}(x)}^k} - \overline{X_{\Phi^{k-1}(x)}^k, X_z^{k+1}} \|}$$

$$+ \frac{\| \overline{X_q^k, X_r^{k+1}} \|}{\sum_{x=1}^m \sum_{z=1}^m \| \overline{X_{\Phi^{k-1}(x)}^k, X_z^{k+1}} \|}$$

where $1 \leq p, q, r \leq m$; $2 \leq k \leq m - 1$; $q = \Phi^{k-1}(p)$; $\overline{X_q^k, X_r^{k+1}}$ is the vector from point q in frame k to the point r in frame $k + 1$ and $\|X\|$ denotes the magnitude of the vector X . The i th point in the j th frame is denoted by the vector X_i^j in 2-D coordinates.

A major assumption being made with this proximal uniformity function is that F^1 , an initial correspondence, is known. Knowing the initial correspondence the algorithm correctly marks the trajectories of the points from frame to frame. The correspondence F^k is determined by minimizing the cost function $\sum \delta(X_p^{k-1}, X_q^k, X_r^{k+1})$.

Implementation of the Algorithm

For $k = 2$ to $n - 1$ do

- A matrix M ($m * m_{k+1}$) is constructed, where

$$M[i, j] = \delta(X_p^{k-1}, X_q^k, X_r^{k+1}), \text{ when } F^{k-1}(p) = i.$$

m_k points from the k th frame are along the rows and m_{k+1} points from $(k+1)$ th frame along the columns.
- If ($m_{k+1} < m_k$) then it is a case of occlusion so do
 - i. For $a = 1$ to m_{k+1} do
 - ★ The minimum element $[l_j, j]$ in each column j of M is identified.
 - ★ The priority matrix B is computed such that

$$B[l_j, j] = \sum_{i=1, i \neq l_j}^m M[i, j].$$
 - ★ The pair $[l_j, j]$ with the highest priority value $B[l_j, j]$ is selected and assignment $F^k(l_j) = j$ is made.
 - ★ Row l_j and column j are masked in M .

- ii. The points $m_k - m_{k+1}$ for which correspondence has not been found are identified. New feature points are created in frame $k+1$ for the missing points by extrapolating the correspondence from frame $k - 1$ to frame k .
- iii. Set $m_{k+1} = m$
else there is no occlusion
- iv. The row i and the column l_i in M is masked.

Points in the frame k that do not have corresponding points in the frame $k + 1$ are identified and new points are created corresponding to those missing points. If a point a in frame k with coordinates (x_a^k, y_a^k) which correspond to a point c in frame $k - 1$ with coordinates (x_c^{k-1}, y_c^{k-1}) does not have a corresponding point in the frame $k + 1$, then a point b with coordinates (x_b^{k+1}, y_b^{k+1}) is created to correspond to point a with the equations,

$$x_b^{k+1} = x_a^k + (x_a^k - x_c^{k-1})$$

$$y_b^{k+1} = y_a^k + (y_a^k - y_c^{k-1})$$

This extrapolation ensures smoothness in velocity, both in magnitude and direction.

APPENDIX C

THE PROCEDURE OF EVALUATING COLLISION PARAMETERS

Kodak Ektapro 1000 high speed video camera captures images of two sphere collision and later frames of image are transferred to data images in a PC486. In order to obtain the velocities of colliding spheres, six frame images are processed for each collision experiment, i.e., three pre-collision frames and three post-collision frames. The processing steps are in following.

For each image frame, use

- * *lap_of_gaussian* in Visilog package to get LoG image,
- * *pix2sub_3* or *pix2sub_4* to expand the LoG image to 9 or 16 times based on the subpixel technique,
- * *tovisi* in the PC to write the expanded image to the Visilog format,
- * *subtract* in Visilog package to transform the shifted image into the LoG image,
- * *zero_xing* in Visilog package to get the edge image,
- * *label* in Visilog package to obtain the labeled edge image,
- * *afc_1* to get the 2-D central coordinates of spheres and markers.

For all image frames, use

- * *match* to match markers among the six frames,
- * *ch_rel_coord* to transfer the marker center to relative coordinates original at the corresponding sphere center,
- * *coef* to calculate the pre and post-collision normal and surface velocities.

For all experiments, use linear regression method to obtain rolling and sliding lines and collision parameters.

REFERENCES

1. German, R. M. Particle Packing Characteristics, Metal Powder Industries Federation, Princeton, NJ (1989).
2. Walton, O. R. Numerical Simulation of Inelastic, Frictional Particle-Particle Interactions, in Particulate Two-Phase Flow (M. C. Roco edited) Chapter 25, Butterworth-Heinemann, Boston, (1992).
3. Rosato, A., Dave, R. and Fischer, I. "Measurement of Particle Properties in 3-D Collisions Using High-Speed Motion Analysis," *DOE Project Report* (1992).
4. Huerta, A. and Medioni, G. "Detection of Intensity Changes with Subpixel Accuracy Using Laplacian-Gaussian Masks," *IEEE Trans. on Pattern Analysis and Machine Intelligence* PAMI-8, 651-664 (1986)
5. Dave, R. and Bhaswan, K. "Adaptive Fuzzy c-Shells Clustering and Detection of Ellipses," *IEEE Trans. on Neural Networks* 3(5) (1992)
6. Rangarajan, K., and M. Shah. "Establishing Motion Correspondence." *CVIGP: Image Understanding* 54(1) (1991):56-73.
7. Sondergaard, R. and Chaney, K. "Measurements of Solid Spheres Bouncing off Flat Plates." *Journal of Applied Mechanics* 57(9) (1990):694-699.
8. Maw, N., Barber, J. and Fawcett, J. "The Role of Elastic Tangential Compliance in Oblique Impact" *Journal of Lubrication Technology* 103(1) (1981):74-80.
9. Drake, T. and Shreve, R. "High Speed Motion Pictures of Nearly Steady Uniform, Two-Dimensional, Inertial Flows of Granular Material," *J. Rheology* 30(5) (1986):981-993.
10. Nevatia, R. Machine Perception. Princeton Hall. (1982).
11. Rosenfeld, A., and A. C. Kak. Digital Picture Processing. Academic Press, Vol. 2 (1982).
12. Canny, J. "A Computational Approach to Edge Detection." *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8 (1986): 679-698.

13. Horn, B.K.P., Robot Vision, The MIT Press, Cambridge, Mass. (1986).
14. Marr, D. Vision, W. H. Freeman and Co., San Francisco, (1982).
15. Marr, D. and E. C. Hildreth. "A Theory of Edge Detection," *Phil. Trans. Roy. Soc. London B207*, (1980):187-217.
16. Hough, P.V.C. "Method and Means for Recognizing Complex Patterns," *U.S. Patent 3069654*, (1962).
17. Tsuji, S. and Matsumoto, M. "Detection of Ellipses by a Modified Hough Transform," *IEEE Trans. Comput.* 27 (1978):777-781.
18. Illingworth, J., and J. Kittler. "The adaptive Hough transform," *IEEE T-PAMI.* 5 (1987):690-697.
19. Dave, R. and Bhaswan, K. "Adaptive Fuzzy C-Shells Clustering," *Proc. of NAFIPS'91* (1991):195-199
20. Bezdek, J. C., Coray, C., Gunderson, R., and Watson, J. Detection and characterization of cluster substructure, *SIAM J. Appl. Math.* 40 (1981):339-372.
21. Dave, R. N. "Fuzzy Shell Clustering and Applications to circle Detection in Digital Images." *International J. of General Systems*, (1990).
22. Dave, R. "Generalized Fuzzy C-Shells Clustering and Detection of Circular and Elliptical Boundaries," *Pattern Recognition*, 25(7), pp 713-721, (1992).
23. Dave, R. "Boundary Detection through Fuzzy Clustering," Invited Paper, IEEE International Conference on Fuzzy Systems, San Diego, California, pp. 127-134, March 8-12, 1992.
24. Bhaswan, K. "Development of an Adaptive Fuzzy Shell Clustering Algorithm" *M. S. Thesis NJIT*, Newark, NJ (1991)
25. Volcy, J., "Kodak Ektapro 1000 Serial Interface for IBM PCs," *Technical report*, NJIT, Newark, NJ (1991).
26. Caesar, J. "Automatic Motion Analysis of Colliding Spheres", *M. S. Thesis NJIT*, Newark, NJ (1992)
27. Walton, O.R., Private communication (1993).
28. Louge, M., Private communication (1993).
29. Louge, M., U.S. DOE GFARO Spring Review Meeting, April, 1993.

30 Goldsmith, W., Impact, E. Arnold Pub., London (1960).