PREDICTING MISSING MARKER POSITIONS IN SIMULATED GAIT ANALYSIS SYSTEMS

JOSHUA SWAMIDAS

PREDICTING MISSING MARKER POSITIONS IN SIMULATED GAIT ANALYSIS SYSTEMS

BY

JOSHUA SWAMIDAS, B. ENG.

JANUARY 1997

A THESIS SUBMITTED TO THE SCHOOL OF GRADUATE
STUDIES IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING

FACULTY OF ENGINEERING AND APPLIED SCIENCE
MEMORIAL UNIVERSITY OF NEWFOUNDLAND
ST. JOHN'S, NEWFOUNDLAND

ST. JOHN'S NEWFOUNDLAND CANADA

ABSTRACT

Modem computer-aided vision motion systems provide a computerized and fully integrated tool kit for biomechanical measurement and analysis. These tools are useful for evaluation of problems, prescription of treatment and evaluation of such treatment. Many of these systems use reflective markers placed on key anatomical sites of the body to detect accurate three-dimensional spatial positions of the limbs being measured. While these systems ease automated data gathering, there are issues, such as the correspondence between an observed target and an established track, that require significant human intervention when markers disappear from view for short periods of time. When the system loses sight of a marker, it has no way of knowing where that marker will reappear and the track becomes broken or disjointed. Once the missing marker comes back into view, many current systems do not easily establish an association between the marker and its original track.

In this thesis a solution to the problem of making correspondence between markers and their track histories was designed and tested. This solution also provided the capability of predicting the path of markers when they were out of view of the cameras. To test the algorithm three different repetitive motions were tracked using the Flock of Birds measurement system.

The solution used a three-state Kalman filter to predict marker locations. The Kalman filter was coupled with constraints to determine matches between tracks and their corresponding marker positions. These constraints modelled a Region of Acceptance (ROA), distance from the center of the ROA to the last known position of a marker, and velocity matching.

The Kalman predictor algorithm, because it is linear in nature, was able to predict the motion accurately while there was no change in acceleration. However, the Kalman predictor, coupled with the constraints, was useful in predicting and matching markers over a longer (100-500% longer) missing interval than the test case.

To improve the prediction and matching capabilities of the Kalman predictor algorithm a physical motion model, that considers angular rotations at joints, was developed. The model is named the angular component model. This algorithm used an estimated or pre-measured motion model to check the location of the Kalman predictor. If the prediction did not match the model (within certain error bounds), it was corrected by the model algorithm using a calculation process that estimated the location of the marker based on its model. The addition of this algorithm to the Kalman prediction algorithm improved the prediction and matching capabilities. The matching worked well over the length of a 2 second gap (the longest used in testing) and the prediction of the marker path was excellent. The use of this model with the available tracking algorithms used in gait analysis will help in preventing the problem of disappearing markers in computer vision systems.

ACKNOWLEDGMENTS

I would like to thank my supervisors, Dr. Ray Gosine and Dr. John Molgaard. for the invaluable guidance they provided and work they put into helping me with my master's program. I would also like to thank them for the time they spent editing the thesis despite their busy schedules.

I greatly appreciate the timely help of Dr. Edmund Biden. Mr. Dov Goldvasser and the Biomedical Institute at the University of New Brunswick for providing my research topic and expert assistance in times of need. I am grateful to Mr. John Tucker for allowing me to use the facilities at the Biomedical laboratory, Memorial University of Newfoundland, including the Flock of Birds system. The computing and printing facilities provided by the Faculty of Engineering and Applied Science, Memorial University of Newfoundland, are also greatly appreciated.

I would especially like to thank Dr. Ray Gosine, the School of Graduate Studies at Memorial University of Newfoundland and the Faculty of Engineering and Applied Science for providing me with financial assistance during my master's program.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

ϵ_z^2, ϵ_z^2. ϵ_z^2	Error squared between predicted and observed tracks in x, y and z coordinates
δx	Change in X
δy	Change in Y
λ	Parametric variable
σ^2	Variance
θ_x, θ_y, θ_z	Refers to angles between axes (orientation of camera)
ϕ	refers to the helical angle
CCD	Charge-coupled device
EMG	Electromyogram
FBB	Fast Bird Bus
FOB	Flock of Birds
R	3x3 rotation matrix
RS232	Communication protocol for serial port on personal computer
SNR	Signal to Noise Ratio
t	3x1 translation matrix
T	4x4 transformation martix
ROA	Region of Acceptance

CHAPTER 1

INTRODUCTION

1.1 Background

Human walking is the process of locomotion in which the erect, moving body is mainly supported first by one leg and then the other. Knowledge of the mechanical and physiological mechanisms of human walking has been advanced by methods of gait analysis. Much of the recent gait analysis is being done in laboratories. Advances in electronics and use of new materials have improved the precision and reliability of detection and recording procedures. High speed film and recording techniques have enabled visualization of subtle events that were previously invisible to the examiner. Computer technology has provided the means to process more digital information at faster rates for lower costs. It is now possible to sample analog data at higher rates, use more sophisticated filtering and smoothing techniques and to sample more variables simultaneously.

Gait laboratories contain the necessary instrumentation to visualize and quantify the parameters used to describe walking. Modem computer-aided vision motion systems provide a computerized and fully integrated tool kit for biomechanical measurement and analysis for quantifying and analyzing normal and pathological patterns of locomotion, evaluation of problems, prescription of treatment and evaluation of such treatment. Most of these systems use reflective markers placed on key anatomical sites to detect accurate three-dimensional spatial positions of the limbs being measured. Three-dimensional motion models can be created using this positional data.

The study of gait and other body motions using these modern laboratories has obvious economic and patient comfort benefits associated with it. The motion analysis system can help physicians to plan studies for new treatments, to quantify how effective current treatments (before and after measurements) are thereby enabling patients to recover quickly and thus lowering the patient's hospital stay and all related costs (Ariel Web Page, 1996 - Appendix D). It also reintroduces a healthier individual into society. The motion analysis system can measure and analyze the intricate movements of athletes and provide coaches with visual aids to help in the improvement of athletic performance and decrease dangerous movements. This is of advantage to the athletes, their trainers, and the people they are competing for, especially when it can be accomplished with lower risk to the athletes (Ariel Web Page, 1996). The assessment of compensation claims would help reduce false claims by making a true assessment of damage. Applications in product testing and assessment before marketing will reduce potential claims against manufacturing companies.

1.2 Objectives of Study

In this thesis, the study and work was concentrated on the motion analysis component using a computer vision system. While these systems eased automated data gathering, there were issues, such as the correspondence between an observed target and an established track, that require significant human intervention when markers disappeared from view for short periods of time. When the system lost sight of a marker, it had no way of knowing where that marker would reappear and the track became broken or disjointed.

The first objective of this study was to provide a robust solution to the problem of establishing correspondence between markers and their track histories using data obtained from multi-camera computer vision systems, when the markers disappear from the view of the cameras. The second objective was to predict accurately the motion of the markers when they were not in the view of the cameras. To accomplish this, work was done to simplify and comprehend the mechanics of human motion from related earlier theoretical developments and modify them to be suitable for motion tracking and subsequent analysis. This included understanding the intricacies of walking, the laboratory equipment and software used to measure gait, and identification of possible topics for research.

1.3 Contents of Thesis

This thesis includes five chapters. Besides the introductory first chapter, Chapter 2 of the thesis provides a literature review concerning the state of the art of gait analysis and a critique of the techniques used and assumptions made. Chapter 3 outlines the necessary theory required for the study. Chapter 4 describes the experimental procedures used to test the different components of the software developed for the purpose and discuss the results provided by the tests. Finally, Chapter 5 highlights the salient findings of this study and makes recommendations for future work in this area.

CHAPTER 2

LITERATURE REVIEW

A modern, fully functional Gait Laboratory uses instrumentation to record data on the patient's walking pattern. A multi-camera video system, floor-contained pressure plates, and special equipment to determine activity within the muscular system are connected to an advanced computer system with vision capabilities to help document, visualize and analyze the body's motion.

As the patient walks in the laboratory, the instruments record and relay information to a computer. The data is then collected and analyzed to help physicians determine the type of surgery, braces, therapy or other corrective measures that will help the patient walk more effectively. Testing is mostly non-invasive (i.e., done without puncturing the skin), using cameras and specialized instruments. These instruments and procedures provide a complete analysis of the gait pattern, providing information that may be overlooked by the naked eye of even the most trained professional.

During the gait analysis study, the patient is asked to demonstrate his or her walking movements in the laboratory. The patient will be examined with and without different braces or assistive devices (such as crutches or walkers), computers will record all necessary data while the patient completes the walk on a specially designed floor plan. The actual walking portion of the study takes less than 20 minutes.

As was stated in the introduction, the problem of camera-aided vision systems losing sight of markers due to various reasons such as body rotations and obstructions is a serious one. Mah (Personal Correspondence (Appendix A), Mah, 1996) stated that there was no complete way to rectify the problem, since any method which was used involved constructing estimates of missing data. It was suggested that, especially for highly pathological gait, clever methods to replace missing data were not worth the effort and often made matters worse. It was also suggested that the only defense against this problem was to eliminate, but in case that was not possible, to reduce the seriousness of the problem by reducing the size of the gaps. The best way to reduce the size of the gaps is to add extra cameras to provide redundancy in the cameras viewing the markers. Unfortunately, in many cases, no matter how many cameras are added, a few gaps will occur. Also, the addition of extra cameras requires significant increases in costs of both hardware and software. So software solutions are needed for filling gaps (predicting marker motion and connecting disjointed markers), especially when using fewer cameras.

This chapter describes various techniques and instruments used for measuring motion and will summarize state of the art commercial systems. It briefly outlines some algorithms used for three-dimensional reconstruction from camera images (2D), tracking markers in 3D space, predicting marker motion, and models of motion used to analyze gait.

2.1 Measurement of Motion

Kinematic analysis of gait requires measurement of displacements of the body segments during the walking cycle. The measurement techniques can be subdivided into direct measurement techniques and imaging measurement techniques. Examples of direct measurement systems include goniometers, accelerometers, resistive grid walkways and foot switches. The direct measurement techniques are adequate for many applications but can be cumbersome to use, especially with they have to be attached to the patient's body and while they are inexpensive compared to the motion analysis systems, the information produced is often lacking in detail (O'Mallay, 1993, Biden et al. 1990).

Electrogoniometry: A goniometer is a device attached to the subject and measures the angle orientation between two limb segments. The standard device uses a potentiometer that converts changes in rotary motion to a proportional electrical output. They are inexpensive, and the output voltage may be processed in real-time (O'Mailay, 1993). One problem is that the device is difficult to fit to the body segment and constrains the movement of the patient's body. The output is unreliable and is relative angular data and not absolute values. Another problem is tracking more than one rotation per joint. There are some modified versions of these that allow for motion to occur outside the plane of measurement. Strain gauges and fibre optics are used as alternatives for potentiometers. Strain gauges are sensitive to deformation and can be designed to record motion in two planes simultaneously without additional attachments. Despite the problems, it is a useful tool for providing joint displacement data quickly (Smidt, 1990).

Accelerometry: Accelerometers on limbs can be used to measure normal and tangential components of acceleration. The output is available in real time but the acceleration signal is relative to the position of the accelerometer on the limb. The wiring of the accelerometer can also constrain the movement of the subject (O'Mallay, 1993, Smidt, 1990).

Resistive Grid Pathway and Foot Switches: The resistive grid is placed in the floor of the walkway upon which the patient will tread. The foot switches are placed on the soles of the person's feet. The output is available in real time but it only gives information when the foot is in contact with the ground; therefore many joint movements cannot be measured.

Imaging techniques overcome the difficulties encountered by direct measurement techniques. These techniques include photographic, cinefilm and video or other electronic techniques. These techniques are used to calculate the position and orientation of each body segment to reconstruct the movements that took place. Also, measurements can be made in two or three dimensions.

Photographs: Photographic records of gait can be obtained using interrupted light to identify points in the body. There are many different interrupted light techniques. Two of the more common ones are: (i) a camera with an open lens and a rotating slotted shutter. lighted by flood lights, which exposes the film briefly at specific intervals to record the position of the marker; and (ii) use of a strobe instead of a flood light and rotating shutter. Both these techniques can demonstrate serial changes in limb position at precise intervals (Sutherland et al, 1988).

Measurements on stride dimensions can be made using time and distance parameters of successive gait cycles and sagittal plane motion can be visualized or measured manually from the photographs. The advantage of photographs is that they are relatively inexpensive and cause minimal encumbrance to the subject. The disadvantages are that they are not useful for tracking in three dimensions and converting to digital data for further processing is time consuming and nearly impossible because there are several images of the walking sequence in one photograph.

Film and Video Techniques: These two techniques can capture a separate image of the subject during each successive time segment of the walking cycle. Also, these techniques can increase the amount of samples, which in turn enables the examiner to view and reproduce movements that occur quickly. However, it also increases data processing. Normal walking alone requires about 50 to 60 pictures per second while running requires up to 200 pictures per second (Sutherland et al, 1988, O'Mallay, 1993, Smidt, 1990).

Both of these techniques track markers placed in strategic locations of the subject's frame. For three-dimensional reconstruction the markers must be seen by at least two cameras. Multiple cameras, four to six, are used to record different views of the walking subject. Positioning of the markers is critical because serious errors can occur due to improper marker placement. While the use of markers is one of the better techniques to track motion there are some problems. Problems occur when trying to attach surface markers over certain joints such as the hips. This can be taken care of by using other markers at known anatomical locations to determine the motion of another. Moreover, knee motion is difficult to track because the axis of rotation changes during the movement. Another problem is movement of skin and underlying soft tissue. Some examples of this problem are: first, if a marker is placed on the knee when the subject is sitting, the marker will move forward when the subject stands; and second, thigh markers may oscillate as the subject walks because of the movement of the underlying tissue. It should be noted that the marker placement must be reliable to be useful for repeated testing.

An advantage of video techniques over film techniques is the *automated 3D systems*. These systems eliminate the need for processing film and since the signal is already electronic and projected on a 2D grid, the X-Y coordinates are already known. Conventional systems can scan between 50-60 cycles per second. There are two types of marker schemes used in these systems. First, passive markers are made up of a highly reflective material and are tracked by shining an infrared or bright light source on them. These markers often can be seen as an array of markers that are attached to track the motion of limb segments. Color coded prisms around the world. The significant companies include VICON (VICON Web Page, 1996), Ariel Dynamics, Inc. (Ariel Web Page, 1996), Peak Performance Analysis (Peak Performance Web Page, 1996), Qualysis (Personal Correspondence, Johnsson, 1996), Elite (BTS Web Page, 1996), Motion Analysis Corporation (Personal Correspondence, Greaves 1996) and OPTOTRAK (Northern Digital Web Page, 1996). These companies provide hardware consisting of either off-the-shelf products or their own proprietary systems. Each company includes software to operate the computer and follow bio-mechanical procedures to analyze the data and present the results in useful, acceptable formats. This section will outline some equipment and software techniques available.

VICON Motion Systems (Oxford, England and Santa Fe Springs, CA, USA) is the largest commercial company for the motion analysis products. Its top of the line system is the VICON System 370. It consists of up to seven synchronised CCD cameras and has several peripheral devices such as force platforms and electromyogram (EMG) measurement devices that can be attached to it. These devices are connected to a Pentium workstation running Windows. VICON Clinical Manager, VICON Reporter and VICON BodyBuilder are three high-end software packages that are available for analyzing the data obtained from the hardware. The price tag for this system is around $200,000 (US) for the basic 370 and it goes higher as extras are added to the system.

The VICON Bodybuilder software, introduced to the market during the past year (1996) can automatically fill gaps, interpolate and smooth data using complex tracking and matching algorithms (Personal Correspondence, Roudanez, 1996). The software allows the user to choose from a variety of algorithms to either automatically or manually join disjointed tracks and fill in the gaps. The software uses kinematic equations and/or some physical modeling of the body to connect disjointed marker tracks. It also allows for the user to edit manually and model the data.

The Ariel Performance Analysis System (APAS), designed and manufactured by Ariel Dynamics, Inc. (Englewood, CA, USA) is a modular system using off-the-shelf computers, A/D boards, VCRs, any additional computer peripherals and cameras that sample with speeds of 60 Hz to 10,000 Hz. The software available provides the tools necessary for analyzing the data. This system has been used extensively in the sports medicine field for task improvement, post-injury assessment, risks assessment and disability evaluations. It is a good system for analysis of non-complicated motion. It is the cheapest system available with costs ranging between $20,000 and $30,0000 (US).

Ariel handles the joining of small gaps in many ways and two were mentioned in a personal correspondence (Personal Correspondence, Wise, 1996): (i) using a complex search algorithm along with simple linear extrapolation based on a history of data and (ii) using smoothing functions, with user intervention to direct the algorithms, to interpolate over spikes and gaps in data. However, the larger the gaps the greater the error associated with these two approaches. They also allow the user to manually fill gaps and connect disjointed tracks.

Peak Performance Technologies, Inc. (Englewood, CO, USA) supplies video and computer motion measurement systems. These systems save images from the cameras onto a video tape that are then analyzed by its Peak Motus software (Windows-based). Data from devices such as force platforms and EMG can be sampled and displayed in synchronization with video. The cost of this system is more than $150,000 (US).

Peak Performance Technologies reiterated that the best solution for eliminating gaps in marker tracks is by adding more cameras (Personal Correspondence, Sturkol, 1996). Data shows that with a four-camera system and a standardized 15 point Helen Hayes marker set only about six or seven markers can be reconstructed without filling any gaps. The others may have up to 50% of their image space missing. Using five cameras, this percentage is reduced to about 30% and with six cameras it is reduced to about 10% or less in all but the most difficult cases. Their latest software, released in the mid-1996, does handle filling in the gaps to a certain extent. Their latest extrapolation and interpolation algorithms are based on human motion and are used to predict not only which path segment may join to another, but also, what direction the point is moving in the camera's image space so that it can be self correcting. Their software allows the user to connect the gaps manually by connecting points, semiautomatically by allowing the user to run different algorithms, or automatically by allowing the user to set several tracking parameters. While much has been done using more cameras and brute force mathematical extrapolation and interpolation to fill missing data accurately, there is still plenty of room for improvement, especially in using fewer cameras with software or hardware that is smart enough to fill bigger and bigger gaps in the data. The algorithms for the software developed in this thesis, while different from the ones listed here, have followed a similar approach to solving the missing gap problem.

Qualysis (Sweden & USA) manufactures the MacReflex Motion Measurement Systems for the Macintosh or PC computers. This system uses passive reflectors with digital cameras to track motion and the data can be interfaced with force plate and EMG measurement devices. It is designed as a biomechanics research system, adaptable to many experimental protocols including gait analysis. The software provided is used for data analysis of various kinds. This system costs about $75,000 US.

To track markers, the Qualysis software extrapolates (Personal Correspondence, Johnsson, 1996) the approximate next position of the markers. This extrapolation helps the tracker to look in the right places when looking for the next positions of the markers. In situations where the addition of extra cameras will not help solve the markers out-of-view-of-cameras problems, the operator can run the tracking software manually or run the tracking separately for specific segments of motion stored in a file. Since the tracking can be run both forwards and backwards in the file, a point with missing markers can be approached and tracked from both directions. A user of this software (Personal Correspondence, Zhang, 1996) commented that when the discontinuities were too large (>½-⅔ seconds) the filling in of gaps did not work well.

The ELITE system by Bioengineering Technology & Systems (Milano, Italy) provides up to an 8-camera system, eight channels EMG with optional telemetry and piezoresistive footswitches, force platforms and general purpose software for data acquisition. This system tracks markers that reflect infrared light. The software provided is based on three different lines: kinematics, kinetics and electomyography. The primary products include: ELITEplus (A three-dimensional motion package), ELICLINIC (Clinical gait analysis), TELEMG (Dynamic electromyography with no limitations on subject motion), GAITemg (Identification of muscle activity in walking), PcVect (Analysis of ground reaction forces), etc. The price tag for this system is greater than $200,000 (US). The Elite software (Personal Correspondence, Huiliger, 1996) provides interpolation to handle relatively short gaps (<¼ (10-15 samples)).

Motion Analysis Corporation's (Santa Rosa, CA, USA) top system has up to six CCD cameras and software that runs on SGI, SUN or SUN/PC systems. HiRes software features include a data collection package, on-screen 2D video viewing, raw data access, 3D track editor and a full tracking system. This software captures the data from the cameras and converts it to a form that is usable by the analysis packages for clinical applications programs such as KinTrak (allows users to create 2D and 3D kinetic and kinematic assessments, animations, and EMG reports) and OrthoTrack (uses standard and/or user customized kinematic and kinetic reports, graphs, and charts to assess gait). The top of the line system can track a full body that consists of 17 segments (with 51 markers). This system costs more than $200,000 (US).

According to the founder and Senior Vice President of Motion Analysis Corporation, only pathological gait motion is difficult to track (Personal Correspondence, Greaves, 1996). Their best solution is to add more cameras. Their software requires a minimum of four or a recommended six cameras for a full gait analysis system, unless specialized marker sets, such as placing markers only in the front of the body, are used. They have software, the Track Mender, which looks at the continuity of 3D path segments and automatically joins them if they are no more than a small number (< ¼ seconds) of frames apart. Another technique used by their software is to allow the user to manually spline across the gaps.

Northern Digital, Inc. (Waterloo, Ontario, Canada) designs and manufactures the OPTOTRAK 3D position Measurement systems. It uses off-the-shelf hardware and infrared light-emitting diodes. It's biomedical applications include motor control research, rehabilitation research and therapy, dentistry, neurosurgery, and gait analysis. It's industrial applications involve robotics, aeronautics, reverse engineering and virtual reality.

2.3 Reconstruction of Marker Positions

The algorithms in this section assume that the data is acquired by a multi-camera image system. The system described in this section is based on standard commercially available hardware and it tracks reflective markers placed on key anatomical sites. The digitizing of images and recognition of the markers is not discussed as it is not in the scope of this thesis. It is assumed that the image coordinates of the marker positions are already obtained.

Reconstruction of three-dimensional marker positions. in a base reference coordinate system. is needed since a digitized camera image only contains two-dimensional information relative to the camera position and orientation (Gonzales and Woods. 1992). There are many different algorithms available to determine three-dimensional coordinates from two-dimensional images; some of these are based on iterative methods and others on analytical methods. There are three main methods that are in use: triangulation; ray shooting (tracing); and reconstruction using the epipolar line geometric constraint.

2.3.1 Triangulation

Triangulation is a standard technique that uses similar triangles to determine distances and sizes of objects. This technique can be used in a one camera system or a multi-camera system. The one and two camera systems will be described in this subsection.

Monocular Imaging (Ayache, 1991, Ballard and Brown, 1982, Kasturi, 1991): This technique uses *point projection* that is the fundamental model for the transformation wrought by our eyes, cameras or many other imaging devices. Figure 2.1 shows a pinhole camera. The image results from projecting the image through a single point onto an image plane.

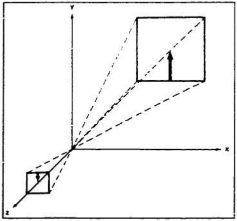

Figure 2.1 A pinhole camera model

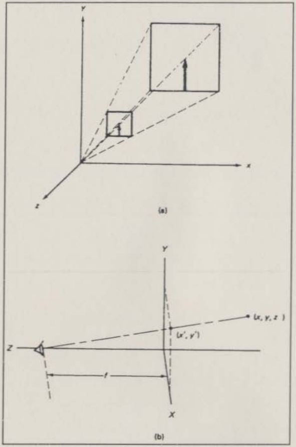

Figure 2.2 (a) Camera model equivalent to Fig 2.1;
(b) graphical definition of terms

where *(x', y')* and *(x", y")* are the retinal coordinates of the world point imaged through each eye. The *baseline* of the binocular system is *2d (let d1=d2=d)*. Thus

$$\begin{aligned}(f-z)\,x' &= (x-d)\,f \\ (f-z)\,x'' &= (x+d)\,f\end{aligned} \tag{2.4}$$

and subtraction of these two equations gives

$$(f-z)(x''-x) = 2df \tag{2.5}$$

or

$$z = f - \frac{2df}{x''-x'} \tag{2.6}$$

Therefore if the points can be matched to determine the disparity *(x"-x')*, and the baseline and focal length are also known, then the z coordinate can be calculated.

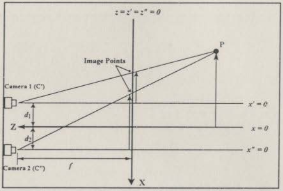

Figure 2.3 A Binocular Imaging System

An important step in extracting depth information from stereo images is the matching of points for disparity calculations. There are several methods to do this but they will not be discussed here since they are out of the scope of this study.

2.3.2 Ray Shooting

This technique involves the shooting of rays from a point behind the image plane (the distance depending on the focal length) through corresponding pixels of the image plane. As seen in Figure 2.4 two rays are projected from the two corresponding points in the two image planes, respectively. These projected rays, depending on the quality of the system, may or may not intersect but will have a point where they are closest to each other. Therefore, it is assumed that the location of the three-dimensional point occurs at this shortest distance between the rays. The lines are defined parametrically (these are rays that go to infinity but certain range restrictions can be placed on them),

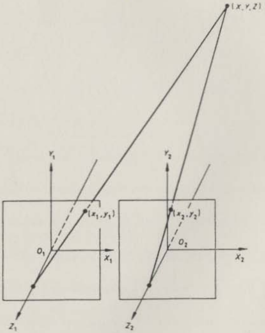

Figure 2.4 Stereo imaging system

$$\begin{aligned} x_1 &= Au + B \\ y_1 &= Cu + D \\ z_1 &= f \end{aligned} \tag{2.7}$$

and,

$$\begin{aligned} x_2 &= Ev + F \\ y_2 &= Gv + H \\ z_2 &= f \end{aligned} \tag{2.8}$$

where the parameters u and v vary from 0 to 1 along the length of a normalized finite line segment. The distance between these lines for any values of u and v is given by

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{2.9}$$

and the values of u and v that minimize d will determine the (approximate) location of the three-dimensional point. The disadvantage of this technique is that it requires an iterative technique to find the solution.

2.3.3 Epipolar Constraints

This technique uses geometric constraints for matching and 3D reconstruction of stereoscopic images. The technique provides a closed form solution to the set of 10 equations and 9 variables that will be discussed in greater detail in the next chapter. It also allows for the arbitrary placement of cameras and uses the cameras in which the point falls within the camera's line of sight. This technique is general and can be easily modified to use as many cameras as possible (Ayache, 1991, Kanazawa and Kanatani, 1995).

Of the above three methods (triangulation, ray shooting and epipolar constraint), the epipolar constraint method was chosen because it provided a closed form solution for the reconstruction of the three-dimensional marker positions. The camera orientation and position can be arbitrary since they are taken into account by the equations and this technique can be extended to use as many cameras as are available.

2.3.4 Concepts of Uncertainty

Uncertainty of measurement is generally caused by the equipment. Such equipment includes the cameras and the processing units. The first type of uncertainty comes from the digitization process, which is well known in signal processing. The second type of error comes from image processing, while extracting the image features. The third and last type of uncertainty is generated by calibration errors in setting up all instrumentation components (i.e., the orientation of cameras with respect to each other and the base), and the uncertainties in camera parameters such as focal distances and distortions due to the pin hole model.

The sums of these uncertainties (in 2D) are modeled by an isotropic error around each pixel, and given as a number of pixels (Bonnin and Zavidovique,1991). This type of assumption in 3D space is not possible, as is shown in Figure 2.5 in translating the 2D uncertainty to 3D space. Translating the 2D uncertainty to 3D space

$$\begin{aligned} Z_{max} &= \frac{Df}{\delta x(dx-2\delta n)} \\ Z_{min} &= \frac{Df}{\delta x(dx+2\delta n)} \end{aligned} \tag{2.10}$$

with f: focal length

D: Distance between the cameras

δx and δy: size in X and Y of a pixel on a focal plane

dx: the disparity (=nlx-nrx)

nlx and nrx are X pixel coordinates in left and right images

It can be shown that the error in Z is proportional to the square of the distance in Z-axis.

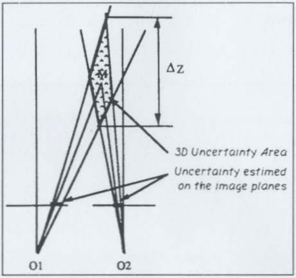

Figure 2.5 3D space uncertainty

2.4 Software/Hardware Tracking of Markers

In the camera vision system, once three-dimensional reconstruction is completed, the marker positions need to be matched to their corresponding tracks. The track consists of the history of a marker's motion through 3D space and is used to produce the traces of motion which are used in the analysis of the motion of the limbs to which the markers are attached.

Many different procedures are available to track markers. This is a well-developed field of computer vision, and some of the latest methods are listed here. These methods can all be used in gait analysis.

2.4.1 Statistical Data Association Techniques

These techniques have been studied in radar imagery for target tracking and have only recently been introduced into the field of computer vision. These technique can be used to track a sequence of images over short time intervals; if the time interval is small and the object velocity is constrained by physical laws, the interframe displacements of the objects are bounded. Also, since we can assume that the objects move smoothly, motion coherence can be used to predict the occurrence of markers in the future, which considerably reduces the search space (Zhang, 1994). The technique can be used to track markers despite the following problems:

- A previously unseen object may partially or totally come into view
- A moving object in the current field of vision may move partially or totally out of it in the subsequent frames
- A moving object may partially or totally be occluded by the background or by other objects.
- Some markers that should be present are not seen, due to failure of the feature extraction (or reconstruction) process.

This technique uses the Kalman filter (described in Subsection 2.4.5) to perform the prediction in the prediction-matching algorithm. It then uses Mahalanobis (squared) distance to decide which scene marker matches the predicted makers. The calculation of Mahalanobis distance involves matrix inversion and is therefore computationally expensive especially with larger numbers of markers. Also, in tracking of moving markers over larger intervals (>½ seconds), the algorithms and heuristics used were very complicated and required a greater number of parameters eg. the probabilities of detection and termination of a token and the probability of appearance of a new token. After some initial studies, it was decided that many of the matching and predicting could be done using simpler non-statistical techniques.

2.4.2 Extrapolation of Incomplete Marker Tracks by Lower Rank Approximation

This technique (Muijtjens et al, 1993) is used to track the motion of objects like the deformation of the heart, it's walls and the like. In experimental situations markers may not be detected due to occlusion or lack of contrast. As a result the continuous marker track is observed in separate parts, which cannot often be directly identified as corresponding to one marker. The extractions are obtained by iteratively fitting a lower rank matrix to a set of noisy, incomplete marker tracks. This technique is computationally heavy since the mathematics in creating the lower ranked matrix and the iteration are quite involved.

2.4.3 Precise 3D Motion Analysis System for Real-time Applications

This motion analysis system was developed using strategically placed markers, one or more and motion history and matches the observed marker positions to its predictions.

The filter, also known as the "optimum recursive predictor," gives a linear "best" estimate of the k+nth value (n=1,2,3 . . . ∞), where k is the current value. The model is a first-order autoregressive process with random noise, W(k) (Bozic, 1979),

$$x(k) = Fx(k-1) + W(k-1) \qquad (2.11)$$

and the observation includes additive white noise, V(k),

$$y(k) = Cx(k) + V(k) \qquad (2.12)$$

The equations of the vector Kalman predictor (Bozic, 1979) are:

$$\textit{Predictor equation:}\ \hat{x}(k+1|k)=F\hat{x}(k|k-1) - G(k)[y(k) - C^{T}\hat{x}(k|k-$$

$$\textit{Predictor gain:}\ G(k) = FP(k|k-)C^{T}[CP(k|k-1)C^{T} - R(k)]^{-1}(2.13)$$

$$\textit{Predictor mean square error:}\ P(k-1|k) = [F - G(k)C]P(k|k-1)'F^{T} - Q(k)$$

The update of filter variables is linear and is a function of the predicted variables. However, the prediction of a marker position is not necessarily linear because the prediction is based on the motion of the marker which, in this case, includes velocity and acceleration. The function of this filter is to predict the future state of the system given an estimate of the current state. The Kalman algorithm will be described in greater detail in the next section.

The filter used in this experiment is a decoupled, Kalman predictor with three states; position, velocity and acceleration. The position data is observed by the vision system and the velocity and acceleration components are derived from the position data. This technique was chosen because of many reasons: (i) ease of implementation; (ii) availability of many sources of information; (iii) inputs to the filter are easily measured or calculated, (iv) filter constraints and precalculations can be changed to suit the conditions of the experiment (discussed in Chapter 3) and most importantly (v) while the filter's function is in the tracking of markers, it can be easily modified to predict the motion of a marker several time steps into the future and thereby avoiding a separate algorithm for motion prediction (section 2.5). Another indirect reason for choosing this technique was that with some modifications it could be converted into a non-linear filter which is briefly described in section 2.4.6.

2.4.6 Extended Kalman Filter Tracking

This filter is similar to the linearized Kalman filter. The difference is that the extended Kalman filter uses measurable characteristics or associations to modify the predictions (Brown and Hwang, 1992, Wu et al, 1989) of the linearized Kalman filter. The extended Kalman filter will use measurable quantities such as the static distance between two points, associations such as three markers to make a unique segment in space, etc. (things that can be identified without knowing the nature of the motion), to modify its prediction. Ideally, this would be the filter to use since it performs better in nonlinear applications of motion. However, in gait analysis making these associations between markers is difficult unless special shapes of markers or special marker sets are used. Since a generalized technique that could work with any marker set was wanted this technique has not been chosen.

2.5 The Prediction of Motion

One of the aspects of this research was the assessment of the problem of missing markers and possible solutions by discussion with members of the Biomechanics list server (BIOMECH-L). The personal correspondents agreed that missing markers in motion analysis were a problem (to varying degrees of seriousness) and no one had a good general solution. It was suggested that the best solution was to make sure the problem did not occur. This could be achieved by increasing the number of cameras viewing the tracking region, using non-camera methods of tracking such as magnetic tracking (Personal Correspondence, Cao, 1996), carefully choosing a marker set that would stay in view despite obstructions, rotations, and other problems causing occlusions, and using special assistive devices that reduce interference with the cameras (Personal Correspondence, Hulliger, 1996). Nevertheless, in cases where the missing marker problem could not be eliminated software solutions were suggested.

Various suggestions were discussed for the prediction of marker motion or the filling-in-of-gaps. These include:

1) Joining short gaps using spline functions (cubic or quintic) (Personal correspondence, Cao, 1996). It may not be very accurate over large gaps. Human involvement is necessary to determine the start and end points for the spline functions.

2) For short isolated gaps, using a straight line or polynomial algorithm works well (Personal Correspondence, Mah, 1996). This is done manually after computer

tracking and matching are completed.

3) Curve fitting algorithms are fine for small gap sizes (10 frames) (Personal correspondence, Keezel, 1996). This is done manually after computer tracking and matching are completed.

4) For longer gaps using a proxy marker to provide an offset-recording (Personal Correspondence, Mah, 1996, Hulliger, 1996) of the marker that consistently disappears from view. This proxy marker can then be used to infer the position of the missing marker automatically.

5) For longer gaps, extrapolating along a straight line between two markers (Personal Correspondence, Mah, 1996). For example, if there is trouble in viewing an ankle marker, put two markers on the lower leg, so that the ankle is further along the straight line drawn between the two markers.

6) If the general forms of a specific movement are known, then a tracking algorithm can be written for this specific problem (Personal Correspondence, Zhang, 1996).

7) Three markers uniquely define a segment's location and orientation in space. If a 4th marker is used, the redundancy could be used to calculate a missing marker (Personal Correspondence, Olree, 1996).

8) If the distances between three markers on a segment are known and the locations of two of the markers are known, the location of the third marker can be constrained to a small area. This might be used in conjunction with an interpolative procedure to give a better estimate of location (Personal Correspondence, Olree, 1996).

9) If the data is cyclical, it may be possible to "guess" where a reasonable location of the

marker might be based on where it was one period before or after the instant (Personal Correspondence, Olree, 1996). However, in gait analysis cases where an assistive device obstructs the camera view, they tend to happen in the same portion of each stride (Personal Correspondence, Geil, 1996).

10) For path matching, sometimes paths can be easily identified by some unique characteristic of their location (Personal Correspondence, Olree, 1996) such as a marker on the upper part of the body would have greater values in the height coordinates.

11) Using direction and velocity to help identify possible path matches (Personal Correspondence, Olree, 1996). This will work well if the gaps are short and there are no sudden reversals in direction of the marker. This technique was used as the reference test to compare the Kalman filter and the angle model techniques.

12) Again for path matching, using a marker path reconstruction technique to reconstruct a path and match it to an observed path (Personal Correspondence, Olree, 1996).

Many of these suggestions were good only for filling short gaps of missing data and worked for only limited cases, the longer the gaps the less accurate the prediction. During the discussion, it was realized that all prediction (gap-filling) methods used for calculating marker motion were only as good as the equations and algorithms being used and that there was no real general solution to this problem. The Kalman filter approach had not been used in this area previously and since it was a well proven algorithm in other tracking and predicting applications it was decided to test it in tracking markers placed on the human anatomy. This general filter predicts target motion for several time steps (camera frames) into the future. It can be used with any type of marker motion, and calculates the prediction based on the history of the marker motion. The initialization variables and the constraints used are developed in detail in Chapter 3.

It was found, during testing, that since it is a linear filter, it cannot predict the complex gait motion accurately (Chapters 3 and 4 detail the capabilities and limitations of using the Kalman filter as a predictor of motion). Because of this, it was decided to couple the filter prediction with a physical model of the gait. This modeling is described in greater detail in the next subsection.

2.6 Modeling of Gait

The Kalman filter, used in this work, predicts the motion of the markers accurately, without additional constraints, for short gaps (less than 15 frames or approximately ½ second) of missing data. Since it is a linear filter, it cannot accurately predict the complex motion involved in gait, especially over longer gaps of missing data. Therefore, a physical model of gait is used to check the predictions and make necessary corrections to the predictions. The physical model provides a general framework for a model of motion. Given a location in the model's cycle and some information at that point, one can calculate the actual motion at that point in time. Some methods of modeling and analyzing gait will be discussed in this subsection.

2.6.1 Joint Angle Motion Models

The joint angle motion models involve calculating three-dimensional relative motion between rigid bodies. These models are used primarily to analyze gait. Using positional data, the three-dimensional angular motion at a particular joint is calculated by first assigning embedded coordinate systems to both the proximal and the distal segments (Kadaba etal, 1990, Mah et al, 1994, Ramakrishnan and Kadaba, 1991, Sutherland et al, 1988). Euler angles or helical (screw) axis definitions are then used to compute the relative positions between the embedded coordinate systems.

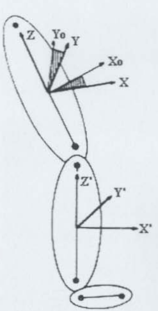

Figure 2.6 Schematic diagram of thigh (X, Y, Z) and shank (X', Y', Z') embedded axis used for the estimation of knee joint angles and helical parameters (Ramakrishnan and Kadaba, 1991)

Both Euler and helical motion definitions are sensitive to the orientation of the three axes about which the rotations are assumed to take place in sequence. It is difficult, if not impossible, to ensure that the defined axes coincide with the base reference axes of rotation. This introduces a range of errors in joint motion (Ramakrishnan and Kadaba1991).

The following description of the embedded coordinate system is for the knee of the right-sided limb (Figure 2.6) and can easily be extended to other joints. The distal to proximal direction of the thigh and the shank segment are defined as the Z- and the Z'-axes, respectively. The lateral to medial direction of the thigh that is perpendicular to the Z-axis is defined as the Y-axis while the lateral to medial direction of the shank that is perpendicular to the Z-axis is defined as the Y'-axis. As a consequence, the posterior to anterior direction, perpendicular to both the Y- and Z-axis is the X axis of the thigh. Similarly, the posterior to anterior direction, perpendicular to both Y'- and Z'-axes is the X'-axis of the shank.

Two sets of unit vectors, **I**, **J**, **K**, and **I'**, **J'**, **K'** are assigned to the coordinate directions X, Y, Z and X', Y', Z', respectively. From these two sets of unit vectors, the rotational transformation matrix, **R**, for any relative arbitrary orientation of two limbs is calculated as:

$$\boldsymbol{R} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} = \begin{bmatrix} I' \cdot I & I' \cdot J & I' \cdot K \\ J' \cdot I & J' \cdot J & J' \cdot K \\ K' \cdot I & K' \cdot J & K' \cdot K \end{bmatrix} \tag{2.14}$$

To extract the rotational information from this transformation matrix, the transformation matrix for the Euler model is (Ramakrishnan and Kadaba,1991):

$$\boldsymbol{R_e} = \begin{bmatrix} C1*C3+S1*S2*S3 & C2*S3 & -S1*C3+C1*S2*S3 \\ -C1*S3+S1*S2*C3 & C2*C3 & S1*S3+C1*S2*C3 \\ S1*C2 & -S2 & C1*C2 \end{bmatrix} \tag{2.15}$$

where C1 refers to the cosine of angle θ, and S1 refers to the sine of angle,θ. Similar definitions apply to other terms, and the helical motion model is given by:

$$\boldsymbol{R_h} = \begin{bmatrix} U_x*U_x-U_x*U_x*C+C & U_x*U_y-U_x*U_y*C-U_z*S & U_x*U_z-U_x*U_z*C+U_y*S \\ U_x*U_y-U_x*U_y*C+U_z*S & U_y*U_y-U_y*U_y*C+C & U_y*U_z-U_y*U_z*C-U_x*S \\ U_x*U_z-U_x*U_z*C-U_y*S & U_y*U_z-U_y*U_z*C+U_x*S & U_z*U_z-U_z*U_z*C+C \end{bmatrix} \tag{2.16}$$

where U_x, U_y, and U_z represent the direction cosines of a unit vector, U, along the helical axis; ϕ refers to the helical angle; C refers to the cosine of ϕ; and S refers to the sine of ϕ.

These models are similar to and form the basis for the model defined and used in this thesis. The angle models for the analysis, Figure 2.7, are calculated by taking the mean of several cycles of gait. This normalized gait cycle is then compared against the population norm.

While this model is good for visual analysis, considerable information is lost in the conversion from positional data to the angle data. The positional data, the length of the limbs, and the motion of the limbs cannot be calculated in the backward direction because these angles are relative angles. Therefore, it is not useful for this application.

2.6.2 Component Angle Model

The component angle model was designed specifically for comparing the Kalman prediction against a physical model of gait. It is similar to the Joint Angle model with a few important differences. Instead of using relative positions between joints, this model calculates three component angles (X, Y, Z) with respect to a base reference frame (of the vision system). Instead of using actual limbs, the model uses the marker positions to define rigid bodies. Two adjacent markers define one rigid body. So the model contains the lengths of the limbs, the connectivity of the limbs (adjacent markers) and the component angles. From this information, the locations of all the markers can be determined if the location of one marker is known. This model will be discussed in detail in the next chapter.

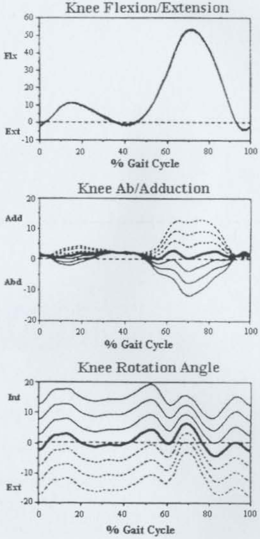

Figure 2.7 Variations in knee joint angles (Euler Model) of a representative normal subject. The outermost curves correspond to a perturbation of 15° (Ramakrishnan and Kadaba, 1991)

CHAPTER 3

ANALYSIS OF TRACKING GAIT WITH MISSING MARKER DATA

This chapter describes the theory used to develop the software for the tracking of markers in gait analysis. The software matches image points from different cameras and reconstructs their three dimensional marker locations (world coordinates). These reconstructed points are then matched to their corresponding tracks of marker movement using the Kalman filter. When location data is missing, the Kalman filter predicts the motion of the markers in the gap. This prediction is then checked against the physical model that has been implemented.

Figure 3.1 shows the overview of the relationship between the algorithms that are discussed in this chapter. There are two inputs to the software. The first input block depicts the marker track (2D) data files as obtained from camera image planes and the initialization files consisting of the orientations and locations of the simulated cameras (to be explained in next section) and their focal lengths. The second input block is the expected motion model which is calculated or measured before starting the tracking software. The matching of the 2D tracks from two cameras occurs in the image matching algorithm. After the matching has been determined, the image points are passed to the 3D reconstruction algorithm where the 3D world coordinate marker locations are calculated. After these marker locations are calculated, they are passed to the Kalman predictor/physical model algorithm for matching

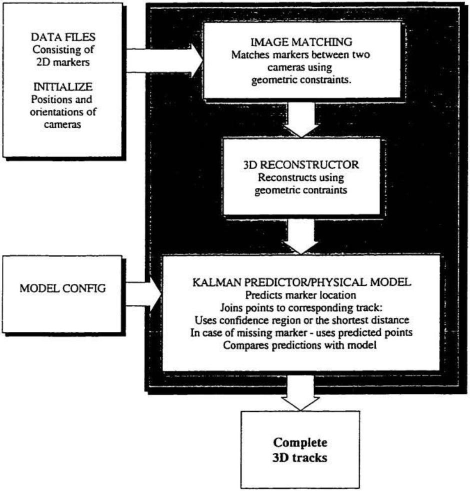

Figure 3.1 Overview of all blocks of software described in section

markers to their corresponding tracks and in the case of missing markers, predicting their locations. The output of the software are the complete 3D tracks of motion for all the markers present.

The first section on epipolar matching presents the theory on epipolar lines and then shows the development of the image matching algorithm. The second section describes the 3D reconstruction algorithm used. The third section describes the Kalman predictor technique and shows the development of the various components of the predictor equations. The fourth section provides the reasoning behind the matching of three-dimensional marker locations to their corresponding tracks. The fifth section describes the development of the physical model. The sixth and final section describes how the algorithms in the first five sections fit together to form a complete analysis system.

At the outset of the project, it was proposed that the data to test these routines would be obtained from a camera vision system. Due to delays in obtaining the vision system, an alternate methodology was used to obtain the data required for the verification of the theory. The data for the experiment comes from the *Flock of Birds*^{TM} (FOB) measurement system and is explained further in Chapter 4. The output marker locations from the system are in 3D world coordinates (3D measurement system). To simulate tracking by a camera vision system, these files are converted to the image planes (2D) of the simulated cameras using simple linear translation, rotation and projection. The procedure for simulating the data is explained in greater detail in Chapter 5.

3.1 Epipolar Matching

One component of gait analysis involves the computation of a three-dimensional representation of the leg trajectory when walking from one point to another. This procedure involves computing a 3D representation of motion from several images taken simultaneously from different viewpoints.

The method used in this thesis consists of modeling the process for the formation of each camera image by a linear transformation in projective coordinates, followed by a computation of the parameters of the transformation. These are subsequently used to specify geometric constraints and to determine the spatial position of a marker from multiple images (Ayache, 1991).

This technique is an intuitive approach to the matching and reconstruction problem. It also allows for the arbitrary placement of multiple cameras and uses the cameras in which the point falls in the camera's line of sight.

3.1.1 Image Modeling

Each camera is modeled by its optical center, **C**, and its image plane, Q (Figure 3.2). The cameras are calibrated and modeled as a standard pinhole (Zhang, 1995). A point $\mathbf{P}_1$(x, y, z) in the world coordinate space projects onto the camera's image plane at the image point I_1(x, y) and similarly $\mathbf{P}_2$(x, y, z) to I_2(x, y). Point I_1 is the intersection of line P_1C with plane

Q. The relation between each point in 3D space and it's corresponding point in the image plane is described by a transformation (Ayache, 1991, Zhang, 1995). The transformation is modeled by a transformation matrix **T** in projective coordinates, $\mathbf{I}^*$. The relationship is given by

$$I^* = \begin{pmatrix} U \\ V \\ S \end{pmatrix} = T \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \tag{3.1}$$

where **T** is a 4x4 matrix, generally called the *perspective matrix* of the camera and U, V, and S are the transformed 3D coordinates in the camera's coordinate system. **T** has the simple form

$$T = [R_{wc}, t_{wc}] \tag{3.2}$$

where $\mathbf{R}_{wc}$ is the 3x3 the rotation matrix and $\mathbf{t}_{wc}$ is the 3x1 translation matrix that describes the transformation from the world coordinate frame, in which the 3D points $\mathbf{P}_1$ and $\mathbf{P}_2$ are

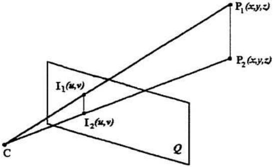

Figure 3.2 The pinhole camera model

Each match of an image point *I* with an associated scene point *P* gives two sets of linear equations from the matrix **T**, i.e.,

$$\begin{aligned} P't_1 + t_{14} - u(P't_3 + 1) &= 0 \\ P't_2 + t_{24} - v(P't_3 + 1) &= 0 \end{aligned} \qquad (3.5)$$

where t_{ij} if the *(i,j)* element of **T**, $\mathbf{t}_i$ is the vector composed of the first three elements of the row *i* of **T** (Ayache 1991) and **P** is the point in the world coordinate system.

3.1.3 Epipolar Lines

The epipolar lines are calculated based on relations between multiple cameras. In Figure 3.3 point $\mathbf{I}_1$ in image 1, is the match for point $\mathbf{I}_2$ in image 2. Point $\mathbf{I}_2$, in camera 2, is located on a straight line of image 2 determined completely by $\mathbf{I}_1$, and is called the *epipolar line* associated with $\mathbf{I}_1$. This epipolar line in image 2 is the projection of the line defined by the set of points **P**, whose image corresponds to $\mathbf{I}_1$ that can be defined as line $\mathbf{PI}_1$.

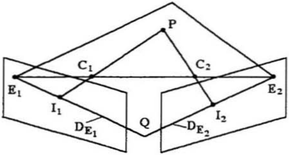

Figure 3.3 The geometry of binocular stereo vision

A plane, *Q*, can be defined by $\mathbf{I}_1$, $\mathbf{C}_1$, and $\mathbf{C}_2$ (three points). This plane intersects the image planes along two straight lines $\mathbf{D}_{E1}$ and $\mathbf{D}_{E2}$. Any point $\mathbf{I}_1$ on the epipolar line, $\mathbf{D}_{E1}$, has its potential matches on $\mathbf{D}_{E2}$ and vice versa. It can be seen that any epipolar line of image plane 2 is the image of a line passing through $\mathbf{C}_1$, i.e., all epipolar lines in image plane 2 will pass through $\mathbf{E}_2$ that is the projection of the optical center of camera 1. $\mathbf{E}_2$ is called the epipole of image 2 and similarly $\mathbf{E}_1$, the image of $\mathbf{C}_2$, in image plane 1, is the epipole of image 1.

3.1.4 Epipolar Matching

Given the point $\mathbf{I}_1$ in the first image, it's corresponding point **P** in 3D space must be on the line $\mathbf{C}_1\mathbf{P}_\infty$ (Figure 3.4) passing through $\mathbf{I}_1$, where $\mathbf{P}_\infty$ is a point at infinity (∞). Figure 3.4 shows the epipolar geometry with the optical center of the camera behind the image plane instead of between the image plane and the object as in Figure 3.3. This inversion of the optical center does not change any equations and is well known. Assume that the coordinate frame for camera $\mathbf{C}_1$ is the reference frame. If, for simplicity sake, the world coordinate frame was allowed to coincide with this reference frame, and $\mathbf{I}_1 = [u,v]^T$, then point **P** can be represented as

$$P = \lambda\ \bar{I}_1 = \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \qquad (3.6)$$

where u, v are the image coordinates, and λ a variable between $[0,\infty)$. This is the parametric representation of the line $\mathbf{C}_1\mathbf{P}_\infty$ (Zhang, 1995). If the world coordinate system does not coincide then a second transformation would transform **P** to the correct values in 3D space. The projection of the line $\mathbf{C}_1\mathbf{P}_\infty$ on camera $\mathbf{C}_2$ is a line, the epipolar line, denoted by $\mathbf{D}_{E2}$, on which the corresponding point in the second image of point $\mathbf{I}_2$ must lie.

The point **P** can be transformed, denoted as **P'**, to the coordinate frame of the second camera by

$$P' = R * P + t = \lambda R \tilde{I}_1 + t, \quad \lambda \in (0, \infty) \tag{3.7}$$

The epipolar line can be defined by two points. The first point can be obtained by projecting **P** with $\lambda = x_1$ and the second with $\lambda = {}_2x$. These two points can be transformed to the coordinate frame of the second camera $\mathbf{P}_1'$ and $\mathbf{P}_2'$. These points can be projected onto the image plane of camera $\mathbf{C}_2$. The point that lies on this line is the match to $\mathbf{I}_1$ in the image plane of camera $\mathbf{C}_1$.

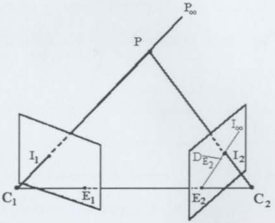

Figure 3.4 Epipolar geometry with image plane between the point and optical centre of the cameras

3.1.5 Epipolar Matching Algorithm

The algorithm's input consists of the two cameras (camera to world) transformation matrices, their corresponding inverse transformation matrices and the image points from the two image planes. This algorithm will be run for each sampled data point of the trial run. The image points will consist of all the 2D marker coordinates observed by each camera. A check is made to determine if both camera images contain the same number of points. If they do not contain the same number of points, a flag is set to pass back only the matched points.

The flowchart for the matching algorithm is shown in Figure 3.5. To start the matching process one camera is chosen as the base reference camera. The choice of the camera has been soft-coded as the first camera transformation to be passed into the algorithm. The coordinate system of this camera is assumed to be the world coordinate system. The rotation and translation between the cameras 1 and 2 are calculated using the data from the two camera (camera coordinates to world coordinates) transformation matrices. Then each image point seen in camera 1 is taken in turn.. First, the image point from Camera 1 is used to set up the parametric equations of (3.6). Two values for λ are used (soft-coded) to calculate two sets of 3D coordinates. Second, these 3D coordinates, used as reference points, are transformed to the coordinate system of the second camera. Third, the 3D reference point coordinates are projected back into the image plane (2D) of the second camera by dividing the x and y components by the z component. Fourth, the equation of a line through the two reference points is determined. Fifth, a search is done using the nearest vertical distance from all image points on that line. The one closest to the line is chosen as the match. Checks are made to make sure there are image points available in both camera images if the reduced points flag is set.

Generally geometric constraints such as epipolar lines are not sufficient to determine stereoscopic matches. However, in this particular application, the low number of markers and the positioning of cameras to avoid problems along with some linear conditions provide good matching capability. Tests in Section 4 will show that this algorithm works very well in a noiseless environment but it does not work as well in a noisy environment. To make improvements in the future, additional constraints can be added to this matching technique.

3.2 Three-dimensional Reconstruction

Once the matches of image points are determined, the reconstruction of the marker locations in a world coordinate frame can begin. Knowledge of $\mathbf{T}_1$ and $\mathbf{T}_2$, from cameras 1 and 2 respectively, are sufficient to compute the three coordinates of any point P, given its two images $\mathbf{I}_1$ and $\mathbf{I}_2$.

The system defined in Equation 3.5, for cameras 1 and 2, can be rewritten to give a new system of four equations in three unknown coordinates (x, y, z) of **P** (Ayache, 1991)

$$\begin{aligned}
(t_1^1 - u_1 t_3^1)' P + t_{14}^1 - u_1 t_{34}^1 &= 0 \\
(t_2^1 - v_1 t_3^1)' P + t_{24}^1 - v_1 t_{34}^1 &= 0 \\
(t_1^2 - u_2 t_3^2)' P + t_{14}^2 - u_2 t_{34}^2 &= 0 \\
(t_2^2 - v_2 t_3^2)' P + t_{24}^2 - v_2 t_{34}^2 &= 0
\end{aligned} \tag{3.8}$$

where the indices *i* and *j* of t_i^j refer to the i^{th} row of the transformation matrix of camera *j*.

In theory, these equations are related since $\mathbf{I}_1$ and $\mathbf{I}_2$ are chosen in the same epipolar plane. However, due to numerical imprecision and the absence of an objective criterion in the choice of the equation to eliminate variables, it appears appropriate to solve the whole system by least squares method outlined below (Ayache, 1991). This approach extends naturally to reconstruction based on an arbitrary number of cameras. For n cameras, set

$$\boldsymbol{A}\boldsymbol{a} = \boldsymbol{b} \tag{3.9}$$

with $\mathbf{a}$=(x, y, z)$'$ and

$$\boldsymbol{A} = \begin{pmatrix} \boldsymbol{A}_1 \\ \cdot \\ \cdot \\ \boldsymbol{A}_n \end{pmatrix} \quad and \quad \boldsymbol{b} = \begin{pmatrix} \boldsymbol{b}_1 \\ \cdot \\ \cdot \\ \boldsymbol{b}_n \end{pmatrix} \tag{3.10}$$

for which

$$\boldsymbol{A}_i = \begin{pmatrix} (t_1^i - u_i t_3^i)' \\ (t_2^i - v_i t_3^i)' \end{pmatrix} \quad and \quad \boldsymbol{b}_i = \begin{pmatrix} u_i t_{34}^i - t_{14}^i \\ v_i t_{34} - t_{24}^i \end{pmatrix} \tag{3.11}$$

The least squares solution is then given by

$$\boldsymbol{a} = (\boldsymbol{A}'\boldsymbol{A})^{-1}\boldsymbol{A}'\boldsymbol{b} \tag{3.12}$$

provided $\mathbf{A}'\mathbf{A}$ is invertible. $\boldsymbol{a}$ is the reconstructed 3D marker position. One needs to store k (k=2,3) matrices $\mathbf{T}_k$ and solve a linear system of $2k$ equations in three unknowns.

3.2.1 Reconstruction Algorithm

Figure 3.6 shows the flowchart of the algorithm used in reconstruction. The inputs to the algorithm are the inverse transformation matrices (world to camera) and the matched image points matrix from both cameras. The matrices **A** and **b** are determined based on equations

(3.10) and (3.11). The least squares approach, equation (3.12), is used for the reconstruction of the 3D marker location, **a**. This loop is continued until all the image points have been reconstructed.

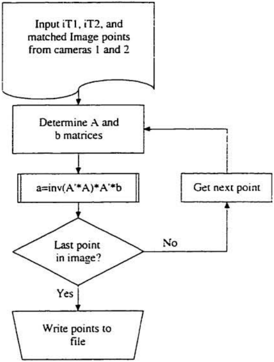
Input iT1, iT2, and matched Image points from cameras 1 and 2
Determine A and b matrices
a=inv(A'*A)*A'*b
Get next point
Last point in image?
No
Yes
Write points to file

Figure 3.6 Flowchart of reconstruction algorithm

3.3 Kalman Prediction

This section will describe the algorithm that can be used to track the markers attached to the legs and to predict their positions when the markers are hidden from the cameras (thus making three-dimensional reconstruction impossible) (Bozic. 1979).

Many tracking algorithms have been developed (Ramachandran. 1987) that make use of position measurements. These have been mainly used for tracking aeroplanes. The most common ones used in application are the various types of Kalman filters that are linear filters/predictors over a set time interval.

It has been shown (Ramachandran. 1987. Ramachandran et al. 1993) that increasing the order of the Kalman filter improves the accuracy of the prediction. The increasing of the order of the filter involves using measurements of velocity and acceleration as inputs for the filter. Unfortunately equipment for measuring velocity and acceleration is more specialized and therefore more expensive but there are methods to estimate these inputs. In this study, the position measurements were used to estimate velocity and acceleration. A three-state Kalman tracking filter that predicts position based on range, velocity and acceleration was developed. The equations of dynamics (Ramachandran et al. 1993) for an object moving at constant acceleration with a uniform time interval T (at each position coordinate) are

$$x''_{n+1} = x''_n + a_n T \tag{3.13}$$

$$x_{n+1} = x_n + x'_n T + x''_n \frac{T^2}{2} + a_n \frac{T^3}{6} \tag{3.14}$$

$$x'_{n+1} = x'_n + x''_n T + a_n \frac{T^2}{2} \tag{3.15}$$

3.3.1 Kalman Predictor

The algorithm developed is a one-step predictor. It predicts the location of a marker one time step into the future. It can be easily modified to predict *m* steps into the future (Bozic, 1979).

The signal model is a first-order autoregressive process (Figure 3.7)

$$\overline{x}(k+1) = \overline{F}\,\overline{x}(k) + \overline{w}(k) \tag{3.16}$$

where **w**(k) is zero mean white noise.

$$\overline{y}(k) = \overline{C}\,\overline{x}(k) + \overline{v}(k) \tag{3.17}$$

where **v**(k) is an independent additive white noise with zero-mean and unknown variance σ^2.

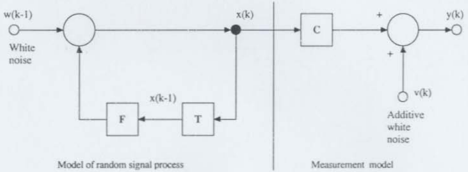

Figure 3.7 Random signal process and measurement model

The optimum one step prediction is shown in Figure 3.8. The summarized set of equations are shown below. The development of these equations is not dealt within this thesis since it is a standard filter and the information can be found easily (Ayache, 1991). The predictor equations are:

giving

$$Q = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ U_x^2(k) & U_y^2(k) & U_z^2(k) \end{bmatrix} \tag{3.25}$$

From equation (3.17) we define the terms as following

$$\begin{bmatrix} P_{xobs} & P_{yobs} & P_{zobs} \\ V_{xobs} & V_{yobs} & V_{zobs} \\ A_{xobs} & A_{yobs} & A_{zobs} \end{bmatrix}(k) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_{xact} & P_{yact} & P_{zact} \\ V_{xact} & V_{yact} & V_{zact} \\ A_{xact} & A_{yact} & A_{zact} \end{bmatrix}(k) - \begin{bmatrix} \sigma_{px} & \sigma_{py} & \sigma_{pz} \\ \sigma_{vx} & \sigma_{vy} & \sigma_{vz} \\ \sigma_{ax} & \sigma_{ay} & \sigma_{az} \end{bmatrix}(k) \tag{3.26}$$

Also

$$R = E\left[V(k)\,V'(k)\right] = \begin{bmatrix} \sigma_{px} & \sigma_{py} & \sigma_{pz} \\ \sigma_{vx} & \sigma_{vy} & \sigma_{vz} \\ \sigma_{ax} & \sigma_{ay} & \sigma_{az} \end{bmatrix} \begin{bmatrix} \sigma_{px} & \sigma_{vx} & \sigma_{ax} \\ \sigma_{py} & \sigma_{vy} & \sigma_{ay} \\ \sigma_{pz} & \sigma_{vz} & \sigma_{az} \end{bmatrix} \tag{3.27}$$

and assuming that $\sigma_{px} * \sigma_{px} = \sigma_{px}^2$ and $\sigma_{px} * \sigma_{py} = 0$, i.e., no cross-correlation, then

$$R = \begin{bmatrix} \sigma_{px}^2 - \sigma_{py}^2 + \sigma_{pz}^2 & 0 & 0 \\ 0 & \sigma_{vx}^2 - \sigma_{vy}^2 + \sigma_{vz}^2 & 0 \\ 0 & 0 & \sigma_{ax}^2 - \sigma_{ay}^2 - \sigma_{az}^2 \end{bmatrix} \tag{3.28}$$

To start the Kalman predictor, calculations the gain matrix **G**(k) has to be initialized. For this purpose the error covariance matrix **P**(k) has to be specified. An *ad hoc* technique used in Ayache (1991) is based on the first few measurements of data to calculate an initial value for **P**(k). In this application since the velocity and acceleration are calculated the first three measurements of position at times k=1, 2, and 3 are used. From these three sets of

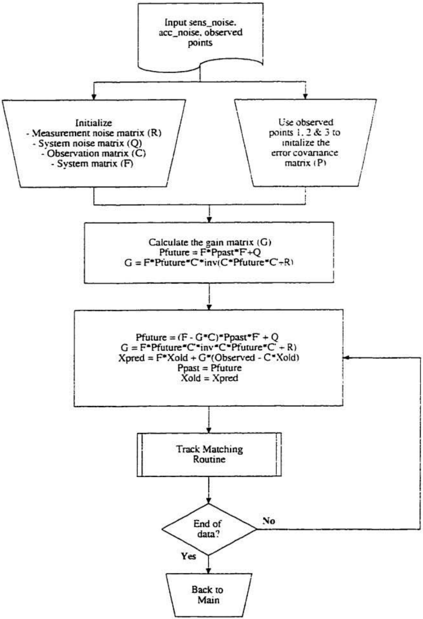
Input sens_noise, acc_noise, observed points
Initialize
- Measurement noise matrix (R)
- System noise matrix (Q)
- Observation matrix (C)
- System matrix (F)
Use observed points 1, 2 & 3 to initalize the error covariance matrix (P)
Calculate the gain matrix (G)
Pfuture = F*Ppast*F'+Q
G = F*Pfuture*C'*inv(C*Pfuture*C'+R)
Pfuture = (F - G*C)*Ppast*F' + Q
G = F*Pfuture*C'*inv*C*Pfuture*C' + R)
Xpred = F*Xold + G*(Observed - C*Xold)
Ppast = Pfuture
Xold = Xpred
Track Matching Routine
End of data?
No
Yes
Back to Main

Figure 3.9 Flowchart of Kalman prediction routine

3.4.1 Single Step Tracking

The Kalman predictor approach used predicts future positions of the marker based on past track data. At time *t* the predictor predicts the marker positions at time *t+1*. At time *t+1* the reconstructed (observed) 3D marker location is matched to the predicted marker locations and thus the markers are connected to their corresponding tracks.

This one-to-one correspondence occurs only under ideal conditions. Uncertainty due to white noise, errors in measurement, numerical imprecision and the linear predictive nature of the Kalman predictor cause the predicted locations to rarely match with the observed locations. Therefore, some conditions need to be used to match the prediction with the observation. To this end, a region of uncertainty is established around each predicted point. This is called the *region of acceptance* (ROA) (Figure 3.10).

The predicted marker has an upper and lower bound of uncertainty associated with each coordinate. *ex, ey, and ez* are the uncertainty lines for coordinate x, y, and z. They combine to form a cube volume around the predicted marker.

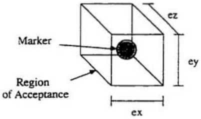

Figure 3.10 Region of acceptance

The upper and lower bounds of uncertainty are defined as the location of the predicted marker plus and minus the variance of measurement. This technique of matching markers to tracks works well with one step prediction.

3.4.2 Multi-Step Tracking

One of the problems with the camera vision computer-aided-systems is that due to obstruction, occlusion, or some other reason the system loses track of one or more markers for a period of time. Once this happens the system does not know where that marker will reappear and the track becomes broken or disjointed.

The Kalman predictor is used to predict the marker locations until a match for that location is found. Figure 3.11 shows the prediction of a segment of a marker track between times t = k and t = k+n. The motion of the marker between k and k+n is unknown and is depicted in the figure by two possible paths (solid lines) that it might have traveled. The ROA at t = k+n is larger than at t = k. This is because the region of acceptance grows larger the further into the future the prediction is continued; the greater the region of acceptance, the greater the possibility of error.

The growth of the region of acceptance is denoted by the dashed lines. In this application, the region of acceptance has two factors influencing its growth. The first is the error due to the measurement devices. This error is estimated at the beginning of the trial run and added to the x, y and z component for every consecutive prediction step. The second factor influencing the growth of the region of acceptance is the last calculated velocity vector. The percentage of the possible distance traveled in the time interval using this velocity vector is calculated and also added to the x, y, and z components of the region of acceptance.

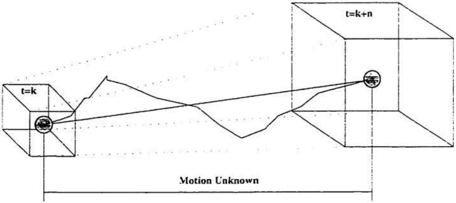

Figure 3.11 Difference in region of acceptance at different times of prediction

Since the Kalman filter is a linear predictor it works well in predicting motion that is simple and that has very little change in acceleration over a period of time. In gait. however. the motion is complex. With complex motion, even after as few as 20 time steps into the future the predictor gives inaccurate predictions. Therefore to predict gait motion for any length of time and match the predictions to their corresponding tracks requires additional constraints. These additional constraints include the distance of observed marker to the center of any region of acceptance and the velocity profile of the observed marker matched to that of the predicted track. If one track and one marker are left unmatched after the constraints are checked then these markers are considered to be matched in a process of elimination. These additional constraints can be used. in theory, as long as the regions of acceptance do not overlap.

The algorithm for matching is shown in Figure 3.12. First, the straightforward matching is done. Any marker that is found in a ROA is then attached to the associated track. If there are two markers found in a ROA then the ROA is removed from matching consideration until all the other markers are matched. If there are any unmatched markers left. then additional constraints will be used. It should be noted that there may be more ROAs than observed markers because one or more markers may not have been in view of the cameras. Second, the distance constraint is used. The Euler distances, which are the square root of the sum of squares of the difference between the remaining marker locations and the predicted location, are calculated. The markers are then matched using minimum distances. If there are markers unmatched, the velocity constraint is used. Finally, by a process of elimination, if the above constraints found matches then no markers should be left. If one is left then the final match is made to the only remaining ROA using distance as the deciding criterion. If additional markers are left after this step, then they are discarded.

Even with these additional constraints, however, if the Kalman predictor is allowed to predict far enough into the future (greater than 1 second, i.e. 30 time steps), the regions of acceptance for different markers will overlap and thus the prediction will become useless for the matching of markers to tracks. To solve this problem a physical model has been devised to help improve the prediction and matching process. This will be described in the following section.

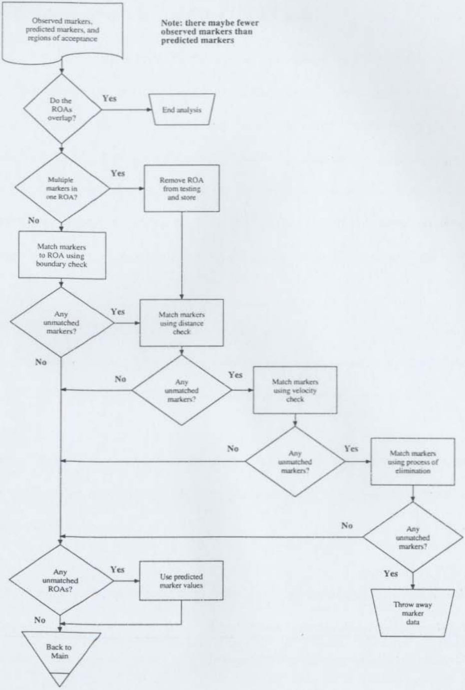
Observed markers, predicted markers, and regions of acceptance
Note: there maybe fewer observed markers than predicted markers
Do the ROAs overlap?
Yes
End analysis
Multiple markers in one ROA?
Yes
Remove ROA from testing and store
No
Match markers to ROA using boundary check
Any unmatched markers?
Yes
Match markers using distance check
No
No
Any unmatched markers?
Yes
Match markers using velocity check
No
Any unmatched markers?
Yes
Match markers using process of elimination
No
Any unmatched markers?
Yes
Throw away marker data
Any unmatched ROAs?
Yes
Use predicted marker values
No
Back to Main

Figure 3.12 Flowchart of matching process

3.5 Component Angle Model of Gait

This model is a representation of the angles that adjacent body segments. and their corresponding lengths. make with world coordinate system. These angles are broken into their x. y. and z components and therefore given the name *component angle model*. It was designed specifically for the purpose of comparing the Kalman prediction against a physical model of gait. Component angles are calculated from the 3D Kalman prediction data which can then be compared against the expected or measured model. This subsection will describe the model and the algorithm used to calculate the model from 3D coordinate data.

3.5.1 Discussion of the Model

The model uses 3D marker position data to define rigid bodies: two adjacent markers define one rigid body. Figure 3.13 shows a three marker system with two rigid bodies.

Each rigid body forms three component angles (X. Y. Z) with respect to the world coordinate system. The model consists of the lengths of the rigid bodies, the connectivity of the markers and the component angles.

Using this model, the locations of all the markers can be determined if the location of one or more markers is known. The motion model has been developed by considering the three dimensional motion of the rigid bodies (Kadaba et al,1990, Ramakrishnan and Kadaba, 1991, Sutherland et al, 1988).

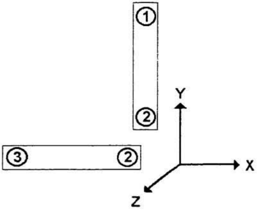

Figure 3.13 Rigid body model

The connectivity matrix for the example shown in Figure 3.13 is

$$CONNECT = \begin{bmatrix} 2 & 1 \\ 3 & 2 \end{bmatrix} \tag{3.33}$$

This matrix shows that marker 2 forms a rigid body with marker 1 and marker 3 forms one with marker 2. The component angles are easily determined by using the normalized vectors, **nR**$_1$ and **nR**$_2$, of the rigid bodies which are defined by

$$\begin{aligned} nR_1(k) &= \frac{P_2 - P_1}{|P_2 - P_1|}(k) \\ nR_2(k) &= \frac{P_3 - P_2}{|P_3 - P_2|}(k) \end{aligned} \tag{3.34}$$

where **P**$_1$, **P**$_2$, and **P**$_3$ are the three dimensional marker locations at time step k. These normalized vectors are used to compute the component x, y, and z angles that the rigid body forms with the world coordinate system as shown in the set of equations (3.35)

$$
\begin{aligned}
\theta_{x1}(k) &= acos([1\ 0\ 0]\cdot \boldsymbol{nR}_1{}^T(k))\\
\theta_{y1}(k) &= acos([0\ 1\ 0]\cdot \boldsymbol{nR}_1{}^T(k))\\
\theta_{z1}(k) &= acos([0\ 0\ 1]\cdot \boldsymbol{nR}_1{}^T(k))\\
&\textit{and}\\
\theta_{x2}(k) &= acos([1\ 0\ 0]\cdot \boldsymbol{nR}_2{}^T(k))\\
\theta_{y2}(k) &= acos([0\ 1\ 0]\cdot \boldsymbol{nR}_2{}^T(k))\\
\theta_{z2}(k) &= acos([0\ 0\ 1]\cdot \boldsymbol{nR}_2{}^T(k))
\end{aligned}
\tag{3.35}
$$

These angles provide a motion history of the rigid body movements. Since the rigid body motion observed here is cyclic in nature, the length of one cycle is determined and the mean of several cycles is taken to form a percentage cycle model of motion shown in Figure 3.14. The figure shows the component angle cycles for one of the rigid bodies using data from one of the test runs. The test runs will be explained in the next chapter.

3.5.2 Development of the Algorithm

The physical model is used to compare predicted marker locations with a set model of motion and provide the data necessary to move the predictor back on track. The flowchart for this procedure is shown in Figure 3.15. The only time that the comparison to the physical model is made is if there are one or more markers missing from view. If one or more markers are missing and therefore their locations predicted, the physical model checking routine is called. This routine uses the marker positions, which have already been matched to their corresponding tracks, to calculate the component angles of the rigid bodies that they form. A check is then made against the expected angle values at that particular location in the cycle. If the angles are within a small range around the model component angles, the

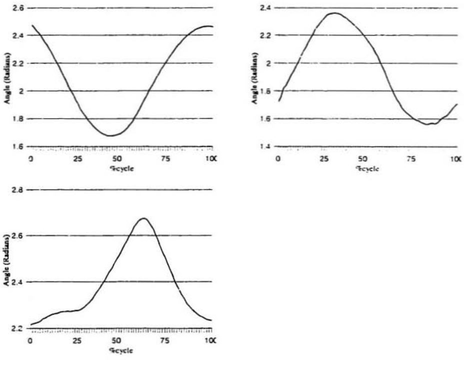

Figure 3.14 x (top left), y (top right), and z (bottom) graphs of the component angle cycle

comparison passes; if not it fails.

As long as there is one observed marker, the approximate locations of all other markers can be calculated. The connectivity matrix, along with the component angles per rigid body, and their lengths, can be used to calculate the positions of the markers. For example, if P_2 were missing either P_1 or P_3 can be used along with their corresponding angles and the length to determine the location of P_2.

One important constraint is that the software needs to know in which part of the cycle the motion is occurring. Only then can a proper comparison can be made. As mentioned previously, the angle model based on the predicted and observed markers is calculated and compared with the component angles at that location in the cycle. If the angles are within a small range around the model angle (set to ±5% of the gait cycle), the comparison passes; if not it fails. If it fails then the locations of the predicted markers are calculated using the known locations of the observed markers, the length of the rigid bodies and the expected angles at that point in the cycle.

CHAPTER 4

EXPERIMENTAL PROCEDURE, RESULTS, AND DISCUSSION

This chapter describes the procedure used for testing the theory discussed in the previous chapter, where the motion of a person (described in terms of her/his position coordinates) is tracked by two (or more) cameras placed at strategic locations. Markers are located on key anatomical sites on his or her body. Figure 4.1 shows the data flow of the various algorithms described in the last section. Software procedures, written in the Matlab^{TM} programmers' environment, were developed to test the theory outlined in Chapter 3.

At the outset of the project it was expected that the data to test these routines would be obtained from a camera vision system. Due to delays in obtaining the vision system, an alternate methodology was used to obtain the data required for the verification of the theory. The data for the experiment came from the FOB measurement system. This system is composed of three electromagnetic sensors and a base magnetic field generator. The sensors return position and orientation data to a data collection system. Section 4.1 describes the

FOB system in greater detail. Since the range of measurement of the FOBs is a maximum of three feet from the field generator, it does not allow for tracking of gait. As a substitute for periodic motion, markers were placed on a subject's arm and cyclic/periodic motions were performed.

The data from the sensors were already in 3D coordinates with respect to the base and could be input directly to the Kalman prediction routine. However, to simulate a camera vision system, the data were transformed to two simulated camera image planes which were assumed to be the input to the system. Image data were removed from sections of the record, as required, to simulate periods when a marker was not seen by a camera. Figure 4.1 shows the generalized flow of the data from one stage (subroutine) of analysis to the next. These stages, depicted by the rectangular blocks, are discussed in detail in this chapter. The 3D position data (marker tracks) from the FOB system were passed through a moving average filter to remove some of the high frequency components from the data. These data (3D) were then converted to the camera image planes (2D) using simulated camera transformation matrices. The 2D tracks of data were then matched and reconstructed to retrieve the 3D coordinates of the markers. The 3D coordinates were entered into the Kalman predictor routine to be matched to their corresponding tracks (the tracking algorithm). The Kalman predictor also filled the gaps in the tracks when markers were missing from view. Finally, the matched markers were checked against the motion model discussed in Chapter 3 to determine the correctness of the match.

4.1 Experimental Setup

4.1.1 The Flock of Birds

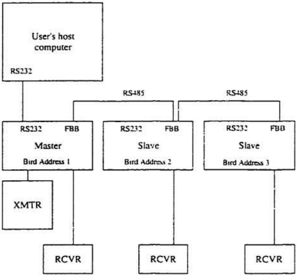

Figure 4.2 FOB configuration with three birds

The FOB is a "six degree-of-freedom" measurement device that can track position and orientation of up to thirty magnetic markers (Flock of Birds, 1995). Figure 4.2 shows the setup used in the laboratory. It has one transmitter (XMTR) and three receivers (RCVR). The FOBs are connected to the host computer through a full duplex RS232C interface. The second interface is a dedicated RS485 interface for communication between flock members and is generically called the Fast Bird Bus (FBB). A picture of the FOB system along with the three transmitters/receiver on the wooden arm can be seen in Figure 4.3. A picture of the FOB system can also be seen in appendix B.

The FOB system is capable of making from 10 to 144 measurements per second depending on various factors such as the number of birds used, the communication rate, the type of measurement required, etc. With the current setup of three birds (markers) and a communication (baud) rate of 9600, the maximum sampling rate for measuring positional data is between 32-38 Hz.

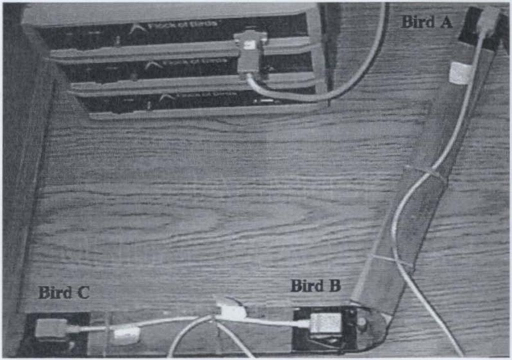

Figure 4.3 Wooden arm attached with birds

When set to measure positional data, each receiver (the bird) returns the x, y, and z coordinate distances, in inches, from the transmitter. The transmitter is the reference point with coordinate (0, 0, 0). The static positional accuracy of the FOB is stated to be .1" RMS average (Flock of Birds, 1995) over the translational range. From static measurement trials conducted it was shown that the measured position values had a higher variance when a bird remove the curvature of the data (since we expect a line), it does remove much of the high frequency noise (smooths the data). Another possible source of error is that the transmitter generates magnetic fields and hence is very sensitive to ferrous metals and thick obstructions. This made the gathering of valid data extremely difficult.

Since the data from the FOB was already in 3D coordinates, no effort was made to determine a calibration or computation to remove the curvature of the lines. Any errors in the measurement were assumed to be the actual motion of the object in 3D space. This error was allowed because in general situations there were many unknowns with markers placed on skin such as slippage of skin over a region and amount of body fat that caused the marker to wobble differently with different subjects. This 3D data was then transformed to the image planes of the simulated cameras to simulate measurement from a camera-aided vision system.

4.1.2 Simulating Camera Positioning

A simple pinhole camera model is simulated using the position (x, y, z), the orientation (θ_x, θ_y, θ_z) parameters of the camera with respect to the world coordinate reference point (location and orientation of the FOB transmitter) and the focal length of the camera.

The 3D reconstruction of images require a minimum of two cameras. Therefore, for testing purposes and simplification of the algorithms it was assumed that only two cameras were used. Figure 4.6 shows two cameras, with independent axes, pointed to a track or area where for the "angle model algorithm."

The first motion, run #1 (Figure 4.8), was a slow periodic circular movement over the top of a plastic cylinder. The cylinder was weighted down to avoid any accidental movements and carefully set to be within the range of the transmitter for the FOB. The figure shows a large movement in the x and y directions and minimal movement in the z direction.

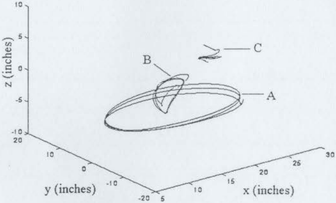

Figure 4.8 3D representation of motion history of the markers A, B, and C in run #1

The second motion, run #2 (Figure 4.9), was a periodic motion over a semi-rectangular configuration with rounded corners. The rectangular object was positioned vertically and placed at a 45° angle in the x-y plane to allow for large changes in the x, y and z coordinates. The figure shows that marker A has the greatest range in motion and is periodic. Marker B also gives a periodic motion with a reduced range. And marker C has a small planar periodic motion, since a slight rocking of the body was required to complete the rectangular motion.

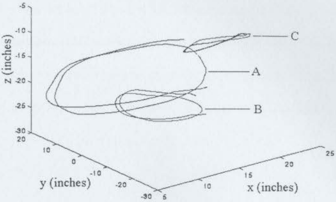

Figure 4.9 3D representation of motion history of markers A, B, and C in run #2

The third motion, run #3 (Figure 4.10), was created by the subject flexing his/her arm. In this motion, marker A was moved to marker position D to simulate markers in close proximity with each other. The subject was positioned in the transmitter's range so as to give the greatest motion in the x, y and z planes. The close proximity of the markers B and D allowed for testing of overlapping ROAs.

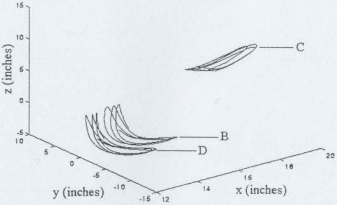

Figure 4.10 3D representation of motion history of markers B, C and D in run #3

This algorithm was tested by injecting different levels of white noise in the camera image planes. The injected noise in the image planes of both cameras simulates digitizing errors, numerical imprecision and electronic noise in video cameras. Appendix C describes the types of noise in charge-coupled device (CCD) cameras and describes the procedure used to inject noise in the image plane and determine how the noise affects the precision of the marker coordinates. The RMS error in pixels was calculated for three different intensities of noise: 5, 10 and 20 percent. Table 4.1 contains one row of data from Table C.1 and these RMS values are used to corrupt the image coordinates to varying degrees by adding or subtracting a randomly generated percentage of the RMS error from the known image coordinates.

Var	Type	5% Error		10% Error		20% Error	
		RMS_X	RMS_Y	RMS_X	RMS_Y	RMS_X	RMS_Y
10	Edge	0.004	0.003	0.008	0.007	0.027	0.0221

Table 4.1 RMS error in pixel values caused by different intensities of white noise

To assess the success of the algorithm, the output array of matches was visually checked against an expected output. The algorithm was considered to have failed if any of the marker positions were found to be matched incorrectly. For example, if the marker A in camera 1 was matched to marker B in camera 2 then the algorithm had failed. For each noise intensity the algorithm was run three times (since the noise is random and gives different percentage values) and the average percentage error of failures was then calculated. The error was rounded up to the nearest 5%. Table 4.2 shows the results of the test for the 0%, 5%, 10% and 20% noise levels.

Before injecting the noise, the markers were matched between the images. This procedure was successful for all three data sets. It can be seen from Table 4.2 that with less than 20% signal-to-noise ratio in the image plane, the matching process failed. As was stated in the literature review section, geometric matching was not sufficient for matching in noisy scenarios and additional constraints needed to be added.

With run #1 more than 5% failures started at approximately 15% noise level, run #2 started failing at approximately 17% noise level, and run #3 started failing at 8% noise level. The % error entry showed the maximum percentage of failures, i.e., the % error for run #1 at 20% noise reads as less than 20% of the markers were matched incorrectly. In other words 80% of the matches were done correctly.

Run #1		Run #2		Run #3	
% Noise	% Error	% Noise	% Error	% Noise	% Error
0	0	0	0	0	0
5	0	5	0	5	0
10	0	10	0	10	<10
20	<20	20	<15	20	<85

Table 4.2 Results from noise injection test of the epipolar matching algorithm

Improving the matching capabilities of the epipolar algorithm by the addition of extra matching constraints was not considered. The reason being that, since these markers were moving through 3D space without simple constraints, it would be difficult to establish additional physical constraints in the image plane without introducing many computationally heavy rules. Another reason was that in commercial systems all the markers were identical and the lighting conditions and coatings on the markers were set to maximize the view of the markers and remove all other background features. Therefore, no other identifiable features such as edges could be used to establish relationships between markers.

One solution to improve matching in noisy environments is to use active markers such as color coded markers or markers blinking with a set frequency. Another solution is to use the distance between markers as fixed lengths (Personal Correspondence, Mah 1996, Strukol 1996). But this would require extra information about the connectivity of the markers in the image plane, as well as the orientation of the lengths. Another would be to use specific marker sets such as using three markers per body segment (Personal Correspondence, Forstien 1996). This would require more complex algorithms for tracking and prediction of the markers.

In addition to the white noise injection tests, two additional tests without noise were done to test the constraints. In the first test, image points were removed from one camera plane to simulate obstructions to the camera and occlusion of markers, and the matching algorithm was executed. The algorithm successfully matched the corresponding image points and discarded the unmatched ones. In the second test, additional marker coordinates were added to the image plane of one of the cameras to simulate unwanted artifacts. The test for missing markers passed without any errors. The matching test only failed in test cases where the injected point was closer to the calculated epipolar line than the actual point. A form of linear programming (Ayache 1991) in addition to epipolar matching may solve this problem. This was not pursued further because it wasn't the main purpose of this work and because the acquiring of data from cameras and their 3D reconstruction algorithms were usually provided with the hardware.

4.2.2 3D Reconstruction

This algorithm reconstructed 3D (world) coordinates using matched 2D (image) coordinates. It used the image match information and the transformation matrices of the cameras in a least squares approach to reconstruct the 3D coordinates. The same noise injection method used in the epipolar matching was used for testing the 3D reconstruction. Different noise levels were added to the image plane data values to determine the effect they would have in determining the 3D positions. To avoid any problems of incorrect matching, the noise in the image plane was introduced after the matching algorithm was completed. A squared error (in the x, y, and z direction) was calculated between the observed marker location and the reconstructed location to determine the performance of the reconstruction algorithm. This error gave the minimum distance between the two points. The observed location values (x, y, z) were subtracted from the calculated location values, then squared, then summed, and the square root of the sum was obtained.

Table 4.3 shows the results of this test for noise ratios of 5%, 10% and 20%. For runs #1 and #2, the markers used are A, B, and C and for run #3 the markers used are markers D, B, and C (refer to Section 4.1.3 for explanation on change of markers).

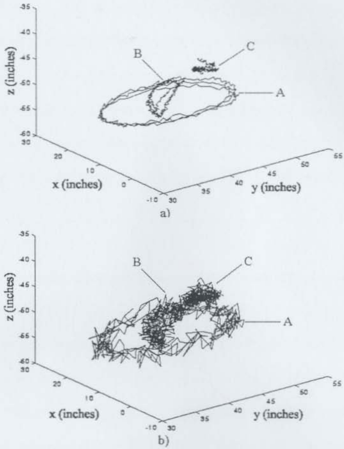

Figure 4.12 Reconstructed 3D tracks of run #1: a) 5% noise and b) 10% noise

4.2.3 Kalman Prediction Matching of Disjointed Tracks

Consider one sampling frame, t, of this two camera simulated vision system. The output from the system is composed of two sets (list) of image coordinate data. These two sets of image coordinates are matched to their corresponding markers in the two images based on the epipolar constraints. After the image coordinates are matched, the three dimensional coordinates are calculated. At time t+1 a new set of data is captured and the 3D coordinate data reconstructed. These coordinate data have no determined relationship to marker data from previously sampled frames. This algorithm serves two purposes. The first purpose of this algorithm is to match the marker data from time step t to the marker in time step t+1. i.e.. marker A in time step t is still marker A in time step t+1 and so on until time step t+n: to track the markers through space. The second purpose is to predict the motion of the marker while it out of view of the vision system so that a correct match can be made once the marker is back in view.

In order to track the markers through space. the Kalman prediction algorithm. at time step t. predicts the motion of a particular marker one step into the future. t+1. This one step prediction has an associated region of uncertainty around it which is assumed to be system noise and measurement error. At time step t+1 when the 3D locations are extracted from the vision system a search is done to match the observed (extracted) markers with their prediction (regions) from time step t. If a marker. at t+1. falls within the region of uncertainty (acceptance) for a particular prediction. from t. then it is assumed by the algorithm to be that same marker. So the connection is made and a track of that marker is determined.

When data is missing for a period of time. the Kalman predictor continues its prediction for as long as the data is missing. The moment the marker comes back into the view of the vision system. the algorithm attempts to make a match to the previous marker locations based on the last prediction. The only difference between this multi-step matching and the one-step matching is that the region of uncertainty will be much larger (Section 3.4.2). For testing this algorithm, only the three data sets from the Flock of Birds system were used.

The reconstructed marker data, from the previous subsection, was used as the input for this algorithm. At this stage, no noise was added to the coordinate data since the effect of noise was analyzed in the previous stages of the vision system, discussed in Section 4.2.

Two conditions (tests) were devised for the algorithm. The first test involved the addition of extra markers, while the second test involved the removal of markers. A squared error was calculated between the algorithm's matched marker (the final track output) and the observed marker. This showed the average error based on the length of prediction. It also showed if there was any type of relationship between the error and the length of prediction. The marker-to-track matching capability of the algorithm was tested visually (on the computer screen) by observing the actual tracks against the predicted tracks. For the algorithm to pass, the observed markers had to be matched to their correct tracks. As mentioned in Section 3.4, matching markers to their tracks was done using three constraints which are: (i) the prediction ROAs; (ii) distance of the observed marker from the center of any ROA and; (iii) matching of the velocity profile of an observed marker with that of a predicted track. The test involving the addition of extra markers passed for most cases and the correct matches between the observed markers and their corresponding track histories were found. The only time the matching failed was when there were two markers in the ROA and the inserted marker was closer to the ROA center than the observed marker.

This failed case appeared after the prediction of many (>30) time steps of a marker track. As was stated before, the longer the prediction was allowed to run, the greater the region of uncertainty and therefore the greater the ROA and the greater the possibility of extra markers falling in the ROA. This large ROA, as can be seen in the following tests, did cause some problems with matching. However, the introduction of checking the matched 3D markers against the physical model, as shown in the next section, eliminated this erroneous matching.

Significant testing has been done to test the prediction and matching of the marker locations when the markers disappeared from view for different lengths of time. As a reference case, the prediction and matching was run with no missing data and the error calculated was zero. Twelve tests were done for each run by changing the size of the gap of missing data and changing the number of missing markers. In tests 1-6 one marker at a time was removed for a period of time which increases from test 1 to test 6. For each of the three runs, sampled at 30 Hz, the location and length of missing marker segments are given in Table 4.4 (substitute marker D for marker A in run #3). Figure 4.13 gives a simulated example (missing gaps were exaggerated) of test no. 1 in Table 4.4 which shows three separate tracks with different segments of data missing.

Similarly, for the tests 7-12 two markers were removed simultaneously for increasing periods of time. Three combinations of the three markers are given for each. Table 4.5 shows these combinations. The first row in the table shows that markers A (D) and B have a gap at 10 - 20 time steps, markers B and C have a gap at 80 - 90 time steps and markers A (D) and C have a gap at 150 - 160.

Test No.	Length (sec)	Location of missing data (time steps)		
		Marker A (D)	Marker B	Marker C
1	⅓	10 - 20	80 - 90	150 - 160
2	⅔	10 - 30	80 - 100	150 - 170
3	1	10 - 40	80 - 110	150 - 180
4	1⅓	10 - 50	80 - 120	150 - 190
5	1⅔	10 - 60	80 - 130	150 - 200
6	2	10 - 70	80 - 140	150 - 210

Table 4.4 Tests 1-6 based on length and location of missing data

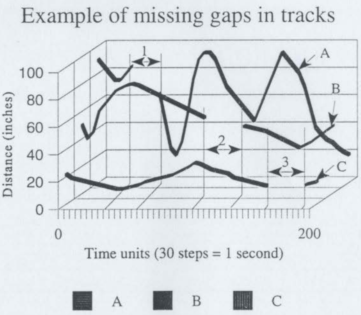

Figure 4.13 Three marker tracks with one gap (missing markers) per track (No. 1)

The rest of this sub-section is separated into three parts, one for each run.

4.2.3.1 Run #1

As can be seen in Table 4.6, for run #1 all the tests passed, but to match the markers to their tracks after the 1⅓ seconds (4) mark required the use of all three constraints.. In run #2, the motion failed at the 1⅔ seconds (5) mark for both the one (1-6) and two (7-12) marker missing tests. It required the first two constraints to make proper match after the ⅓ second (missing length of time) mark. Also, in this case, the third constraint was not useful in obtaining a proper match. With run #3, since the markers were much closer (B and D) and the motion was faster, the matching algorithm failed with the two missing markers tests (7-12) at ⅔ seconds (8). It also failed at the 2 second (6) mark with the one missing marker tests (1-6). And again, the third constraint did not prove to be useful in this test run. The reason for the failures in all the cases was that the ROAs became too large, more than one marker fell in their region and the wrong marker was matched based on the distance measurement to the ROA center.

Tables 4.7, 4.8, and 4.9 (4.8 and 4.9 are given later) show the squared error of selected tests for the three runs. Table 4.7 contains the squared error values for the ⅓ second, 1 second, and 2 seconds for both the one and two markers tests of run # 1. Table 4.8 contains the squared error values for, ⅓ second, 1 second, and 1⅓ second tests for both the one and two marker missing tests of run #2. Table 4.9 contains the squared error values for ⅓ seconds, 1 second and 1⅓ seconds tests are shown for the one marker test and ⅓ seconds for the two marker test of run #2. The general trend that can be observed in these tables is that while the markers are matched properly, especially for run #1, the longer the prediction of the motion was allowed to continue, the greater the error between the prediction and the actual motion.

Test no. and gap	Error square mean (inch^2)	1 missing marker			Test no. and gap	Error square mean (inch^2)	2 missing markers		
		A	B	C			A	B	C
1 (⅓ s)	ϵ_x^2	0	0	0	7 (⅓ s)	ϵ_x^2	0.01	0.01	0
	ϵ_y^2	0	0	0		ϵ_y^2	0	0.01	0
	ϵ_z^2	0	0	0		ϵ_z^2	0	0.02	0
3 (1 s)	ϵ_x^2	0.15	0.03	0.041	9 (1 s)	ϵ_x^2	0.205	0.767	0.05
	ϵ_y^2	0.393	0.196	0.1282		ϵ_y^2	0.225	0.464	0.04
	ϵ_z^2	0	0	0.03		ϵ_z^2	0.08	0.357	0.06
6 (2 s)	ϵ_x^2	1.097	0.906	0.3045	12 (2 s)	ϵ_x^2	1.486	7.696	0.225
	ϵ_y^2	4.232	3.359	1.4433		ϵ_y^2	2.102	5.632	0.07
	ϵ_z^2	0	0.02	0.407		ϵ_z^2	0.571	2.012	0.508

Table 4.7 Selected squared error results from run #1

Table 4.7 shows that there is an increase in error between the known location and the predicted location as the time period for prediction increases. As the prediction was allowed to run beyond the full 2 second interval, the prediction diverged from the actual marker paths. There seems to be no other discernible relationship between the length of the prediction and the error. The divergence for markers A, B, and C can be seen in Figures 4.14, 4.15, and 4.16 which shows test case 12 of run #1.

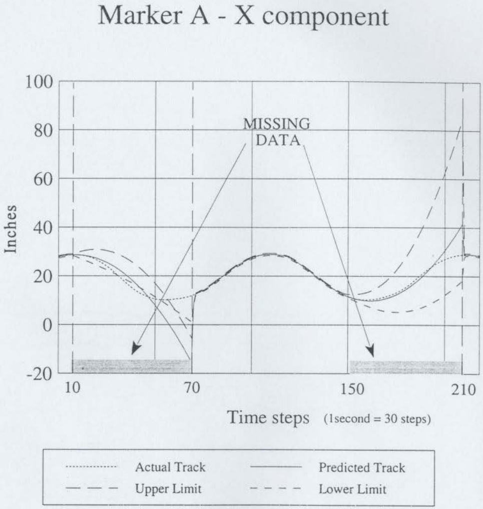

Figure 4.14 (a) X Coordinate Track of Marker A for run # 1 (passed)

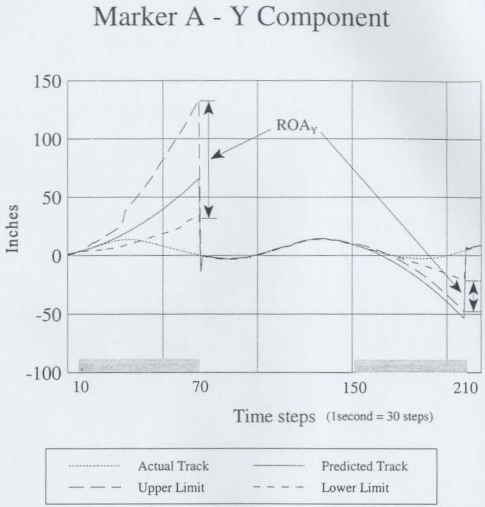

Figure 4.14 (b) Y Coordinate Track of Marker A for run # 1 (passed)

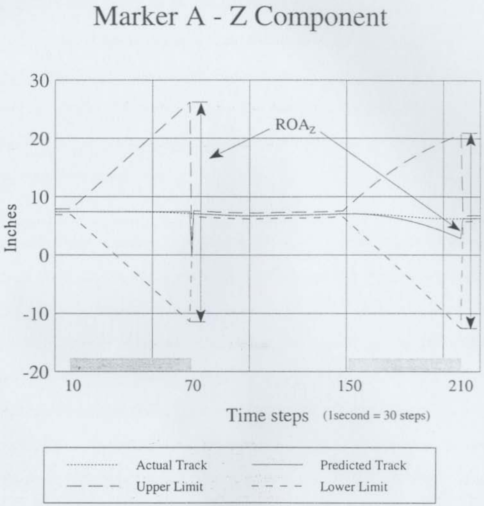

Figure 4.14 (c) Z Coordinate Track of Marker A for run # 1 (passed)

It can be seen in Figure 4.14a and Figure 4.14b that the Kalman filter is accurate in predicting the motion for approximately half of the missing data interval and is as far as 5 feet away at the end of the prediction interval (10-70). To match this ROA to the correct observed marker requires the using the ROA constraint, as well as the distance and velocity constraints. It can be seen from the actual and predicted lines that the correct match is found for this marker at the end of the first interval.

At the end of the second interval (150-210), it is seen that both the X and Z ROA components contained both the observed marker and predicted locations, while the Y component ROA contains only the predicted marker position with the observed marker position approximately 2 feet away from the edge. In this case only the ROA and the distance (second) constraint are needed to match the predicted marker track's ROA to the observed marker location. In this interval, the Kalman filter correctly predicts the motion of the marker for a third of the interval (150-210). The divergence of the Kalman filter from the observed marker values occurrs at the point when the acceleration of the marker changes significantly.

Figure 4.15 also has two intervals of missing data at 10 to 70 and 80 to 140. In the first interval (10-70), only the ROA constraint is used for matching until about the 49th time step when the Y coordinate (Figure 4.15b) prediction ROA no longer contains the observed marker locations. Then the secondary (distance) constraint is used to match the observed marker position to the ROA of the predicted track.

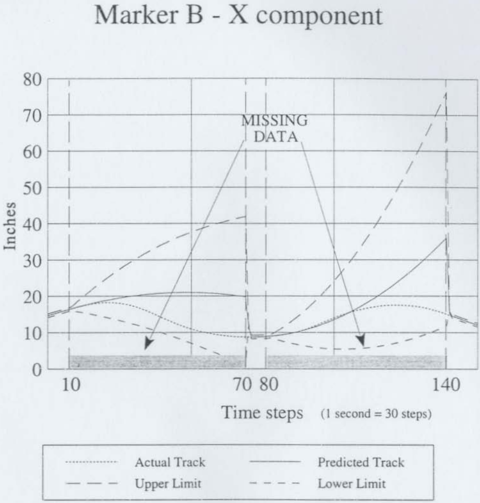

Figure 4.15 (a) X Coordinate Track of Marker B for run # 1 (passed)

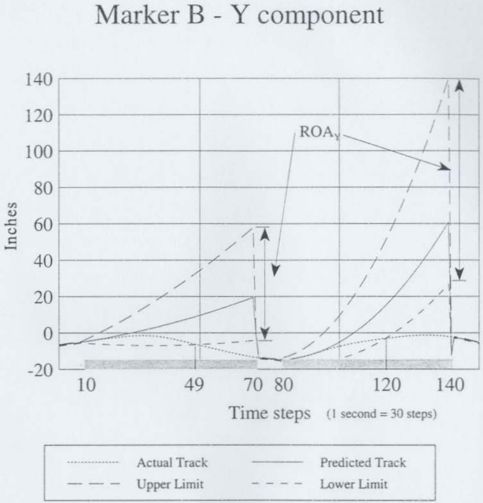

Figure 4.15 (b) Y Coordinate Track of Marker B for run # 1 (passed)

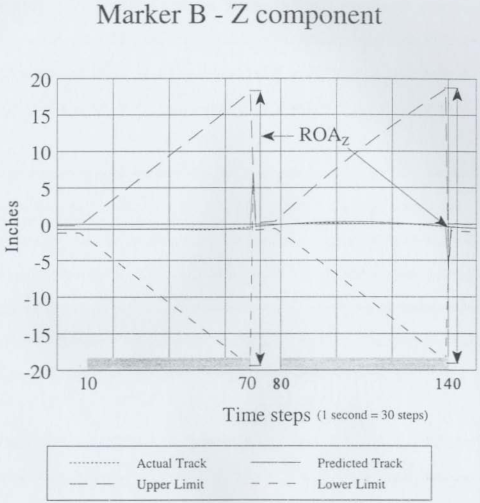

Figure 4.15 (c) Z Coordinate Track of Marker B for run # 1 (passed)

The use of the two constraints (ROA and distance) continuos until the end of the interval (70). It is seen that the Kalman filter accurately predicts the motion of the missing marker for approximately 15 time steps (½ a second) and by the 39th time step. in the interval. the prediction and it's associated ROA no longer contain the observed marker location.

The prediction at the second interval (80-140) followed a pattern similar to the first interval. where only the ROA constraint is used for matching until the Y coordinate prediction. at the 120th time step, and its associated ROA no longer contains the actual observed path of the markers and thus the addition of the distance constraint is needed to perform the correct matching. In this interval it is observed that the Kalman filter predicts the motion accurately for over 20 time steps and by the 49th time step, in the interval. the prediction and it's associated ROA are no longer contained the observed marker location.

In Figure 4.16. the data missing intervals are between 80 to 140 and 150 to 210. In the first missing data interval (80-140). the ROA of the predicted track always contains the observed track. therefore only the first constraint (ROA) is used in matching. In the second interval (150-210), the distance (second) constraint is necessary for matching when the Y coordinate ROA of the predicted track no longer contains the actual track data at the 180th time step.

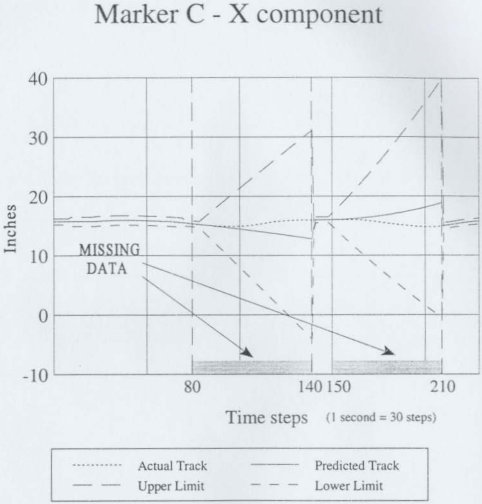

Figure 4.16 (a) X Coordinate Track of Marker C for run # 1 (passed)

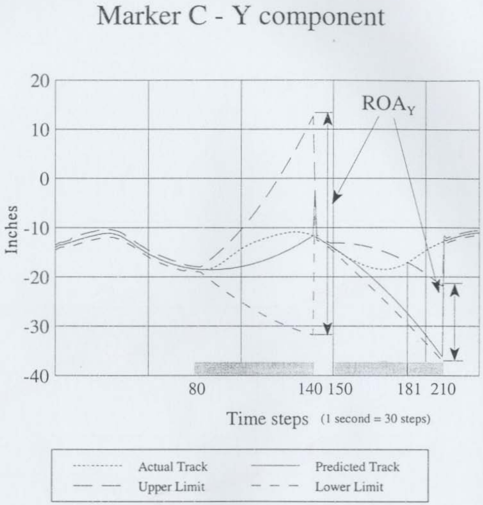

Figure 4.16 (b) Y Coordinate Track of Marker C for run # 1 (passed)

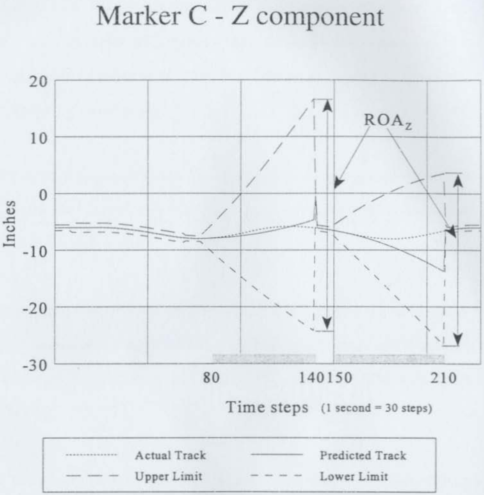

Figure 4.16 (c) Z Coordinate Track of Marker C for run # 1 (passed)

It is seen in both intervals that the Kalman filter accurately predicts the motion for only 10-15 time steps. Even though the prediction and its associated ROA, for the major part of the interval, contain the observed marker and is in close proximity to the observed marker, the motion is not accurately predicted.

The three Figures 4.14, 4.15, and 4.16 show that the prediction continues based on the last known values for position velocity and acceleration, even though the motion had changed in the interval of prediction.

This is a limitation of the linear Kalman filter because there is no constraint provided to change the direction of the prediction when no marker is observed. This is why the prediction algorithm cannot be allowed to run for long periods of time without additional constraints.

The three groups of graphs in Figures 4.14, 4.15, and 4.16 shows that in the first 10 to 20 steps of the missing data sections, the algorithm predicts the motion of the markers adequately. If the missing markers became visible to the camera in this short period after they disappeared, the prediction would be accurate. Generally, the predictor diverged when the prediction was allowed to run longer and the more the actual marker motion underwent accelerations. For example, within the 80 to 140 time interval in Figure 4.16 for the X, Y and Z coordinates the prediction follows the actual motion closely. However, for the time interval of 150 to 210, the prediction diverges from the actual path. It is seen that the predictor is only good for prediction of complex motion for short periods of time since it is a linear predictor. The next section discusses a method to improve the prediction of the actual motion when the marker is out of view of the cameras.

In this run (#1), the algorithm is very successful in the task of connecting a marker to its track after a period of prediction. It is able to correctly match the markers to their corresponding tracks after the 2 seconds, for both the one and two missing marker tests.

4.2.3.2 Run #2

Table 4.8 shows the squared mean error of selected test from run #2. From Table 4.6 it is seen that the matching for this run failed at 1⅔ seconds for both the one missing marker and two missing marker tests. Table 4.8 contains data from selected tests: ⅓ second, 1 second and 1⅓ second tests for both the one marker missing and two marker missing tests. This is different from Table 4.7 in that the 2 second tests could not be included since it failed at the 1⅔ second tests.

Again this table shows an increase in error with the increase in prediction length. No other relationship between prediction length and error can be seen from the data. One of the reasons for not being able to determine any other special relationships between prediction and the error is that the error depends on the change in motion while it is out of view of the vision system. This means that a relationship between the error and type of motion (i.e. slow, fast, straight, twists and turns) may be determined but a relationship between the prediction length and error cannot be determined.

Test no. and gap	Error square mean (inch^2)	1 missing marker			Test no. and gap	Error square mean (inch^2)	2 missing markers		
		A	B	C			A	B	C
1 (⅓ s)	ϵ_x^2	0	0	0	7 (⅓ s)	ϵ_x^2	0	0	0
	ϵ_y^2	0	0	0		ϵ_y^2	0	0	0
	ϵ_z^2	0	0	0		ϵ_z^2	0	0.01	0
3 (1 s)	ϵ_x^2	0.04	0.03	0	9 (1 s)	ϵ_x^2	0.07	0.04	0
	ϵ_y^2	0.02	0.04	0		ϵ_y^2	0.04	0.07	0
	ϵ_z^2	0	0.05	0.01		ϵ_z^2	0.04	0.2	0.04
4 (1⅓ s)	ϵ_x^2	0.09	0.131	0	10 (1⅓ s)	ϵ_x^2	0.153	0.09	0.02
	ϵ_y^2	0.05	0.104	0.021		ϵ_y^2	0.09	0.171	0
	ϵ_z^2	0	0.113	0.01		ϵ_z^2	0.1	0.43	0.09

Table 4.8 Selected squared error results from run #2

The graphs for run #1 shows the successful runs, but for runs #2 and #3 only the failed test graphs are selected, one failure for each run. These graphs are also shown to explain how the matching process fails for the runs. The X component graph labels the length of the gap of missing data. The Y component graph labels the location of the error in matching and the ROA_Y (Y component of the ROA). The Z component graph labels the ROA $_Z$ Also, the data in the graphs have been trimmed to view only the area of interest.

Figure 4.17, from run #2, shows the track for test no. 11, Marker B, in which the error occurs causing the test to fail. The failure occurs in the first missing data interval (10-60) at the 55th time step.

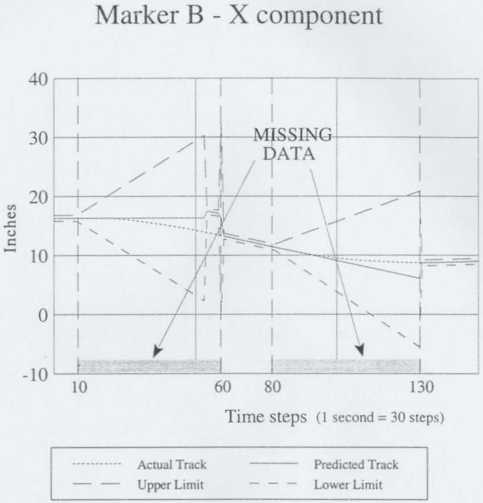

Figure 4.17 (a) X coordinate of marker track in run #2 which failed

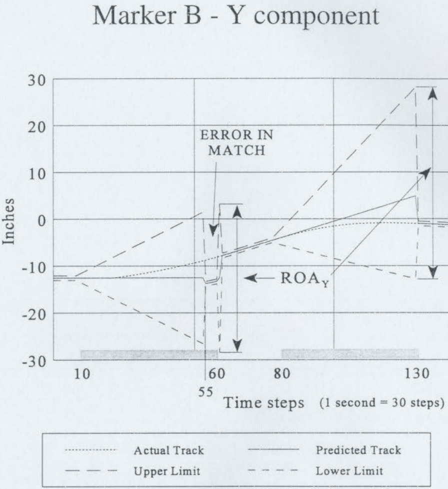

Figure 4.17 (b) Y coordinate of marker track in run#2 which failed

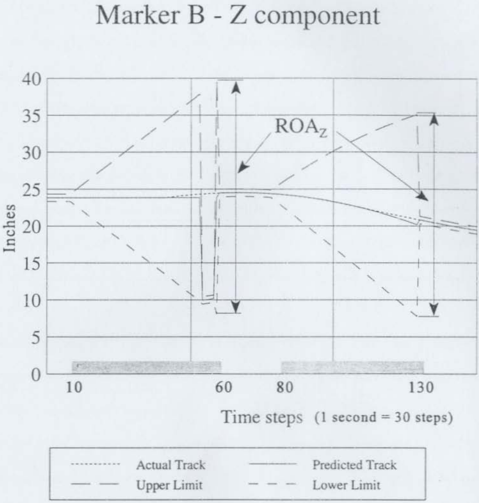

Figure 4.17 (c) Z coordinate of marker track in run #2 which failed

As can be seen in any one of the component graphs the ROA is increasing almost linearly until about time step 55, then a discontinuity starts and lasts until the end of the data missing interval at time step 60. The X and Y component graphs clearly show that this portion in this small error region has a different motion than it's previous track history (a discontinuity) and in the Z component graph the prediction in this error region (Z coordinate only) is nearly 15 inches away from the observed track. This incorrect matching occurs because another visible marker is encompassed by the growing ROA and therefore the marker is matched to the ROA's track (B's track). Once the missing marker (B) is back in view of the cameras the algorithm is able to recover from it's erroneous matching by using the distance constraint.

The two major differences between run #2 and run #1 is that there are slightly larger components of acceleration in run #2, seen by the sharper corners that were in the motion path for run #2, and the straighter lines of motion (Figure 4.9). The run fails because the ROA becomes large enough to encompass a neighboring marker and a wrong match is made. Despite the error in matching, the algorithm corrects itself once the missing markers becomes visible. On comparing Table 4.8 with Table 4.7 it is noted that the error in table 4.8 is much smaller. On further examination, while there are sections of motion with higher acceleration (the corners), the majority of the motion is almost linear and therefore the prediction generally tends to approximate the motion. Thus the error is much smaller. For this test case the Kalman filter predicts the motion of the missing markers accurately for longer gaps (40-50 time steps).

4.2.3.3 Run #3

Table 4.9 shows the error squared mean values from run #3. This table shows selected values for tests at the ⅓ second mark for both the one and two marker missing tests and at the 1 second and 1⅔ seconds marks for the one marker missing tests. This run, as can be seen from Table 4.6, failed at the 2 second mark for the one marker test and at the ⅔ second mark for the two markers test. Table 4.9 shows the large error at test no. 5. Because of the size of the error for Y coordinate it can be assumed that the prediction for that coordinate was rapidly diverging.

Test no. and gap	Error square mean (inch^2)	1 missing marker			Test no. and gap	Error square mean (inch^2)	2 missing markers		
		A/D	B	C			A/D	B	C
1 (⅓ s)	ϵ_x^2	0.02	0.107	0.033	7 (⅓ s)	ϵ_x^2	0.03	0.146	0.06
	ϵ_y^2	0.03	0	0.019		ϵ_y^2	0.07	0.09	0.08
	ϵ_z^2	0.02	0.02	0.044		ϵ_z^2	0.04	0.03	0.06
3 (1 s)	ϵ_x^2	0.357	2.085	0.4951		ϵ_x^2			
	ϵ_y^2	0.99	1.481	1.1842		ϵ_y^2		N/A	
	ϵ_z^2	0.291	0.14	0.4832		ϵ_z^2			
5 (1⅔ s)	ϵ_x^2	1.145	5.626	1.2966		ϵ_x^2			
	ϵ_y^2	4.056	10.53	5.3815		ϵ_y^2		N/A	
	ϵ_z^2	0.291	0.14	0.4832		ϵ_z^2			

Table 4.9 Selected squared error results from run #3

The main reasons for the early failure in the two missing markers test are that the markers B and D are much closer together and that the motion is highly nonlinear (Figure 4.10). However, with the one marker missing tests (1-6) the algorithm is still very successful at

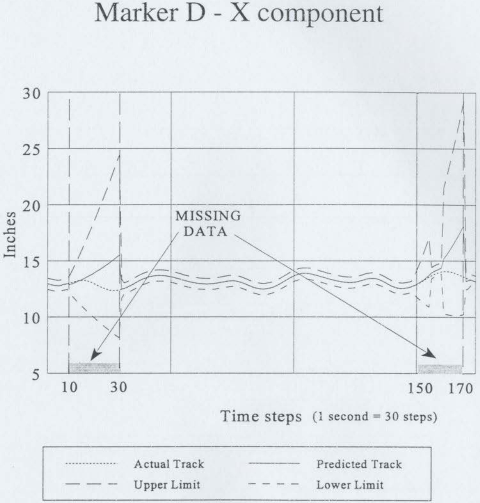

Figure 4.18 (a) X coordinate of marker track in run #3 which failed

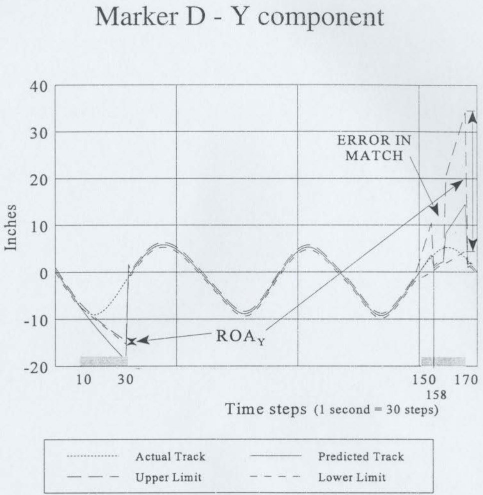

Figure 4.18 (**b**) Y coordinate of marker track in run #3 which failed

Also it can be concluded that this Kalman filter routine with the three constraints is excellent at matching markers to their tracks when there are no missing markers (tracking only) and it performs well for over short gaps in the data if only one marker is missing or if the motion is not too complex over the prediction period.

4.2.4 Distance Constraint Matching of Disjointed Tracks

To determine if the use of the Kalman algorithm is an improvement over simple matching, for example, drawing a straight line between two previous points and extrapolating a future point, a simple distance constraint algorithm has been developed to perform the matching of the disjointed tracks. This algorithm can not predict the motion of the markers when they are missing. It is assumed that once the matches between disjointed tracks are made a curve fitting/filling algorithm could be used to fill the gaps.

The Euler distance between observed markers at time frame t and the known marker tracks at time frame t-1 are calculated. For a 3 marker system this gives nine possible combinations (3 tracks and 3 markers) which are sorted based on shortest distance and then matched to the corresponding tracks. As each track and marker is matched, all other combinations for either the marker or the track are removed from the list of combinations until all the markers are matched to disparate tracks. For example, if one marker is missing then six combinations are calculated (3 tracks and two markers). The two marker and track combinations with the shortest distances between them are matched to each other leaving one track unmatched (the last known position for this track was stored). When the marker comes back into view, the combinations (3 markers and 3 tracks) are again calculated between the last known positions of tracks and the current observed markers. The distance between the tracks and markers are calculated and matching is done by using the shortest distance as the criteria. Table 4.10 contains the size of the gap (in time frames) where the matching of disjointed tracks fails for each of the three runs when there is one or two markers missing. It should be noted that the locations of the gaps are the same as those described in Section 4.3.3.

Run #	Size of gap at failure	
	1 missing marker	2 missing markers
1	18	13
2	31	25
3	3	2

Table 4.10 Length of gap before algorithm failed

The results show that the success of this technique is strongly related to the complexity and speed of motion as well as proximity of markers to each other. For simple motion and one marker missing, this technique works in matching markers with their corresponding tracks for long gaps (>20 frames @ 30 Hz - less than 1 second) as can be seen in run # 1 and #2. In run # 3 the proximity of the markers and the speed of the motion plays an important role in the failure of the algorithm. And with more than one marker missing the performance of the algorithm deteriorates. These results can be compared against Table 4.6 which shows the general location where the Kalman algorithm fails. For run #1 the Kalman algorithm performs the correct matching within a 60 frame (2 second) gap for both one and two missing marker tests. For run #2 the Kalman algorithm fails at the 50 frame gap (1⅔ second) for both one and two missing marker tests. For run #3 the Kalman algorithm fails at the 60 (2 second) frame gap for the one marker missing test and at the 20 frame gap (⅔ second) for the two markers missing test. It can be concluded that the Kalman matching algorithm is an improvement over this distance constraint technique.

4.2.5 Testing the Model

It was shown in section 4.2.3 that the Kalman predictor is not accurate in predicting the motion of the markers if they are out of view of the vision system for long periods of time (greater that 20-25 time steps). An extra criteria is added to the prediction algorithm to predict the actual motion of the markers when they were not visible. It is proposed that a physical motion model could be used to aid the Kalman prediction algorithm. In the tracking of other types of motion physical models, which modeled the approximate motion, have been used to help predict the motion more accurately. The physical model algorithm developed, in conjunction with the Kalman predictor algorithm, seeks to predict such motion.

The physical motion model developed for this work is the angle component model. The model uses the marker track position data and the relative positions of the markers to each other to create three component angle models, one for each coordinate. An algorithm is written to check the prediction from the Kalman predictor/matcher against the computed model.

The algorithm accepts 3D marker data after the prediction/matching routine used the ROAs and other constraints to match the observed and/or predicted markers to their corresponding

The tests consist of removing sections of track data as done in the previous subsection. The 12 cases listed are run again for this algorithm for the three sets of runs (#1. #2. and #3). All the tests passed the matching of the observed markers to their corresponding tracks. While the previous algorithm has failures as are noted in Table 4 6, no failures are observed by the use of the component angle model algorithm. Also, this algorithm accurately predicts the motion of the markers while they are out of view of the vision system. The following tables contain the calculated error squared mean values for selected test cases. These calculations are done to determine the error in the new algorithm and to provide a basis for comparing the performance of the addition of the new algorithm to the Kalman prediction/matching algorithm (alone).

Table 4.11 contains the error values for the same selected tests as in Table 4.7. These tests are the 1/3 second, 1 second and 2 seconds data for both the one marker missing and two marker missing conditions. As can be seen in the table, the error has increased as the length of the prediction interval has increases. The increase, however, doesn't seem to follow any specific pattern. The accuracy of prediction depends on the type of motion that occurs during the interval of prediction. If it is similar to the earlier motion, the predictor is accurate but if large changes in velocity and acceleration occur then the predictor is not accurate. The errors in magnitudes are smaller compared to the results in Table 4.7, especially when the interval of prediction becomes larger. The reason for this is that the model checking algorithm modifies the prediction as it moves away from it's model. This is illustrated clearly in Figures 4.19 , 4.20, and 4.21. These figures containes the graphs of predicted track, the actual track, and the upper and lower error bounds for the X, Y, and Z coordinates for run #1 and the 2 second test with two markers missing.

Test no.	Error square mean (inch^2)	1 missing marker			Test no.	Error square mean (inch^2)	2 missing markers		
		A	B	C			A	B	C
1	ϵ_x^2	0.01	0	0	7	ϵ_x^2	0.02	0	0
	ϵ_y^2	0	0	0		ϵ_y^2	0.01	0.01	0.02
	ϵ_z^2	0	0	0		ϵ_z^2	0	0	0
3	ϵ_x^2	0.05	0.02	0.065	9	ϵ_x^2	0.113	0.04	0.05
	ϵ_y^2	0.03	0.04	0.072		ϵ_y^2	0.198	0.06	0.105
	ϵ_z^2	0.01	0.02	0.081		ϵ_z^2	0.05	0.01	0.08
6	ϵ_x^2	0.08	0.03	0.107	12	ϵ_x^2	0.32	0.08	0.1
	ϵ_y^2	0.07	0.08	0.148		ϵ_y^2	0.82	0.129	0.213
	ϵ_z^2	0.03	0.367	0.16		ϵ_z^2	0.438	0.06	0.172

Table 4.11 Selected squared error results from run #1

These three figures have a similar format as the ones in the Section 4.2.3. All show only the portions where the prediction occurred to keep the resolution of the graphs as high as possible. Also to keep the graphs as uncluttered as possible only the X coordinate graph is marked to show the locations of the missing data and hence the prediction region. The Y coordinate graph is marked to show the oscillation of the model and the predictor. Finally, in the Z coordinate graph, two regions (1 and 2) are marked in the data missing interval (see coordinate Y graph). Region 1 shows, the Kalman prediction (alone) while region 2 shows the predictions with the added model checking. It should be noted that region 2 consists of both Kalman predictions with and without angular component corrections (when the prediction matches the model no correction is required).

In Figure 4.19, it can be seen from the X coordinate graph that the two intervals of prediction for marker A are 10 to 70 and 150 to 210 time steps (test 12). The Z coordinate graph clearly shows the two regions labeled 1 and 2. In the first interval (10-70), the Kalman prediction runs until the 17th time step is reached (27th time step in graph) where the error between the prediction and the expected model is large enough to invoke the model correction algorithm. This shift to the model algorithm can be seen in the Z coordinate graph because the ROA_z grows nicely in Region 1 and becomes seemingly erratic in Region 2 of both intervals 10-70 and 150-210. The oscillations are formed as the Kalman predictor and the angle component algorithm worked together. When the angle component algorithm's corrections are calculated and input fed back into the Kalman algorithm, the predictor's most recent time history data are changed. This obviously affects the future predictions. If the change between the predicted values and the corrected values are great then the Kalman filter interprets the change as a large acceleration and this causes an overshoot in the next predicted value. This prediction is checked against the model again and if it does not match the model, it will again perform a correction which is fed back into the predictor. This continues until the prediction is back on track with the expected model of motion.

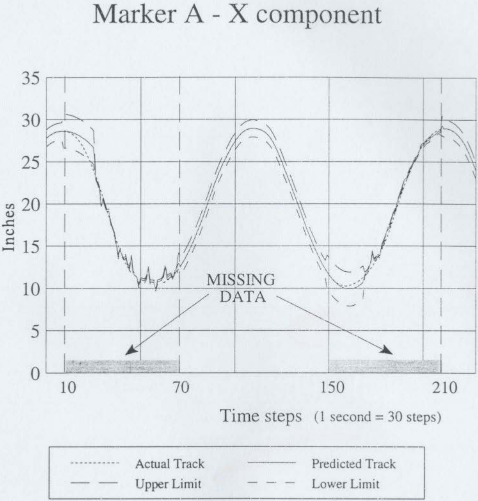

Figure 4.19 (a) X coordinate of marker A in run #1 with Kalman/Model algorithm

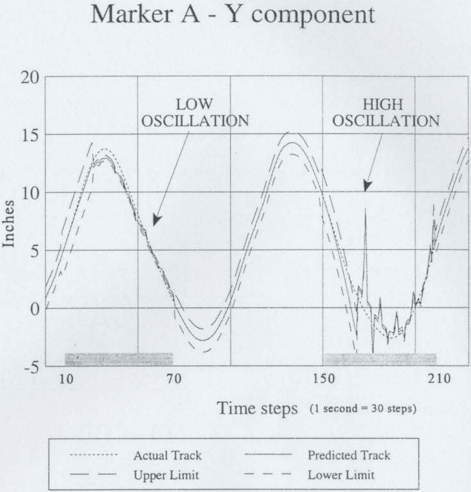

Figure 4.19 (b) Y coordinate of marker A in run #1 with Kalman/Model algorithm

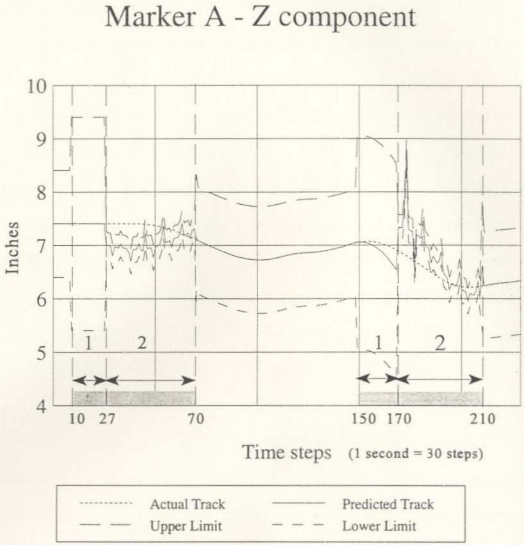

Figure 4.19 (c) Z coordinate of marker A in run #1 with Kalman/Model algorithm

The size of the oscillations varies depending on the coordinate graph viewed. For reference purposes it is observed from the Y coordinate graph that the oscillations in the first interval (10 to 70 time steps) are low and in the second interval (150 to 210 time steps) are high. The magnitude of the oscillations depend on the size of the correction between the prediction algorithm and the angle model algorithm. The greater the difference, the greater the magnitude. Once the prediction algorithm is on track, it will use only the Kalman filter until the predictions diverges again. It should also be noted that the error bounds, the ROAs, during the model/prediction stage are proportional to the variance in the measurement data. The removal of these oscillations is not easy since the corrections have to be input back into the Kalman filter. If they are not put back into the Kalman filter, then the filter will continue to predict along the erroneous path.

Figure 4.20, the graphs of the track of marker B, showes results similar to Figure 4.19. The intervals of prediction (for test no. 12), as seen in the X coordinate graph are between 10 to 70 time steps and 80 to 140 time steps. The Kalman prediction algorithm predicts the motion for 17 time steps in the first interval and 11 time steps in the second interval. The angle model algorithm combined with the Kalman prediction algorithm predicts the motion until the end of the intervals.

The Kalman prediction and angular model correction algorithms are accurate in predicting the motion in the X and Y coordinates. This can be seen by the low error values obtained for this run in Table 4.11. For the Z coordinate graph, the prediction of shape of the motion is accurate but the actual positioning is in error by an offset value. This error could have been caused by the variance in the rigid body length between marker B and marker A or marker C, whichever was correctly observed.

This slightly larger error rate can be from the results given in Table 4.11. Even though there is an offsetting error in the Z component, it is less than .2 inches. Therefore, in 3D space the combination of the two algorithms can be said to be very good in predicting motion.

Figure 4.21, which shows the motion of marker C, exhibits results similar to the previous graphs. The regions of prediction (for test 12), as seen in the X coordinate graph, are between 80 to 140 and 150 to 210 time steps. The prediction and model correction algorithms are good for all three coordinate data. The Z coordinate graph of this figure is good for illustrating, in the 150-210 interval, the predictor error growing larger until (about the 170 time step location) it gets large enough to be changed in the model algorithm.

All three figures, 4.19, 4.20, 4.21, show that the marker matching and prediction of the motion in the regions of missing data are excellent for run #1. Runs #2 and #3 show similar results and therefore the figures of the marker plots were not shown. However the error tables are given to show the improvement in the error between the output of the algorithms and the actual tracks.

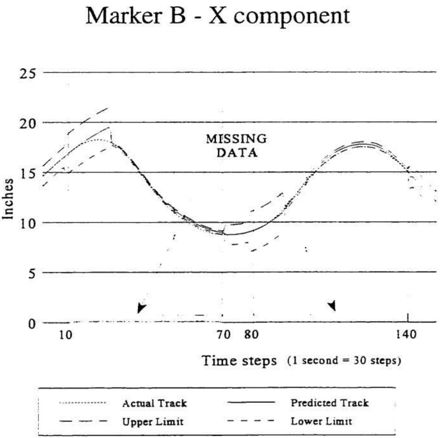

Figure 4.20 (a) X coordinate of marker B in run #1 with Kalman/Model algorithm

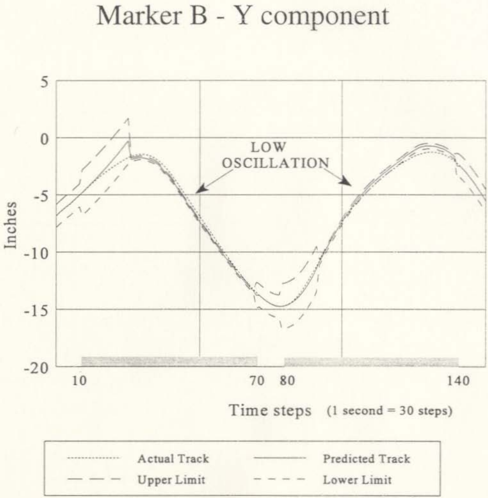

Figure 4.20 (b) Y coordinate of marker B in run #1 with Kalman/Model algorithm

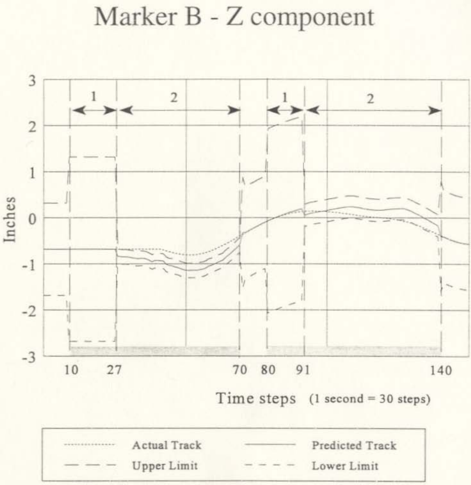

Figure 4.20 (c) Z coordinate of marker B in run #1 with Kalman/Model algorithm

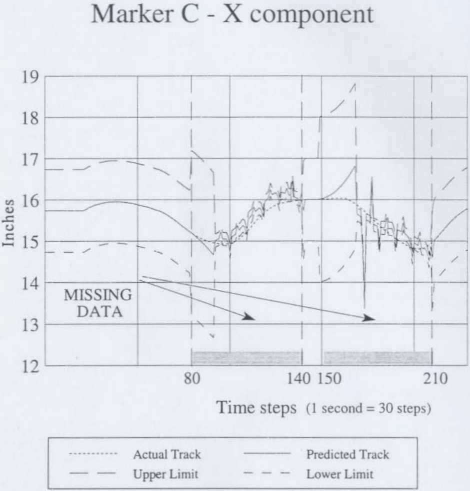

Figure 4.21 **(a)** X coordinate of marker C in run #1 with Kalman/Model algorithm

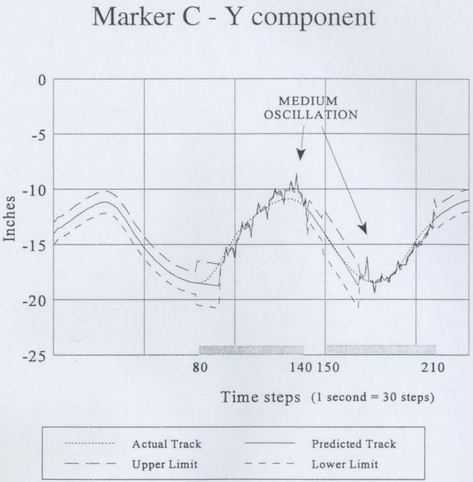

Figure 4.21 (b) Y coordinate of marker C in run #1 with Kalman/Model algorithm

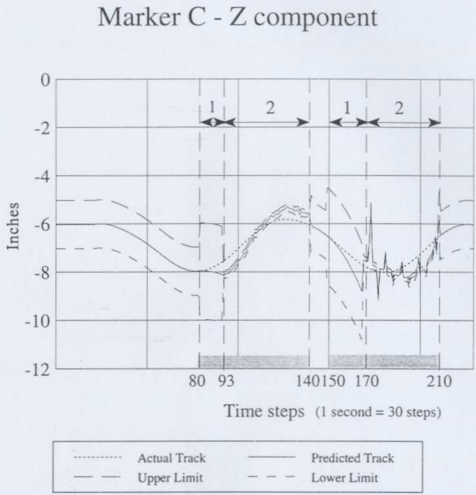

Figure 4.21 (c) Z coordinate of marker C in run #1 with Kalman/Model algorithm

Test no.	Error square mean (inch^2)	1 missing marker			Test no.	Error square mean (inch^2)	2 missing markers		
		A	B	C			A	B	C
1	ϵ_x^2	0	0	0	7	ϵ_x^2	0	0	0
	ϵ_y^2	0	0	0		ϵ_y^2	0	0	0.01
	ϵ_z^2	0	0	0		ϵ_z^2	0	0	0
3	ϵ_x^2	0.02	0.02	0.01	9	ϵ_x^2	0.05	0.02	0.01
	ϵ_y^2	0.02	0.04	0.036		ϵ_y^2	0.03	0.04	0.06
	ϵ_z^2	0	0	0.01		ϵ_z^2	0	0	0.02
4	ϵ_x^2	0.02	0.03	0.01	10	ϵ_x^2	0.07	0.03	0.02
	ϵ_y^2	0.02	0.05	0.052		ϵ_y^2	0.05	0.06	0.09
	ϵ_z^2	0	0	0.01		ϵ_z^2	0.02	0.02	0.03

Table 4.12 Selected squared error results from run #2

Test no.	Error square mean (inch^2)	1 missing marker			Test no.	Error square mean (inch^2)	2 missing markers		
		A	B	C			A	B	C
1	ϵ_x^2	0	0	0.011	7	ϵ_x^2	0	0	0.06
	ϵ_y^2	0	0	0.01		ϵ_y^2	0.03	0.01	0.03
	ϵ_z^2	0	0	0.021		ϵ_z^2	0.02	0.01	0.109
3	ϵ_x^2	0	0.02	0.029	9	ϵ_x^2	0.02	0.02	0.06
	ϵ_y^2	0	0.01	0.021		ϵ_y^2	0.08	0.05	0.05
	ϵ_z^2	0	0.02	0.037		ϵ_z^2	0.06	0.05	0.112
5	ϵ_x^2	0	0.03	0.029	12	ϵ_x^2	0.05	0.03	0.07
	ϵ_y^2	0.1	0.06	0.021		ϵ_y^2	0.168	0.102	0.125
	ϵ_z^2	0	0.06	0.037		ϵ_z^2	0.145	0.09	0.154

Table 4.13 Selected squared error results from run #3

The six tests error data in Table 4.12 can be compared directly with the data in Table 4.8. As can be seen by this comparison, the general trend is that the corresponding entries in Table 4.8 show greater errors than those in Table 4.12, with the exception of the C marker. This can be explained by the fact that the C marker output, as shown in Figure 4.21, is offset by a small error even though the shape of the predicted motion is comparable to the actual motion.

Table 4.13 contains the error values for four of the same tests as in Table 4.9 (only 4 tests available). This has been done so that comparisons could be made between the two tables. These tests are the ⅓ second for both the one and two missing marker tests, the 1 second and the 1⅔ second test for the one missing marker test. The other two consist of the 1 second test for two missing marker case and the 2 second test for the two missing marker case.

The error rate is generally much lower, by orders of magnitudes, in Table 4.13 than Table 4.9 between the four matching tests. As the interval of prediction increases, the gap between the two errors also increases quite dramatically. This can be seen in test 5 of Table 4.13 where the error values are much smaller than the test no. 5 of Table 4.9.

This section has shown that the addition of the physical model (to the Kalman predictor) has significantly improved the matching process between the observed/predicted marker locations and their corresponding tracks. This model also has greatly improved the capability of the software to predict the motion of the markers when they are out of view of the system.

4.2.6 Summary

This section has shown that the epipolar matching routine is adequate for matching image points between cameras as long as the noise in the system doesn't exceed approximately 10% of the signal strength.

The Kalman predictor/matcher is good at matching observed marker points with their tracks. It was shown to work well with the single missing marker tests; all three test runs passed for the 1 1/3 seconds test. It worked fairly well when two markers were involved; the only case where it failed badly was with run #3. However the Kalman predictor was not very good at predicting complex motions. It worked well for a few (15 to 20) time steps and then depending on the motion it might or might not have continued along the path of the marker. It was also noticed that the prediction algorithm diverged fast if the motion was fast and/or complex.

A physical model algorithm based on angular motion was developed to aid the Kalman predictor in checking the matches made and make corrections if necessary. This algorithm, along with the Kalman algorithm, worked very well at matching markers to their tracks and predicting the marker paths when the markers were out of view of the vision system.

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS

The study presented in this thesis was concerned with the marker tracking capabilities of computer-aided vision systems. The markers being tracked were placed at salient points on an arm and periodic motion exercises were performed. The study focused on the matching of markers with their corresponding track histories; matching segments of tracks with corresponding segments when portions of the marker's path, the track, were obstructed or lost from view of the vision system; and the recovery of the motion of the markers while they were obstructed or lost from view of the vision system.

Relevant theory was described in Chapter 3 and since a computer vision laboratory was not available for testing, smaller scale testing was done with the Flock of Birds (FOB) system. The FOB system provided positional data in 3D coordinates. This data was transformed to camera image coordinates of simulated cameras in a simulated gait lab. The data was also corrupted with white noise and put through a matching and reconstruction process to recover the 3D position coordinates. These recovered coordinates were then used in the prediction/matching/tracking algorithm.

Three test runs were examined by this simulated system. To get the periodic motion the markers were placed on a subject's limb and the subject had to move through specific paths. These were: (i) a slow motion over the mouth of a plastic cylinder (run #1); (ii) a faster one around a semi-rectangular object with rounded corners (run #2) and (iii) a still faster motion, the flexing of the arm of the subject (run #3).

The Kalman predictor was used to match the markers with their track histories and predict complex motion for short periods (⅓ to ½ of a second) of motion. The cycle times varied from run #1, approximately 2.2 seconds, to run #3 which was approximately 1 second (typical length for gait cycles). The Kalman algorithm showed a decrease in performance over the three runs. For the longer periods of prediction (greater than 1 second), the Kalman predictor was used with the angle component model. The angle component model was used to check the predictor matches against and internal physical model and make corrections as necessary. These techniques were successful in accomplishing the objective, i.e. the tracking of markers and predicting their motion when the markers were out of view of the vision system, set out by this study.

Considering the results, especially run #3, it is proposed that these techniques can be applied to lower limb gait studies to match image coordinates and reconstruct 3D coordinates from camera image data, match markers to their corresponding tracks, predict the gait, and make connections between disjointed tracks. The matching algorithm should be useful in lower limb studies because the markers are generally placed sufficiently far apart that the regions of acceptance will not overlap until after many steps of prediction. The next section will list the conclusions established from the above studies and give some recommendations for improvements in the study and possible areas of future research.

5.1 Conclusions

This study has shown several interesting results:

- Since the update of the filter variables is linear, it is good for prediction of complex motion only for brief periods of motion. In the test runs used to verify the algorithm, the Kalman filter accurately predicted the general motion for approximately ⅓ of a second.

- Though the Kalman predictor cannot accurately predict the motion of the markers when they are out of view of the vision system, the predictor with the simple constraint matching algorithm, consisting of the Region of Acceptance (ROA) and the distance to the center of the ROA constraints, is very good at matching markers to tracks after they have been out of view for over one second, depending on the type of motion. As noted in Chapter 4, the algorithm was excellent at matching markers to their tracks especially when only one marker was missing. In run #1 the predictor correctly matched the disjointed tracks for all single marker missing tests. In runs #2 and #3, the matching passed until the 1⅔ seconds and 2 second marks, respectively.

positions of markers.

- Currently the motion model algorithm knows where it is in the limb cycle and this makes the comparison between the predicted points and the motion model relatively simple. To generalize the algorithm an automated algorithm to match a predicted position to a specific position in a limb cycle would be an asset.

- The forward Kalman predictor was used to simulate an intuitive approach, i.e., since the motion was occurring in a forward direction, it made sense to predict it in the same direction. However, since the analysis is not done at real-time, backward and also forward predictions are possible and this would require some reworking of the predictor equations, but a possible advantage may be that the prediction would only run for half of the interval when the marker is missing, i.e., the forward prediction and backward prediction would start at the two ends of the interval and meet in the middle.

- Identification of specific events during the gait cycle would be an asset. This would help in tracking and making comparisons of the time length of these events.

- Since gait is periodic, a study of its frequency and harmonics may help characterize different facets of motion and diseases. An exhaustive literature search on the topic provided no information.

The performance degraded when two markers were missing. For the first two runs the matching with two missing markers was excellent, all tests passed for run #1 and all but the last two passed for run #2. However, with run #3, the slightly faster motion of markers and closer proximity to each other caused the matching to fail at the ⅔ second mark.

- The addition of a physical motion model, the angle component model, helped to aid the Kalman prediction/matching algorithm in predicting the motion over the periods tested. The addition of this model to the Kalman predictor/matching routine enabled the software to predict the motion of the markers accurately. This showed that the motion model was an excellent aid in the prediction of complex motion of limbs when the markers placed on them disappear from the view of the vision system.

5.2 Recommendations

- For matching between cameras, the use of color coded markers, different shape markers, banks of markers, or reflective strips along the axis of the limbs may enhance the tracking capabilities of a vision system.

- In improving the motion model, one could use the angle models that are standardized in hospitals. This was not done because it was determined that the angle graphs by themselves did not contain enough information to back calculate the expected

CHAPTER 6

REFERENCES

Ariel Web Page (1996). *http://www.arielnet.com*, Ariel Dynamics, Inc., Englewood, CA, USA.

Ayache N., (1991). *Artificial Vision for Mobile Robots: Stereo Vision and Multisensory Perception*, The MIT Press, Cambridge, Mass., London, England, pp 13–42.

Ball, K. A. <BALL@phe.utoronto.ca> (1996). "Re: Some Help please - gait analysis," *Personal Correspondence.*

Ballard, D. H. and Brown, C. M. (1982). *Computer Vision*, Prentice-Hall, Englewood Cliffs, NJ.

Biden E., O'Conner J. and Collins J. J. (1990). *Knee ligaments: Structure, Function, Injury, and Repair (Ch. 14 Gait Analysis)*, Raven Press Ltd., pp 291-311.

Bonnin, P. and Zavidovique (1991). "New Fast and Accurate Reconstruction Method of 3D Planes," *Intelligent Robots and Computer Vision X: Neural, Biological, and 3D Methods*, SPIE-The International Society for Optical Engineering, Boston, Massachusetts, pp 47-49.

Bozic S. M. (1979). *Digital and Kalman Filtering*, Edward Arnold (Publishers) Ltd, London, pp 115-120, 126-140.

Brown, R. G. and Hwang, P. Y. C. (1992). *Introduction to Random Signals and Applied Kalman Filtering*, John Wiley and Sons, Inc, pp 357-366.

BTS Web Page (1996). *http://www.bts.t/bts*, Bioengineering Technology & Systems (ELITE), Milano, Italy

Cao, C. <chengcao@engin.umich.edu> (1996). "Re: Some heip please - gait analysis." *Personal Correspondence.*

Flock of Birds (1995). *Installation and Operations Guide*, Ascension Technology Corporation, POB 527, Burlington, Vermont 05402

Faugeras, O. D. and Toscani, G. (1986). "The Calibration Problem for Stereo," *Proceedings CVPR '86*, Miami Beach, Florida, IEEE, pp 15-20.

Furnee, E. H. and Jobbagy, A. (1993). "Precision 3-D Motion Analysis System for Real-Time Applications, Microprocessors and Microsystems", Butterworth-Heinemann Ltd., Vol. 17, No. 4, pp. 223-231.

Geil, M. D. <Geil.1@osu.edu> (1996). "Missing Markers." *Personal Correspondence.*

Gonzalez, R. C. and Woods, R. E. (1992). *Digital Image Processing*, Addison-Wesley Publishing Company, Inc., New York.

Greaves, J. <John.Greaves@MotionAnalysis.com> (1996). "Re: Some Help please - gait analysis," *Personal Correspondence*, Motion Analysis Corporation, Santa Rosa, CA, USA.

Hof, A. L. <a.l.hof@med.rug.nl> (1996). "Re: Some Help piease - gait analysis," *Personal Correspondence.*

Hulliger, M. <manuel@cns.ucalgary.ca> (1996). "Marker obstruction - gait analysis," *Personal Correspondence*.

Johnsson, H. <hakan.johnsson@qualisys.se> (1996). "QUALISYS." *Personal Correspondence*. QUALYSIS, Sweden.

Kadaba M. P., Ramakrishnan H. K. and Wooteen M. E. (1990). "Measurement of Lower Extremity Kinematics During Level Walking", *Journal of Orthopaedic Research*, Raven Press Ltd., Vol. 8, No. 3, 1990, pp 383-392.

Kanazawa, Y. and Kanatani, K. (1995). "Reliability of 3-D Reconstruction by Stereo Vision," *IEEE Transactions on Information and Systems*, Vol. E78-D, No. 10.

Kasturi, R. and Jain, R. (1991). *Computer Vision: Principles*, IEEE Computer Society Press, Los Alamitos, California, pp 1-3.

Keezel, B. <tsrhgait@ix.netcom.com> (1996). "Re: Some Help please - gait analysis," *Personal Correspondence*.

Maas, H. G., Gruen, A. and Papamtonion, D. (1993). "Particle Tracking Velocimetry in Three-Dimensional Flows, Experiments in Fluids," *Springer-Verlag*, Vol 15, No. 2, Pp. 133-146.

Mah, C. <cmah@cns.ucalgary.ca> (1996). "missing markers," *Personal Correspondence*.

Mah, C. D., Hulliger, M., Lee, R. G. and O'Callaghan, I. S. (1994) "Quantitative Analysis of Human Movement Synergies: Constructive Pattern Analysis for Gait," *Journal of Motor Behaviour*, Vol. 26, No. 2, pp 83-102.

Muijtjens, A. M. M, Roos, J. M. A., Arts, T., Hasman, A. and Reneman, R. S. (1993). "Extrapolation of Incomplete Marker Tracks by Lower Tank Approximation," *International Journal of Biomedical Computing*, Vol 33, No 3-4, Pp 219-239.

Northern Digital Web Page (1996). *http://www.ndigital.com/*, Northern Digital, Inc. (OPTHOTRAK), Waterloo, Ontario, Canada.

O' Mallay, M. (1993). "Kinematic Analysis of Human Walking Gait using Digital Image Processing", *Medical and Biological Engineering and Computing*, Vol. 31, pp 392-398.

Olree, K. <kolree@roadrunner.carenet.org> (1996). "Missing Markers," *Personal Correspondence*.

Peak Performance Web Page (1996). *http://www,peakperform.com/*, Peak Performance Technologies, Inc., Englewood, CO, USA.

Ramachandran, K. V., Mohan, B. R. and Geetha B. R. (1993). "A Three-State Kalman Tracker Using Position and Rate Measurements", *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, No. 1, pp 215-221.

Ramachandran, K. V. (1987). "Optimal Steady State Position, Velocity, and Accleration Estimation using Noisy Sampled Position Data," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. AES-23, No. 5, pp 705-708.

Ramakrishnan, H. K., and Kadaba, M. P. (1991). "On Estimation of Joint Kinematics During Gait," *Journal of Biomechanics*, Vol. 24, No. 10, pp 969-977.

Roudanez, G. <georges@viconsys.com> (1996). "Re: mailto:georges@viconsys.com," *Personal Correspondence*, VICON Systems.

Smidt, G. L. (1990). *Gait in Rehabilitation*, Churchill Livingstone Inc, New York.

Sturkol, E. <esturkol@peakperform.com> (1996). "Missing Marker Interpolation," *Personal Correspondence*, Peak Performance Inc.

Sutherland, D. H., Olshen, R. A., Biden, E. N. and Wyatt, M. P. (1988). *The Development of Mature Walking*, Mac Keith Press, Oxford, Blackwell Scientific Publications Ltd. Philadelphia, J. B. Lippincott Co.

VICON Web Page (1996), *http://www.viconsys.com/viconsys*, VICON Motion Systems, Oxford, England and Santa Fe Springs, CA, USA.

Wise, J. <jwise@oitunix.oit.umass.edu> (1996). "Re: missing marker solution ," *Personal Correspondence*, Ariel Dynamics.

Wu, J. J., Rink, R. E., Caelli, T. M. and Gourishankar, V. G. (1989). "Recovery of 3-D Location and Motion of a Rigid Body Through Camera Image (An Extended Kalman Filter Approach)," *International Journal of Computer Vision*, Vol. 3, pp 373-394.

Zhang, X. <xudong@umich.edu> (1996). "Re: Some Help please - gait analysis," *Personal Correspondence*.

Zhang, Z. (1994). "Token Tracking in a Cluttered Scene," *Image and Vision Computing*, Vol 12, No 2, March, Pp. 110-120, Butterworth-Heinemann Ltd.

Zhang, Z. (1995). "Estimating Motion and Structure from Correspondences of Line Segments between Two Perspective Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 17, No. 12, pp 1129-1139.

APPENDIX A

PERSONAL CORRESPONDENCES

A.1 Letter of Request

From joshua@tera.engr.mun.caWed Nov 13 11:57:43 1996
Date: Tue, 10 Sep 1996 11:00:37 -0230 (NDT)
From: Joshua Swamidas <joshua@tera.engr.mun.ca>
To: Multiple recipients of list BIOMCH-L <BIOMCH-L@NIC.SURFNET.NL>
Subject: Summary - Missing Markers

Hello all,

I would like to apologize for the tardiness of this summary and I'd like to thank all of you who took the time to help me. The information was very helpful.

First, here is the message that I had sent:
I'm a master's student who is working in the area of gait. I have access to data from a multi-camera vision system. I am looking into the problem of missing markers and the ability to track them and predict their path while they are out of view of the vision systems. I have come across some interesting techniques.

Unfortunately, I haven't been able to find out how big this missing marker problem really is. I understand that as long as the markers placed in the anatomical sites are viewed by any two cameras the 3D positions of these markers can be calculated. However, in case of pathological gait, the use of canes, walkers, walking pattern, etc., can cause the marker to be obstructed from view of the cameras. I haven't been able to find any information to show if this really is a problem other than the words of a few people who work in gait laboratories.

In many of the vision systems, when the vision system loses sight of the markers the tracks of the marker paths become disjointed. Generally the user has had to connect the disjointed tracks manually. Is there software that makes these connections automatically? If so how do they do it?

Finally, in the prediction of motion, I have narrowed my search to using motion models of those limbs. Are there better techniques, esp. for pathological gait?

A.2 Grouping of responses

MISSING MARKERS:
Pertaining to the problem of missing markers, it seems that almost everyone agreed that it was a problem (to varying degrees of seriousness). The best solution, as one person said, is to make sue the

problem does not occur and this could be done by: increasing the number of cameras viewing the tracking region; using non-camera methods of tracking such as magnetic tracking[1]; carefully choosing a marker set that would stay in view despite obstructions, rotations, etc.; and using special assistive devices that minimize intereference with the cameras [2].

PREDICTING MARKERS AND CONNECTING DISJOINTED TRACKS:

As for actually filling in the gaps using software, I have broken this down into two separate sections: the first one deals with comments from individuals and the second would be the companies that specialize in motion analysis' hardware and software such as VICON, Ariel Dyanmics, Inc., QUALYSIS, Peak Performance Technologies, and Elite (BTS).

The techniques are listed in point form and the reference number attached to it is based on the order of the email responses which are concatenated to the end of this summary. The following is a list of techniques and algorithms:

- Join missing gaps using spline functions (cubic or quintic) [1]. It may not be very accurate over large gaps. Cheng cautions that some kind of human involvment is usually necessary when using visual systems.
- Using a proxy marker to provide an offset-recording [2,3] of the marker that consistently disappears from view. This proxy marker can then be used to infer the position of the missing marker.
- For short isolated gaps, using a straight line or polynomial algorithm works well [3]
- Extrapolate along a straight line between two markers [3]. For example, if you have trouble seeing an ankle marker, put two markers on the lower leg, so that the ankle is further along the straight line drawn between the twp markers.
- For constructing analytical models for analysis [3] some papers to consult are:
    - Quantitative analysis of human movement synergies: constructive pattern analysis for gait, C.D. Mah, M. Hulliger, R.G. Lee and I.O. Callaghan, Journal of Motor Behaviour, 26, 83-102, 1994.
    - Quatitative analysis techniques for human movement: finding multivriate patterns in large data sets, in M. Whitten and D.J. Vincent, eds.
    - Computational Medicine, Public Health, and Biotechnology: Building a man in the machine, Part II, pp. 1056-1069. World Scientific Press, 1996
- Curve fitting alogrithms are fine for small gap sizes (10 frames) [4].
- If the general forms of a specific movement are known, then a tracking algorithm can be written for a this specific problem [5].
- 3 markers uniquely define a segment's location and orientation in space. If a 4th marker is used, the redundancy could be used to calculate a missing marker. Some pros and cons of this technique are discussed in email [6].
- If the distances between three markers on a segment are known and the location of two of the marker are known, the location of the third marker can be constrained to a small area. This might be used in conjunction with a interpolative procedure to give a better estimate of location [6].
- If the data is cyclical, it may be possible to "guess" where a reasonable location of the marker might be based on where it was one period before or after the instant [6]. However, in gait analysis cases where an assistive device obstructs the camera view they tend to happen in the same potion of each stride [8].
- For path matching, sometimes paths can be easily identified by some unique characteristic of their location [6] such as a marker on the upper part of the body would have greater values

in the height coordinates.

- Using direction and velocity to help identify possible path matches [6]. This will work well if the gaps are short and there are no sudden reversals in direction of the marker.
- Again for path matching, using a marker path reconstruction technique to reconstruct a path and match it to an observed path [6].
- Other useful papers[9]:
    - Ball, K.A. and Pierrynowski, M.R. (1991), WalkTrak: Automated analysis of 3D kinematic data from video systems. Proceedingsof International Symposium on 3D Analysis of Human Movement, First Meeting, 28-31 July, Montreal, Quebec, Canada, 6-9.

  Papers on rigid body kinematics:
    - Ball, K.A. and Pierrynowski, M.R. (1995). Estimation of siz degree of freedom rigid body segment motion from two dimensional data, Human Movement Science, 14, 139-154
    - Ball, K.A. and Pierrynowski, M.R. (1996), Classification of errors in locating a rigid body, Journal of Biomechanics, Any day now (September).

The following is a list of the companies and a brief comment on their software.

Qualysis (Sweden & USA) [10] who manufactures the MacReflex Motion Measurement Systems for the Macintosh or PC computers gave me information on how they track the markers and what they do for prediction of the missing markers. In handling the tracking of markers, the Qualysis software extrapolation of the marker movements are carried out to approximate the next position of the marker. This extrapolation helps the tracker to look in the right place when looking for the next position of the marker. In situations where the adding of extra cameras will not help solve the marker out-of-view-of-cameras problems, the operator can run the tracking manually or run the tracking separately for different segments of motion stored in a file. Since the tracking can be run both forwards and backwards in the file, a point with missing markers can be approached and tracked from both directions. A user of this software (E-mail Correspondence, Zhang, 1996) commented that when the discontinuities were too large the filling in of gaps did not work well.

Peak Performance Technologies, Inc. (Englewood, CO, USA), says that their latest software [11] allows the user to connect the gaps manually by connecting points, semi-automatically by allowing the user to run different algorithms, or automatically by allowing the user to set several tracking parameters. According to the engineer, while much has been done using brute force mathematical extrapolation and interpolation, along with more cameras, to accurately fill in missing data, there is still plenty of room for improvement. Especially in the areas of using fewer cameras with software or hardware that is smart enough to fill in bigger and bigger gaps in the data.

The ELITE system by Bioengineering Technology & Systems (Milano, Italy) primary products include: ELITEplus (Three-dimensional motion package), ELICLINIC (Clinical gait analysis), TELEMG (Dynamic electromyography with no limitations on subject motion), GAITemg (Identification of muscle activity in walking), PcVect (Analysis of ground reaction forces), etc. The Elite software provides interpolation to handle relatively short gaps.

According to the founder and Senior Vice President of Motion Analysis Corporation, [12] pathological gait motion is difficult to track. They have software, the Track Mender, which looks at the continuity of 3D path segments and automatically joins them if they are no more than a few frames apart. Another technique used by their software is to allow the user to manually spline across the gaps.

Subject: Marker obstruction - gait anlaysis

Dear Joshua,

Chris Mah from our gait lab will send you a more comprehensive reply. There is therefore no need to include this one in your summary. At any rate, when you compose your summary, you would do the community a service by submitting a genuine digest, summarizing main points thematically and regrouping the various elemnts from different replies accordingly, rather than merely appending poorly correlated and hetero-genous replies in incoming order, as we see so often in the debates going on in this forum.

Marker obstruction often is a genuine problem, even in normal gait: with a two camera system arm movements obscure hip markers quite regularly and predictably. The Elite software we are using provides interpolation software to handle relatively short gaps. In addition, we are using a "tail" attached to the subject's lower back, to get an offset-recording of hip marker coordinates (in crude approximation). You may want to consult Mah et al., Journal of Motor Behaviour 26, 83-102, 1994.

With multi-camera systems (Vicon is going up to 7, I understand) this problem appears to be reduced in magnitude.

The above problems are exacerbated in the conditions you list. Canes and walkers indeed cause additional loss of marker information. For our work with cerebral palsy patients we have constructed a special walker to minimize such interference. For patients using canes, we use parallel bars to provide support (on the side remote to the cameras).

I hope this helps a bit. Chris Mah will reply more extensively.

You might want to visit the lab to get some first hand experience.

Yours,
Manuel Hulliger, D.Phil.
Department of Clinical Neurosciences. Faculty of Medicine
University of Calgary. 3330 Hospital Drive N.W.
Calgary, Alberta, Canada T2n 4N1

Phone: 403-220-6216 [Ellen Wong Phone: 403-220-8389]
Fax: 403-283-8770 [Secretary Fax: 403-283-8731]
E-mail:manuel@cns.ucalgary.ca or mhullige@acs.ucalgary.ca
www: http://www.cns.ucalgary.ca/people/hulliger.html

---

[3] Email from Chris Mah:
This is an informative email about the seriousness of the problem. Christ mentions that the best solution for ths problem is not to have the problem. But in cases that this cannot be avoided he dicusses some techniques that can be used. He also provided two useful references in discussion of pathological gait.

From cmah@cns.ucalgary.caMon Sep 9 11:30:12 1996

Date: Wed, 28 Aug 1996 17:47:27 -0230
From: Chris Mah <cmah@cns.ucalgary.ca>
To: joshua@engr.mun.ca, manuel@cns15.cns.ucalgary.ca,
 cmah@cns15.cns.ucalgary.ca
Subject: missing markers

Dear Joshua:

Despite what some people may tell you, the problem of missing markers is a serious one. When data are missing, there is no good way to rectify the problem, because any method you use involves constructing estimates of data you dont have. For highly pathological gaits, it may occur unpredictably and often. Clever methods to replace missing data are usually not worth the effort, and only make matters worse.

So the first line of defence against missing markers is not to have the problem. This means having more cameras if possible, to increase the probability that at least two can see them all. In addition, you need to think about simplifying your data collection protocol (if that is acceptable) so that the missing marker problem does not occur.

Having said this, there are different degrees of seriousness for this problem. When there is a short (say < 100ms), isolated gap, in the sense that there is a stretch of good data on either size of it, there is no problem filling in the gap with a straight line or polynomial algorithm. When there are longer gaps, or stuttering intervals of missing data it is helpful to have some kind of proxy marker which might allow you to infer the position of the missing marker.

We often have this problem with the hip during sagittal view recordings of normal walking, because it is covered by the hand during normal arm swing. To deal with the problem we have our subects wear a tail piece of 10 - 15 cm in length attached rigidiy to a belt. This remains in view when the hip is covered. Assuming that the subjects are walking in a straight line, there is a fixed spatial relationship between the hip marker and the tail, and the position of the hip during relatively short gaps (say 500ms) can be inferred.

A slightly different, but related strategy (which I can suggest but have not actually employed) is to extrapolate along a straight line between two markers. For example, if you have trouble seeing an ankle marker, put two markers on the lower leg, so that the ankle is (say 10cm) further along a straight line drawn between the two markers.

As far as your analytical problem goes, the best technique depends on what scientific question you want to ask. However, if you are interested in unpredictable pathological gait, you may want to take a look at our methods papers.

1. Quantitative analysis of human movement synergies: constructive pattern analysis for gait (1994). C.D. Mah, M. Hulliger, R.G. Lee and I. O'Callaghan. Journal of motor behavior, 26, 83-102.
2. Quantitative Analysis Techniques for Human Movement: Finding Multivariate Patterns in Large Data Sets C.D. Mah, M. Hulliger, R.G. Lee and I. O'Callaghan. In M. Witten and D. J. Vincent, eds. Computational Medicine, Public Health, and Biotechnology: Building a Man in the Machine, Part II, pp. 1056-1069. World Scientific Press, 1996.

I hope you find this helpful.
Chris Mah

---

[4] Email from Bill Keezel:
Mr. Keezel mentions some packages that will interpolate missing gaps in data. He says that will small gaps and slow moving subjects curve fitting is an acceptable method for approximating the gaps.

From tsrhgait@ix.netcom.comMon Sep 9 11:30:09 1996
Date: Wed, 28 Aug 1996 17:2:03 -0230
From: TSRH Gait Lab <tsrhgait@ix.netcom.com>
To: Joshua Swamidas <joshua@engr.mun.ca>
Subject: Re: Some Help please - gait anlaysis

Hello Joshua,
Yes, there are a few software packages out there that will interpolate over those missing gaps in your trajectory information. However, the accuracy of interpolation is limited by the gap size and the subject's walking velocity. Also, the applications I'm thinking of utilize C3D files for evaluating a subject's kinetics and kinematics. So, what software are you using for your data reduction and what is the range of your gap sizes of missing data?

If your gap size is within 10 frames for a camera system collecting at 60hz and a slow moving (walking) subject, then curve fitting is an acceptable method for approximating that gap. There are three applications that we utilize in our lab for interpolating over gaps: AMASS(Adtech Motion Analysis Software System), Vicon Clinical Manager (Oxford Metrics), and EVENTS (developed at the NIH, contact Steven Stanhope at sstanhop@cc.nih.gov for more details).

Looking forward to your reply,
Bill Keezel
tsrhgait@ix.netcom.com
Texas Scottish Rite Hospital Gait Lab
ph (214)559-7580
TSRHGAIT@ix.netcom.com

---

[5] Email from Xudong Zhang:
He mentions that if one has an idea of specific motion profiles then, one can develop curve fitting methods to fill in gaps.

Date: Wed, 28 Aug 1996 13:20:28 -0230
From: Xudong Zhang <xudong@umich.edu>
To: Joshua Swamidas <joshua@engr.mun.ca>
Subject: Re: Some Help please - gait anlaysis

Hi Joshua,
I have some experience with the MacReflex system. That system provides some capability of filling the drop-outs. However, when the discontinuity is too severe, (i.e., missing for a large number of consecutive frames), the filling does not work well. Also, it does not extrapolate. I would imagine

problems calculating centers of rotation from marker data.

2. If the distances between three markers on a segment are known, and the location of 2 of these markers are known, the location of the 3rd marker is constrained to a circle. This might be used in conjunction with an interpolative procedure to provide a better estimate of the marker's location. It could also be used where the marker is only seen by one camera. One camera view constrains the marker to be on a line. The intersection of the line and circle could provide the location.

3. If the data is cyclical it may be possible to "guess" where a reasonable location for the marker might be based on where it and the other two markers were at one period before or after the instant the marker is missing.

I think the joining of marker segments is a much more complex and time consuming problem and well deserves addressing. My guess is that there will not be a single algorithm that will work well in matching disjointed marker paths. I've noticed that in doing it manually I use a variety of techniques to solve the problem quickly. I think the best approach would combine these techniques and then use some sort of artificial intelligence to make a "best guess" at which path belongs to which. Some of the techniques you might want to consider for path matching are below.

Path Matching

1. Sometimes paths can be easily identified by some unique characteristic of their location. For instance, in our lab, superior markers on a subject will always have a greater Z coordinate than inferior markers. Left sided markers will always have greater Y values. Anterior segment markers will have greater X values, although this may not be true for markers on the lower extremities as one foot passes the other. This will in general be true for all of our gait analyses, however, in other activities it may not be true at all (e.g. high jump or back flip). Ideally, your algorithm will be more useful if it is applicable to any activity, not just gait.

2. Generally when a marker is lost and then picked up again the drop out will be for a short period of time. The direction and velocity of the two paths will tend to "point" to each other in each of the 3 dimensions if they are really one path. This may not be true however if there are sudden reversals in direction of the marker.

3. If you can reconstruct the marker position using some of the techniques above, you should be able to use that information to find where a marker should be. If you have a data segment that is similar to the reconstruction, it may be a match.

My opinion is that for the marker reconstruction algorithm and the data matching to be really useful it should be applicable to many activities and make as little assumptions about lab coordinates and how the data is "supposed" to look as possible. Ideally the algorithm would be smart enough to make some generalizations about the data from the data itself and not depend on a human telling it what assumptions to make. I also think there will be a need for some human oversight of the algorithm's actions to correct mistakes it will make. This is a project that I think is very worthwhile and would save a lot of time should you develop an algorithm that worked well. I would be very interested in hearing of other responses and hearing of any progress you make.

Good Luck,

Kenneth S. Olree, MS
BJC Human Performance Laboratory   phone: (314) 454-7592
Barnes-Jewish and St. Louis   fax : (314) 454-5500
Children's Hospitals
4555 Forest Park Parkway   kolree@roadrunner.carenet.org
St. Louis, MO 63108, USA

---

[7] Email from At Hof:
Mr. Hof also confirms that with his ELITE system the missing markers are a big problem.

Date: Thu, 29 Aug 1996 06:09:57 -0230
From: "A.L.HOF" <a.l.hof@med.rug.nl>
To: Joshua Swamidas <joshua@engr.mun.ca>
Subject: Re: Some Help please - gait anlaysis

Dear Joshua,

> been able to find any information to show if this really is a problem
> other than the words of a few people who work in gait laboratories.

I am working, sometimes, with the ELITE system, a version of some 5 years old. Missing markers are really a big problem there: when they are missing for more than a few frames, the processing does not go further. Manual correction is even not possible.

With two camera's all the time both cameras should view all markers. This is really a big problem in any kind of real-world movement. Some people here tried to get data for the arm movements from crawling babies. This was very discouraging.

> In many of the vision systems, when the vision system loses sight of the
> markers the tracks of the marker paths becomedisjointed. Generally the
> user has had to connect the disjointed tracks manually. Is there software
> that makes these connections automatically? If so how do they do it?

Part of the solution would be a set-up with more than two cameras, in which for each marker data from those two cameras are used that have the best view on that particular marker.

I am very interested in the responses. Particularly I would like to know whether any of the manufacturers has solutions for this problem. Free software would be even more handy, of course..

Greetings,
At Hof
Department of Medical Physiology
University of Groningen
Bloemsingel 10
NL-9712 KZ GRONINGEN, The Netherlands
Phone: (31) 50 3632645
Fax: (31) 50 3632751

[8] Email from Mark Geil:
He comments that interpolation is made easier when the missing marker information can be copied from a different stride (cycle) of the same trial. However when using assistive devices in pathological gait usually the marker is missing for the same portion of each gait cycle.

Date: Thu, 29 Aug 1996 11:02:59 -0230
From: "Mark D. Geil" <Geil.1@osu.edu>
To: joshua@tera.engr.mun.ca
Subject: Missing Markers

Regarding your question on Biomch-L,
You are correct in your thought that assistive devices common with pathological gait can result in missing markers. This presents a problem, because the marker in question is usually missing for the same portion of each gait cycle. Interpolation is made easier when the missing marker information can be copied from a different stride of the same trial. However, when a crutch or walker obstructs a camera's view of a marke, it typically does this at the same portion of each stride. VICON's software will perform interpolation as part of the AMASS package upon marker identification or as part of the Vicon Clinical Manager package upon processing gait cycles.
Best wishes,
Mark Geil
Ohio State University Gait Analysis Laboratory, Columbus, Ohio
(614) 293-4832

[9] Email from Kevin Arthur Ball:
Mr. Ball writes that the missing marker problem is pretty serious. He provides some references for the automated analysis of gait and for rigid body kinematics (which I found useful).

Date: Fri, 30 Aug 1996 11:25:21 -0230
From: Kevin Arthur Ball <BALL@phe.utoronto.ca>
To: Joshua Swamidas <joshua@engr.mun.ca>, joshua@engr.mun.ca
Subject: Re: Some Help please - gait anlaysis

Hello Joshua

First some comments about your email. You have written that "you are a maters student", I assume by this that you are concerned with all that matters! In the future I would suggest that you drop this prefix altogether. Your question can stand on its own, and it is a good one.

"I am looking into the problem of missing markers and the ability to track them and predict their path while they are out of view of the vision systems. I have come across some interesting techniques. Unfortunately, I haven't been able to find out how big this missing marker problem really is."

In my estimation this problem is far more pervasive than has even been realized. Have you ever wondered why there are virtually no complete 3D kinematic studies of gait or running that includes run-and-cut maneuvers, turns or pirouettes. Each would certainly be interesting to study but given the present state of our technology very few researchers would ever be willing to attempt them. Instead we tend to restrict gait analysis to straight line walking and use as few markers as possible so as to avoid "marker collisions".

"In many of the vision systems, when the vision system loses sight of the markers the tracks of the marker paths become disjointed. Generally the user has had to connect the disjointed tracks manually.

Is there software that makes these connections automatically? If so how do they do it?"

Some years ago Michael Pierrynowski and I were working with a vision system similar to that which you have described. At that time we decided to circumvent the vendors software, so we designed our own system for data processing. This took the better part of a year to write. It is described in the following:

Ball, K.A. and Pierrynowski, M.R. (1991). WalkTrak: Automated analysis of 3D kinematic data from video systems. Proceedings of the International Symposium on 3D Analysis of Human Movement, First Meeting, 28-31 July. Montreal, Quebec, Canada, 6-9.

More recently, we have switched to an activer marker system, thus our previous efforts have been shelved as of late.

If you are willing to explore the use of rigid body methods for kinematic measurements, then I think you will may find the answer for the marker drop-out problem. I could provide a long list of references for you but to save myself some time you will find many of these in the following:

Ball, K.A. and Pierrynowski, M.R. (1995). Estimation of six degree of freedom rigid body segment motion from two dimensional data. Human Movement Science, 14, 139-154

Ball, K.A. and Pierrynowski, M.R. (1996). Classification of errors in locating a rigid body. Journal of Biomechanics. Any Day Now.

I hope this helps!
Bye for now, Kevin.
P.S. I hope all of you at Memorial have a good 500th anniversary of Cabot's sailing, eh!

Kevin Arthur Ball, B.P.H.E., M.Sc.
Director, Biomechanics Laboratory
Sport Medicine / Biomechanics Group
School of Physical and Health Education
University of Toronto
320 Huron St.
Toronto, Ontario M5S 1A1, CANADA
Voice (416) 978-3196
Fax (416) 978-4384
ball@phe.utoronto.ca

---

[10] Email from Hakan Johnsson, QUALYSIS:
He commented on the complications of the problem of tracking and discusses some of the techniques

ExpertVision HiRES system does provide tools for filling in the gaps in marker tracks. With short gaps of no more than a few frames a software (Track Mender) looks at the continuity of the 3D math segments (and a few other things) and automatically joins the segments. Another method is one that involves human intervention to help draw a curve over the gap.

Date: Wed, 28 Aug 1996 14:44:10 -0230
From: John Greaves <John.Greaves@MotionAnalysis.com>
To: Joshua Swamidas <joshua@engr.mun.ca>
Subject: Re: Some Help please - gait analysis

Hi Joshua,

I'm John Greaves, founder and Sr. VP at Motion Analysis Corporation in California. Here are some ways that our ExpertVision HiRES video motion capture system deals with the missing marker problem you mentioned. First of all, gait is not too difficult to "track" (which means reduction to 3D coordinates from the raw camera data in our parlance) in the grand scheme of what people are doing with motion capture systems. What is more difficult is collecting data from two or three persons, each with 30 or 40 markers to accurately represent their limb segment motions. Anyway, pathological gait we also have a lot of experience with (the OrthoTrak gait analysis software has been on the market since about 1987) and pathological gait is more difficult to track than normal subjects, as you know.

The best solution is to have redundant cameras for picking up more marker positions from more camera locations. A minimum of 4 cameras (two for each side) is required for markers placed around the body, although the folks at Newington Children's hospital used 3 cameras for a long time by designing a special marker set with markers only on the front of the body. As you add more cameras, the more redundancy you have in your data collection mechanism and the fewer holes you will have in your data. We sell mostly 6 camera systems (roughly 3 cameras per side) for gait analysisnow, and the data can be tracked with almost no "holes" in the 3D data set. BUT if there are holes, the mechanisms we have are:

1-The Track Mender, which looks at the continuity of 3D path segments and automatically joins them if they are no more than a few frames apart and meet some equations about continuity in the XYZ components and their first derivatives. If you don't like the results, you can Undo it.

2-The Add button in the XYZ time-series editor. Here you can position the mouse in a hole in each component (X, Y and Z) and place a point. The software splines across the gap with your input and you confirm each component separately. It turns out that the human is very good at seeing what "looks right" from this XYZ editor view--much better than what you can "see" from a stick figure or 3D spatial view of your data.

Anyway, I hope this helps. We have spent years refining the tools for accurate data collection, tracking and editing and we were just at the Canadian Society of Biomechanics meeting, but that was on the opposite end of Canada from you.

Sincerely,
John Greaves, Ph.D., Sr. Vice-President
Motion Analysis Corporation

[13] Email from Georger Roundanez (VICON):

Date: Fri, 23 Aug 1996 12:37:08 -0230
From: Georges Roudanez <georges@viconsys.com>
To: joshua@engr.mun.ca
Subject: Re: mailto:georges@viconsys.com

>1) How frequently do these holes (missing data) occur say, with a 2, 3
> and 5 camera system and in studying pathalogical walking? The system
> that will be purcahsed will be used maily wth CF kids and for research

Joshua,

This is a difficult question to answer. The lost trajectories occur as a result of a marker not being seen by at least 2 cameras (in order to provide xyz coordinates). The more cameras, the placement of those cameras and the type of movement being measured determine the opportunity to avoid occlusions (not seen markers). A simple move with two cmeras being constantly in view of the markers will most likely result in no occlusions.

>2) Do you have a way of predicting the marker motion?
The way the Vicon operates it is possible to determine the anitcicipated marker path (trajectory).

>3) Does the software automatically connect disjointed marker tracks or do
> you let the operators handle the connecting of incomplete tracks.

The answer is yes to both. You may choose to have the software (BodyBuilder - a special package that has editing and modelling features) automatically fill gaps, delete and fill noise spikes and interpolate or smooth the data.

[14] Email from Jermy Wise (Ariel):
Dr. Wise comments on the general problem of missing markers and makes comments on how their software handles the problem. They seem to use linear extrapolation of some kind and also some smoothing functions to fill in the shorter gaps.

Subject: Re: missing marker solution
Date: Mon, 9 Sep 1996 18:25:52 -0230
From: Jeremy Wise <jwise@oitunix.oit.umass.edu>
To: joshua@tera.engr.mun.ca
CC: ariel1@ix.netcom.com

Joshua-

Dr Ariel has asked me to respond to your questions regarding handling of incomplete tracks with the APAS system. I'll do my best.

>1) How frequently do these holes (missing data) occur say, with a 2, 3

> and 5 camera system and in studying pathalogical walking? The system
> that will be purcahsed will be used maily wth CF kids and for research purposes.

It's impossible to say how often this occurs because it depends entirely on the activity under consideration. Generally the more rotation about a vertical axis the more often a marker will become obscured. The more cameras one has the more likely atleast 2 cameras will see a marker for all frames. If 2 or more cameras "see" marker then the software can calculate the 3D coordinates.

>2) Do you have a way of predicting the marker motion?

The program predicts marker motion using a rather complex search algorithm. The location the software starts searching at is an extrapolation of the previous 2 frames or previous frame, a software selectable switch. The reason a simple linear extrapolation is used is because higher order extrapolations are more error prone.

>3) Does the software automatically connect disjointed marker tracks or do
> you let the operators handle the connecting of incomplete tracks.

I'm not sure if you are referring to tracks in 2D or reconstructed 3D. If you are talking about the plane of the video, the software can be set to skip and flag as MISSING any markers which the software can't find, or the software can be configured to pause & let the user estimate marker location based on his/her observation of the whole image. If you are talking about 3D, then the program flags any point that cannot be seen by atleast 2 cameras as MISSING and will be considered as missing RAW data. However, when it is smoothed, and incomplete sections of data will be interpolated then smoothed with the selected algorithm. What you are asking the software to do is to make data where it can't be seen be at least 2 cameras. This is risky business.

I hope this helps,
Dr. Jeremy Wise
Dir R&D

APPENDIX B

FLOCK OF BIRDS PICTURE

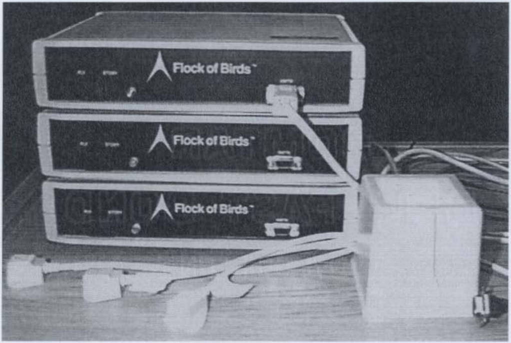

Figure B.1 Flock of Birds system

APPENDIX C

NOISE IN CCD IMAGES

C.1 Introduction

A Charge Coupled Device (CCD) is an array of light-sensitive silicon cells. Each cell produces a signal that is converted into a number representing a pixel in the digital image. The "signal" refers to the number of photons recorded by each cell. Ideally each cell in the CCD-chip is able to detect light corresponding to everything between zero and 278 578 electrons (Walmann, 1996). However, all circuitry generates undesirable noise so that even if the CCD camera is placed in a no light environment the pixel values would vary unpredictably. Since this random part of the signal cannot be reproduced in any way, it can never be removed from the signal. By definition, this unpredictable variation is noise. This and other sources of noise are classified into four categories which are, for the most part, independent of each other and combine to degrade the image quality.

Readout noise. When the accumulated charges are shifted on the CCD chip, electrons may be left behind or may jump ahead; these fluctuations together with noise from the pre-amplifier are called readout noise. This can be reduced but not removed completely.

Dark count. Even in the absence of light, electrons accumulate in a CCD, and this signal is indistinguishable from one produced by light. The rate at which this dark count is produced decreases as the temperature of the CCD decreases. It is even possible to measure and correct for the mean value of the dark count, but the noise component of the current cannot be removed.

Background noise. Light pollution from background sources contributes to the signal collected by a CCD. The process of a photon striking a CCD does not guarantee that it will produce a count. This is due to the quantum nature of light (not discussed here). The photon detection event is considered to be a Poisson event (Newberry, 1996). The statistical nature of photon counting makes the exact value of the signal uncertain. Because this background adds photons, and because all photon measurements have an inherent uncertainty, it is difficult to remove it from the signal. There are ways of reducing, but not completely eliminating background noise.

Processing noise. Basic image processing such as using filters or subtracting dark frames involves combining uncertain numbers with other uncertain numbers on a pixel-by-pixel basis. This results in an increase in the amount of noise in each pixel.

C.2 Experiment

The intricacies of the camera were not considered in the thesis since a model camera was calculated and used. However, in order to test the Epipolar matching and 3D reconstruction algorithms, a routine was written to simulate a circle target on a camera image plane at a particular location and white noise was added to simulate the uncertainties in the measurement and the error (RMS) in pixels was calculated.

The target was created using a Gaussian distribution curve with varying variances to simulate varying reflection coefficients of spherical markers. The sizes of target were used by specifying a radius from the center of the distribution. The target size that was tested represented approximately 1% of the total size of the CCD grid. The target was placed in an arbitrary location on the grid and the white noise was added. The percentage of white noise added to test the algorithms were 5, 10 and 50 percent of the actual signal intnsities.

Two techniques were used to determine the center of the target. The first technique, the centroid method, calculated the center of mass of the whole image and assumed that the noise was low enough to not affect the center of the target. The second technique used, the edge detection method, used the centroid method to find the approximate location of the target and refined the center values by calculating the edge vectors along the surrounding areas and finding the center point of those edges. Tables C.1 shows the RMS error values of the calculated pixel values from the expected locations.

1% Size Target							
Var	Type	5% Error		10% Error		20% Error	
		x	y	x	y	x	y
5	Centroid	1.288	1.304	1.984	1.967	2.623	2.609
	Edge	0.005	0.005	0.104	0.0963	0.241	0.3317
10	Centroid	1.184	1.179	1.839	1.802	2.52	2.59
	Edge	0.004	0.003	0.008	0.007	0.027	0.022
20	Centroid	1.11	1.121	1.732	1.757	2.407	2.555
	Edge	0.003	0.003	0.005	0.005	0.03	0.011

Table C.1 RMS values of the 1% circle target

The table shows an increase in the RMS values as the error increases. The error from the centroid method was larger than the edge detection method. This was expected since the edge detection method used the centroid location as the starting point for determining the real center point. One other trend to note is that the error decreases as the variance increases. This was also expected because the larger variance meant a wide Gaussian curve and therefore a more flat target.

The edge detection entry with variance of 10 in Table C.1 was chosen to be used in determining the successes of the epipolar and reconstruction algorithms.

References:

Newberry, M. V. (1996). *The Signal to Noise Connection*, CCD Astronomy, Summer 1994

APPENDIX D

ARIEL WEB PAGE EXCERPTS

Ariel Dynamics Worldwide
Online Reference Library

Science serving Industry, Sports and Human Performance

The Ariel Performance Analysis System
Three Dimensional Video Analysis for Orthopedic Evaluation

The Ariel Performance Analysis System (APAS), the world's most advanced computerized system for the study of human movement, offers a comprehensive dynamic study of patient performance. The data gained from these studies can aid the orthopedic physician by quantifying deficiencies in functional movements.

PRE-SURGICAL ANALYSIS

The APAS can be used to measure and calculate ranges of motion, joint angle, velocity, acceleration, and force. This information helps to identify and document a patient's biomechanical performance as affected by injury or ergonomic inefficiencies.

Kinematic evaluation using the APAS before surgery procedures provides an opportunity to fully evaluate a disability and how it affects functional performance. Pre-surgical analysis also allows the documentation of a patient's impairment for procedural justification, as well as establishing a patient's condition prior to surgery or treatment.

SURGICAL ANALYSIS

The APAS can provide accurate information as an aid in determining such things as the proper positioning, of implants and surgical procedures as they affect joint movement and future range of motion.

POST-SURGICAL EVALUATION

Data supplied by the APAS may be used in both evaluating surgical results and planning rehabilitation..

REHABILITATION TRACKING AND TREATMENT JUSTIFICATION

Because APAS data can be used to objectively measure patient progress, improvement and rehabilitation protocols can be evaluated and documented These analyses are powerful additions to a medical history from which to plan further patient treatment

AREAS OF APPLICATION

The APAS can be used to evaluate and study the level of impairment in functional movement due to:
A. Upper Extremity Dysfunction

1. Shoulder Dysfunction
    a. shoulder extension-flexion abnormalities
    b. shoulder abduction-adduction abnormalities
    c. shoulder internal-external rotation abnormalities
    d. other shoulder abnormalities
2. Elbow Dysfunction
    a. elbow extension-flexion abnormalities
    b. elbow supination-pronation abnormalities
    c. ther elbow abnormalities
3. Wrist Dysfunction

LIFT TASK

The APAS can be used to calculate all critical forces on the spine during a lift analysis.

GAIT STUDY

The APAS can be used to give comprehensive information on the contribution of each body part during gait.

HANDSTUDY

The APAS can be used to perform detailed studies of any part of the anatomy

◄back | Index | next►

TOP APAS ACES

When transformation is complete, a smoothing or filtering procedure is performed on the image coordinates to remove small, random digitizing efforts and to compute body joint velocities and accelerations. Smoothing algorithms include polynomial, cubic and quintic splines, as well as various filters. At the completion of smoothing, the true three-dimensional body joint displacements, velocities, and accelerations have been determined on a continuous basis throughout the duration of the sequence.

At this point, optional kinetic calculations can be performed to complete the computation phase. Body joint displacements, velocities and accelerations are combined with body segment mass distribution to compute dynamic forces and moments at each of the body joints. Muscular contribution to these forces and moments can then be computed by selectively removing the inertial and gravitational kinetic components.

The **presentation phase** of analysis allows computed results to be viewed and recorded in a number of different formats. Body position and body motion can be presented in both still frame and animated stick figure; movements in three dimensions. Results can be reported numerically in tables, exported to external mediums such as spreadsheets, and graphically. Plots of body joints and segments, linear and angular displacements, velocities, acceleration, forces and moments can be produced in a number of format options.

The preceding discussion has illustrated the use of biomechanical quantification of movement analysis in assessing functional capacity. The technique can be performed with papers, pencils, erasers and large amounts of time or with newer, enhanced technologies incorporating computers and integrated hardware-software. In addition to PC desktop computerized biomechanical systems, it is now possible to perform most of these procedures with a portable, 2 Kg. notebook computer. With the technological innovations of the future, there would seem to be no limit except, perhaps, human reluctance.

While the biomechanical assessment technique just discussed measures functional capacity, it can also be measured directly by resistive dynamometry devices. An ideal system would employ computerized feedback control of both resistance and movement during the exercise. This intelligent dynamometer, would allow the machine to dynamically adapt to the activity being performed rather than the traditional approach of modifying the activity to conform to the limitations of the machine. With this type of equipment, the coach could examine the results of the biomechanical motion analysis and, with his or her knowledge and experience of the sport and the individual athlete, determine the most appropriate training regimen at that time for that person. Concentration on strength acquisition may be important during the off season while speed and strength maintenance are paramount within the competition period. These choices can be made and subsequently modified by the coach.

Case studies in applied biomechanics demonstrate the importance of considering the true patterns of motion in determining efficient performance. One of the most important parameters in training is the ability to allow the performer to achieve a movement pattern of resistance or the pattern of motion experienced by the user during the actual activity. The ability to modify the pattern by reprogramming the dynamometer can be determined by the individual. Standard isokinetic equipment cannot fulfill these requirements.

The value of applying the principles of biomechanics to the assessment of functional performance has been demonstrated. Movement analysis provides the means to quantity human activity and to provide insight into the mechanisms that contribute either to superior or inferior levels of performance. In addition, a technology has been presented that permits exercise and rehabilitation patterns to biomechanically duplicate the target activity as measure of function capacity

◄back | index | next►

TOP APAS ACES

Copyright © MCMXCVI by Ariel Dynamics
Web Site Design by George Capris