A Generalized Joint Motion Analysis System Using Borland C++ For Windows

by

Janice Dale Miller

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements for the Degree of
Master of Science

University of Manitoba
Department of Electrical and Computer Engineering

Winnipeg, Manitoba

April, 1997

THE UNIVERSITY OF MANITOBA

FACULTY OF GRADUATE STUDIES

*****

COPYRIGHT PERMISSION PAGE

A GENERALIZED JOINT MOTION ANALYSIS SYSTEM USING BORLAND C++ FOR WINDOWS

BY

JANICE DALE MILLER

A Thesis/Practicum submitted to the Faculty of Graduate Studies of The University of Manitoba in partial fulfillment of the requirements of the degree

of

MASTER OF SCIENCE

Abstract

The University of Manitoba Motion Analysis System (UM^2AS) has been developed to study upper limb motion. Although upper limb motion is of great interest to clinicians, due to its involvement in many daily activities of living, other motions are also of clinical importance. The current work undertakes the adaptation of the UM^2AS system to a generalized system that allows the examination of whole body motion.

The human body is modeled as 9 rigid-body segments, Head/Neck, Thorax, Pelvis, Thigh, Calf, Foot, Upper Arm, Forearm, and Hand, attached together at 8 joints, C7-T1, Lumbar Sacral, Hip, Knee, Ankle, Shoulder, Elbow, and Wrist. Effectively, one side of the body is modeled. Each joint is allowed 3 degrees of freedom (DOF) and modeled with a spherical joint model. Translation joint motion is not considered.

The software implementing the system was written in Borland C++ for Windows with a standard Windows format. Euler angles, with a z-x-y sequence, are used to find the rotations in the joints. A marker generation program was also developed to generate theoretical marker positions for testing the system. Using these theoretical marker positions, the relative error in calculated Euler Angles ranged from -2.5% to 1.1%, mean -0.001, standard deviation 0.105.

Acknowledgments

First I would like to thank Dr. Ed Shwedyk, my advisor, for his patience and support over the years. His dedication as a teacher and his commitment to his students has, for a large part, inspired my continued effort on this project. Dr. Juliette Cooper provided encouragement and enthusiasm that was very motivating. Her careful reading of the manuscript, and many suggestions, have been invaluable. Janice Freeman encouraged me to continue against my better judgment and Kris Corbett dropped by the lab often and offered me late night rides home. Zahra Moussavi has patiently answered my many questions in the last few months. Dr. Janine Johnston agreed to give me a leave of absence from my work, so that I might finally finish. Thanks to you all.

Albert and Shirley James have always been willing to lend a helping hand. Without their support I might not have finished my undergraduate degree, let alone this thesis. Thank you, you are greatly appreciated. Darrell James has been a constant source of support and encouragement. Thanks for picking up all the slack.

Finally, I would like to thank the Rehabilitation Centre for Children in Winnipeg for financial support of this project.

Table of Contents

List of Tables

List of Illustrations

Nomenclature

1.	**DOF**	**Degrees of Freedom**
2.	**TDC**	**~~Three dimensional coordinate file~~**
3.	**UM^{2}AS**	**University of Manitoba Motion Analysis System**
4.	**FFR**	**Fixed Frame of Reference**

Chapter One
Introduction

1.1 Preface

Quantification of human joint motion has been studied for over a century. One of the earliest studies, done by Braune and Fischer in 1891, attempted to quantify six-degrees of freedom joint motion [Kinzel and Gutkowski, 1983]. Most of the ensuing examinations of joint motion have been limited to simpler models, typically one, two, or three degrees of freedom. There have been several attempts to measure six-degrees of freedom, however since these joint models are difficult to understand in "anatomical" terms, they have not gained widespread use.

Over the years various techniques for measuring joint motion have been developed. Simple one-degree of freedom joint models have been measured using photographs, x-rays, cinegraphs, and of course the single axis goniometer favoured in clinical use.

More complex two and three-degree of freedom joint models require more complex measurement techniques. The most enduring method to date is the stereometric technique. Indeed, this was the method used by Braune and Fischer in 1891 and is still used today in various more technologically advanced manifestations. Stereometric techniques employ two or more image capturing devices, such as cameras or x-ray sources, placed to give different perspectives. Each camera records the image on a two dimensional plane such as film. From the two different images and the geometric

properties of the recording environment, it is possible to reconstruct the three dimensional space that has been recorded. Customarily, only markers representing the limb segments surrounding the joints or joint of interest are reconstructed. The study performed by Braune and Fischer used still cameras to capture the two dimensional flat images. Today it is much more common for stereometric techniques to employ the use of video cameras, although other methods such as radiography, cinefilm, and sonic digitizers also appear in the literature[Small, et al, 1994, Youm and Yoon, 1979].

One other measurement technique that is not as commonly used is the method of instrumented linkages [Kinzel, et al, 1972]. Instrumented linkages are mechanical devices designed to be strapped to the body above and below the joint. The movement of the linkage during joint motion defines the joint motion. Perhaps the difficulty of use, coupled with the weight of the linkage, has lead to the infrequent use of this method. However, some variation of instrumented linkages appears to be the most common method of defining all six-degrees of freedom occurring during joint motion.

1.2 Problem Statement

Many joint quantification techniques and systems have been developed. Each system has inherent strengths and weaknesses. One of the most consistent weaknesses is the inability of the measurement technique to be applied to more than one joint simultaneously. Systems that allow the analysis of more than one joint are often designed for a particular group of joints, for example the upper limb or the spine. Usually because of the requirement for a specific marker placement, these systems are not easily adaptable

to the other joint combinations. In some systems employing instrumented linkages, it is not even possible, in all people, to measure the joints the linkage is designed for. For example, it would be impossible to measure the rotations in the lower limb of a child with a mechanical linkage system developed for an adult, due to the difference of the size of the joints and the limb segments.

It is, therefore, very desirable to develop a system of joint measurement that is flexible both in the number of joints that can be examined and in the population sample that can be studied.

1.3 Literature Review

There are many studies that have examined joint motion. The techniques of measurement are varied and direct comparison of results is difficult. A brief review of the literature in this field is presented in this section.

Kinzel, Hall and Hillberry [Kinzel, et al, 1972] at Purdue University are one of the few groups who have attempted to quantify six-degree of freedom(DOF) joint motion. Their method employed a 6-DOF spatial model which reduces joint motion to rotation about and a translation along a mathematically determined "screw axis". The measurement technique was an instrumented spatial linkage that allowed for the examination of a single joint. The major limitation of the joint model used in this study was the difficulty of interpretation. The translations and rotations in the joint are not presented in a method easily understood or applied by clinicians.

Blacharski, Somerset, and Murray, from Syracuse University, [Blacharski, et al, 1975] examined the three-dimensional motion of the knee joint in five cadaver joints. They found that the motion of the knee is not planar and that the sliding gliding motion in the knee is due to bone geometry and not to the cruciate ligaments as had been earlier proposed. This group tried several measurement techniques and settled upon a strobe light and time lapse photography. A mathematical technique, Reuleaux's method, was used to find the centre of rotation of the tibia as it was rotated about the femur; the location of the instantaneous axis of rotation was quantified throughout the range of motion.

Another examination of the knee joint was carried out several years later by Soudan, Van Audekercke and Martens [Soudan, et al, 1979] at the University of Louvain in Belgium. The Reuleaux method was examined and this time determined to be an unsuitable method for determining the centre of rotation in the knee joint. This group determined that the screw axis or "instant axis" proposed by Kinzel [Kinzel, et al, 1972] was a more precise technique.

Soudan's group were concerned with plotting the instant axis as a representation of the pathologies of a knee joint. A comparison was made between the instant axis pathways of a normal knee to those of several pathological knees. The form, as well as the place, of the pathological pathways differed significantly from the normal pathway and from each other.

Around the same time as the group in Belgium were studying the knee joint, Youm and Yoon, [Youm and Yoon, 1979] attempted to quantify three dimensional wrist joint

motion. They modeled the wrist as two rigid body segments , hand and forearm, with an unknown mechanism between them. Youm and Yoon examined several measurement techniques and finally settled on the three dimensional sonic digitizer method. They used 3 non-collinear sources on each rigid body segment and found the relative rotations using Euler angles in the y-z-x rotational sequence. X-rays of the wrist were used to relate the mechanical analysis to the bony landmarks and to locate the instant centre of rotation for flexion-extension and radial-ulnar deviation. The measurement system was verified using a mechanical joint before cadaver specimens or live subjects were used.

The approach to the measurement of joint motion taken by Chao [Chao, 1980] at the Mayo Clinic in Rochester was to employ a triaxial goniometer. The goniometer used was a gyroscopic device and thus, was not sequence dependent as other methods are. Chao claims it was possible to orient the goniometers to represent anatomical motion. The output of the goniometers was compared to classical Euler angle methods on a mechanical model. This examination yielded very small errors between the goniometer measurements and the Euler technique of measuring rotations. The Euler angles were calculated from anatomical landmarks evident in two different x-rays at each joint position. Chao concludes with an argument for the use of triaxial goniometers in a clinical setting where real-time measurements are very desirable. This method of measurement limits the number of joints studied simultaneously to a single joint.

Grood and Suntay [Grood and Suntay, 1983] defined a floating axis system for the measurement of knee motion. This group, from the University of Cincinnati, described a method of calculating displacements that is more readily understood than the traditional

method of "screw axis". In this method, one fixed axis is defined along the long axis of the tibia and another which passes through the centre of the medial and lateral condyles of the femur. The third axis of rotation is called the "floating axis" and is defined as the axis perpendicular to both fixed axes. The translations calculated are defined as translations along the three defined axes of rotation. Grood and Suntay claimed that this method of calculating rotations is sequence independent and can be modeled as a 4-link kinematic chain consisting of cylindrical joints.

In 1983, Kinzel and Gutkowski, at Ohio State University, published a paper reviewing the joint models and measurement techniques in the literature [Kinzel and Gutkowski, 1983]. Included in this paper were descriptions of the six most commonly used joint models: the one-DOF hinge joint, the three-DOF planar joint, the three-DOF ball and socket joint, the two-DOF spherical joint and the six-DOF spatial joint. Various motion measurement techniques were reviewed and discussed in relation to their effectiveness of use with various joint models. Kinzel seemed to feel that instrumented linkages was the most accurate method of measuring relative motion in a joint at the time the paper was written.

Spiegelman and Woo, at the University of California, San Diego, [Spiegelman and Woo, 1987] took a rigid body approach to the method of finding centres of rotations in planar joints. They compared their error analysis to the analytical expressions for the Reuleaux method discussed by Panjabi [Panjabi, 1979]. Spiegelman and Woo discovered similar problems with locating the centre of rotation if the rotation angle was small, however using their method, the error in calculating the rotation angle was significantly

smaller. In addition, the rigid body method had greatly reduced error as the angle between the two markers ranged between 90° and 180°. This reduced error results in the calculations of both the centre of rotation and the rotation angle.

Safaee-Rad [Safaee-Rad, 1987] detailed a stereometric joint analysis system developed at the University of Manitoba. This system was designed specifically for the study of upper limb motion. The system uses two CCD cameras and VCRs to capture two different views of the subject. A reflective marker system, taped to the limb, defines the coordinate systems of each segment. The shoulder and wrist joints were each modeled as a three-DOF spherical joint and a two-DOF spherical joint model was used for the elbow. The Euler method of calculating rotations in the joints was employed with a z-x-y sequence.

The system developed by Safaee-Rad is significant in that it allowed one of the first studies of functional upper limb motion to be conducted [Safaee-Rad, et al, 1990]. Studies done previous to this one had employed the use of uniaxial or triaxial goniometers; it was not possible in such studies to examine simultaneous joint motion of the upper limb.

Pennock and Clark, Purdue University, [Pennock and Clark, 1990] developed yet another coordinate system for the examination of the knee joint. This system was similar to the "floating" axis system to that of Grood and Suntay [Grood and Suntay, 1983]. The differences lie in the location of the fixed axis placement in the femur and tibia. They did include some discussion of the knee modeled as a six-link open chain, although they did not base their coordinate system on this model, their argument being based on the

hypotheses that knee joint translations do not necessarily occur along the same axis as the rotations.

Pennock and Clark compared the effect of coordinate system choice on the measured parameters of the joint model used by Grood and Suntay. The results indicated that, although several different choices resulted in similar values for rotations in the knee, translations varied quite significantly depending on the choice of the coordinate axes.

Small's group, from Queen's University, [Small,et al, 1992] tackled the problem of comparing kinematic literature. They examined the three alignment based coordinate systems: Euler, Cardan (a special case of Euler), and the floating axis system. Their results demonstrated that all three systems give identical results for angular parameters. Small et al emphasized that their findings hold true only if great care is taken in defining the axes of rotation and the sequence of rotations. They also stated that the "screw axis" method can only be compared directly to the alignment based systems when the direction of the "screw axis" is aligned with one of the coordinate axis of the Euler, Cardan, or "floating axis" systems. They conclude with a discussion of the various methods used to define translations. There does not appear to be an easy method of comparing translational measurements between different measurement systems. In fact, examples are given where different methods give vastly (compared to the magnitude of the measured translations) different results.

A later study by the same group [Small, et al, 1994] examined the precision and accuracy of the location of bony landmarks in the hand and the forearm. A stereometric x-ray technique was used to identify the bony landmarks on four cadaver hands, in various

hand positions. Lead markers were implanted and the accuracy of locating the lead markers was compared to the accuracy of locating the bony landmarks for four different observers. It was determined that the precision of locating the bony landmarks was better than 1.1 mm in all hand positions on a subset of suitable bony landmarks. The inter observer accuracy was 2.3 mm. Small's group suggests that these uncertainties must be taken into consideration in subsequent *in vivo* testing.

Wang, et al, [Wang, et al, 1994] another group from Queen's University, detailed a novel six-DOF joint measurement technique. They developed a system using six spring-loaded variable differential transformers (LVDT), each of which provides a measurement of displacement along its mounting axis. A cube is mounted on one of the moving segments of the joint and the LVDT frame is mounted on the other. The LVDT's are attached between the cube and the LVDT frame. This system calculates the rotational angles using classical Euler techniques, although the sequence of rotation was not specified. The system was tested using a digital milling machine bed and appeared to produce accurate results. The advantages of this system appear to be its low cost of implementation and its ability to measure translational motions. The major disadvantage is the limitation of measurement to a single joint. The system was apparently designed to study motions in the sacro-iliac joint, although no data are presented.

The literature concerning the analysis of joint motion is varied and extensive. Many measurement techniques have been used. The common link between them is their application to a single joint or, at best, to a specific series of joints. It is clear that a more generalized approach to the measurement of joint motion needs to be addressed.

1.4 Objective

The objective of this research is to generalise the joint analysis system defined by Safaee-Rad [Safaee-Rad, 1987]. The system designed by Safaee-Rad, the UM^{2}AS system, is specific to the upper limb, although it has been used to study joint motion in the lower limb as well. Much more complicated motion analysis could be conducted if the whole body is modelled. It would then be possible to analyse the motion of the joints in various areas of the human body, perhaps even all of them.

The generalisation was constructed around a fixed orthogonal marker co-ordinate system. This orthogonal marker system is a standard unit that can be placed on each body segment. Theoretically, the co-ordinate system can be placed at any position on the body segment. Practically, the marker system should be placed on each segment in the area where it is most likely to move in concert with the bone of the segment. In addition, the quest for reducing the number of markers used resulted in specific placement requirements for certain body segments.

The body is modelled as a system of eight joints: 1) C7-T1; 2) Lumbar-Sacral; 3) Hip; 4) Knee; 5) Ankle; 6) Shoulder; 7) Elbow; and 8) Wrist. This eight joint model allows only one side of the body to be examined at a time. It is assumed that the body segments between these joints are rigid body segments and that each joint is a 3-DOF rotational joint. Like most joint analysis systems, translational motions are ignored. Rotations in the joints are resolved using the Euler angle technique in a z-x'-y'' sequence.

Testing of the generalised system was performed using a computer testing system. A marker generation program was developed in Borland C++ for Windows to generate the

marker positions for each body segment. This generation program essentially produces the three dimensional marker positions of each co-ordinate system based on rotations specified for each joint. These three dimensional co-ordinates were then used as test data for the generalised UM^{2}AS system.

This document covers the scope of this research project. Chapter Two presents the original UM^{2}AS system as developed by Safaee-Rad and modified by Giles [Giles, 1994]. Also covered is some basic image processing theory describing stereometric techniques used by Safaee-Rad to produce the three-dimensional co-ordinates required by the generalised UM^{2}AS system developed for this project.

Chapter Three describes software developed in Borland C++ for Windows for this generalised motion analysis system. Presented is both the theoretical basis for the generalised UM^{2}AS software as well as an examination of the software itself. The flow of data through the developed programs and the user interfaces are described.

Chapter Four presents the Gen_Mark system for generating marker locations based on specified rotations for each joint. This chapter also includes a section on the theory behind the testing procedure as well the software developed as a result. The final section examines the results of the testing procedure when generated markers are run through the generalised UM^{2}AS system.

Conclusions and suggestions of further improvements to the generalised UM^{2}AS system are presented in Chapter Five.

Chapter Two
Original UM²AS System

The UM²AS system employs video cameras and recorders and a computer to capture and analyze three dimensional motion of the upper limb. Generally speaking, this is done by instrumenting the upper limb with markers, recording the motion in stereo using two cameras, digitizing the video images and processing the data.

When doing the data processing, software assumes a three segment limb model: upper arm, forearm, and hand. The joint model used for the shoulder and wrist is a three degree of freedom rotational model. A two degree of freedom joint model is used to represent the elbow joint. Although the system was originally designed to study upper limb pathologies, it has also been used to study the sit-to-stand motion in children with cerebral palsy. In this case, the lower limb was modeled as a three segment limb, with three degrees of rotation at the hip, two at the knee, and three at the ankle/foot.

2.1 System Hardware

The hardware components of the UM²AS system includes a PC 486 computer, a PIPEZ video digitizing board, three super BETA VCRs, three CCD cameras, and three monitors. The general laboratory configuration is represented in schematic form in Figure 2.1.

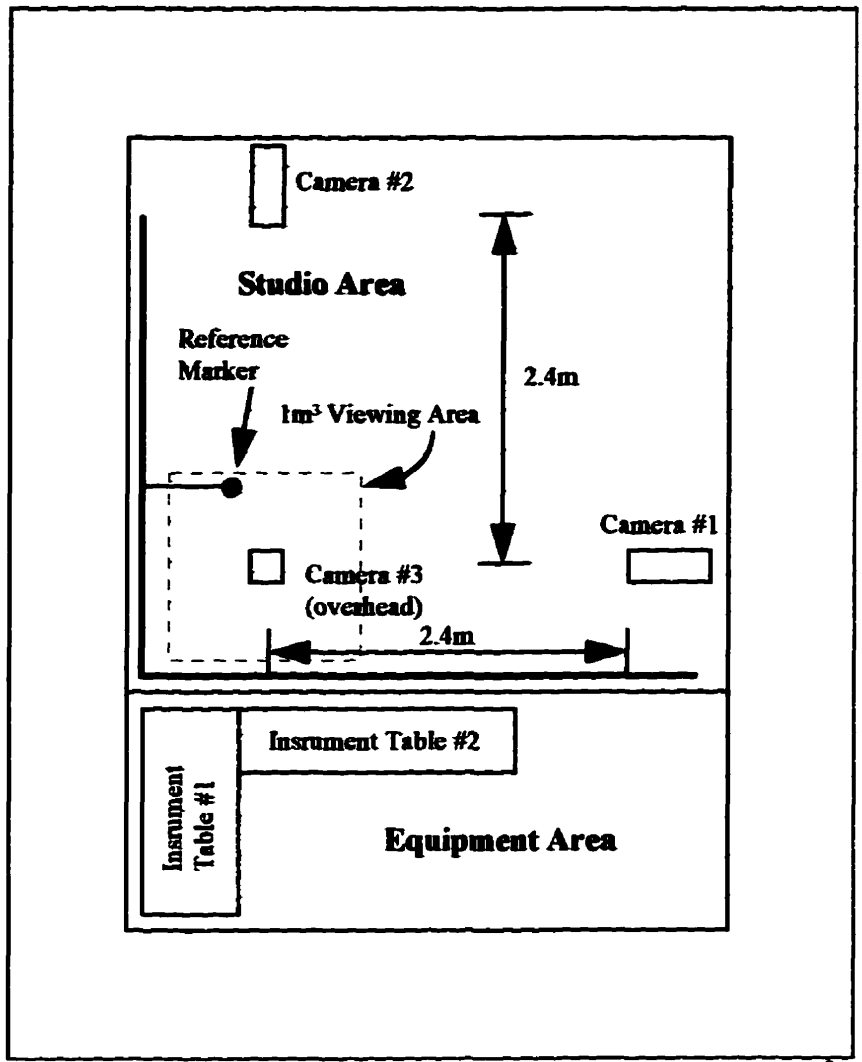


Figure 2.1: An overhead view of the laboratory space in which the UM²AS system operates.

The motion is captured in stereo using at least two cameras and VCRs. As represented in Figure 2.1, the cameras are placed such that the intersection of the optical axes of any two cameras is approximately 2.4 metres from each camera. The intersection point is assumed to be at the centre of the range of motion of the limb being studied.

After the recording has been completed, the video tapes are played back; a sampling rate is selected and each video frame is digitized by the PIPEZ video digitizing board. Once all frames are digitized, the system's software is used to process the

information captured in the digitized frame. The Euler angles are calculated for each joint, at each sampled frame, and are output to a file.

2.2 Image Processing Theory

To better understand the flow of data through the UM^{2}AS system, basic theory about video imaging systems is presented. The first requirement in any video based measurement system is camera calibration. The purpose of camera calibration is to define a method of positional transformation from the image plane of the camera to the actual location of that object in a predetermined global space. Thus, camera calibration is essentially a perspective transformation [Safee-Rad, 1987]. This transformation from the three dimensional global space to a two dimensional image plane is an inherently non-linear process [Schalkoff, 1989]. This non-linear transformation can be converted into a linear transformation with the use of homogeneous coordinates.

Homogenous coordinates are explained in some detail in most image processing text books. Here it is sufficent to say that this non-linear transformation can be converted into a linear transformation by introducing a non-zero arbitrary constant. Doing this, coordinates (u,v) in the image plane become (tu,tv,t), and those in the global space (x,y,z) are transformed into (wx,wy,wz,w). The UM^{2}AS system equates the arbitrary constant w to 1 in the case of the 3-D global space.

The transformation matrix must then be a four by three matrix, if it is to make a transformation in homogeneous coordinates from global space to image space.

$$\begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} L_1 & L_5 & L_9 \\ L_2 & L_6 & L_{10} \\ L_3 & L_7 & L_{11} \\ L_4 & L_8 & L_{12} \end{bmatrix} = \begin{bmatrix} tu \\ tv \\ t \end{bmatrix}$$

2.1

Two equations for each point on the image plane are produced by multiplying out equation 2.1. By simple scaling, it is possible to equate L_{12} to 1. Using this technique, equation 2.2 results for each point on the image plane.

$$\begin{aligned} L_1 x_i + L_2 y_i + L_3 z_i + L_4 - L_9 u_i x_i - L_{10} u_i y_i - L_{11} u_i z_i = u_i \\ L_5 x_i + L_6 y_i + L_7 z_i + L_8 - L_9 v_i x_i - L_{10} v_i y_i - L_{11} v_i z_i = v_i \end{aligned}$$

2.2

Equation 2.2 can then be expanded in matrix form to produce equation 2.3.

$$\begin{bmatrix} x_1 & y_1 & z_1 & 1 & 0 & 0 & 0 & 0 & -u_1 x_1 & -u_1 y_1 & -u_1 z_1 \\ 0 & 0 & 0 & 0 & x_1 & y_1 & z_1 & 1 & -v_1 x_1 & -v_1 y_1 & -v_1 z_1 \\ x_{21} & y_2 & z_2 & 1 & 0 & 0 & 0 & 0 & -u_2 x_2 & -u_2 y_2 & -u_2 z_2 \\ 0 & 0 & 0 & 0 & x_2 & y_2 & z_2 & 1 & -v_2 x_2 & -v_2 y_2 & -v_2 z_2 \\ & & & & \vdots & & & & \vdots & & \\ & & & & \vdots & & & & \vdots & & \\ x_n & y_n & z_n & 1 & 0 & 0 & 0 & 0 & -u_n x_n & -u_n y_n & -u_n z_n \\ 0 & 0 & 0 & 0 & x_n & y_n & z_n & 1 & -v_n x_n & -v_n y_n & -v_n z_n \end{bmatrix} \begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \\ L_5 \\ L_6 \\ L_7 \\ L_8 \\ L_9 \\ L_{10} \\ L_{11} \end{bmatrix} = \begin{bmatrix} u_1 \\ v_1 \\ u_2 \\ v_2 \\ \vdots \\ \vdots \\ u_n \\ v_n \end{bmatrix}$$

2.3

In matrix notation form, equation 2.3 becomes:

$$[\mathbf{P}]_{2n \times 11} [\mathbf{L}]_{11 \times 1} = [\mathbf{P}]_{2n \times 1}$$

2.4

Since each point in the global space produces two equations, it is necessary to use at least six points to determine all eleven calibration parameters in the [L] matrix. The

UM^{2}AS system uses eight calibration markers, each placed at a known location in the global space. These calibration markers are recorded, and their locations on the image plane can then be located. Since eight markers are used, equation 2.4 is an over-determined system. The UM^{2}AS system uses a minimum-squared-error method to solve for the calibration parameters. Thus the pseudo-inverse of [P] is multiplied by [Q] to solve for [L].

$$[\mathbf{L}] = \left([\mathbf{P}]^T[\mathbf{P}]\right)^{-1}[\mathbf{P}]^T[\mathbf{Q}] \tag{2.5}$$

If [R] is equal to the pseudo-inverse of [P], then equation 2.5 becomes:

$$[\mathbf{L}] = [\mathbf{R}]\ [\mathbf{Q}] \tag{2.6}$$

The matrix [L], which represents the calibration parameters, provides a transformation which indicates how the three dimensional global space is converted to a two-dimensional image plane by the camera. Of course in order for UM^{2}AS to be useful, it must convert image plane data back to global space data. To do this it is essential to have stereo recordings of the data, as it is impossible to extract 3-D information from a single image plane.

When the calibration parameters have been determined, equation 2.2 yields two equations and three unknowns. In order to determine three unknowns, a third equation is required. If two (or more) cameras are used, as in the UM^{2}AS system, each camera will produce two equations and thus the result is four equation and three unknowns. This makes absolute determination of the global space possible.

$$
\begin{aligned}
L_1x + L_2y + L_3z + L_4 - L_9u_1x - L_{10}u_1y - L_{11}u_1z &= u_1 \\
L_5x + L_6y + L_7z + L_8 - L_9v_1x - L_{10}v_1y - L_{11}v_1z &= v_1 \\
L_1x + L_2y + L_3z + L_4 - L_9u_2x - L_{10}u_2y - L_{11}u_2z &= u_2 \\
L_5x + L_6y + L_7z + L_8 - L_9v_2x - L_{10}v_2y - L_{11}v_2z &= v_2
\end{aligned}
$$

2.7

By rearranging and simplifying, it is possible to write equation 2.7 as follows:

$$
\begin{aligned}
A_{11}x + A_{12}y + A_{13}z &= B_1 \\
A_{21}x + A_{22}y + A_{23}z &= B_2 \\
A_{31}x + A_{32}y + A_{33}z &= B_3 \\
A_{41}x + A_{42}y + A_{43}z &= B_4
\end{aligned}
$$

2.8

Writing equation 2.8 in matrix form, results in equation 2.9.

$$
\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \\ A_{41} & A_{42} & A_{43} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix}
$$

2.9

In notation form:

$$
[\mathbf{A}]\,[\mathbf{D}] = [\mathbf{B}]
$$

2.10

Since there are four equations and three unknowns, this is also an over-determined system. Again, using the pseudo-inverse for [A] equation 2.11 results.

$$
[\mathbf{D}] = \left([\mathbf{A}]^T[\mathbf{A}]\right)^{-1}[\mathbf{A}]^T[\mathbf{B}]
$$

2.11

Using two cameras, it is possible to determine the (x,y,z) coordinates of a point in global space from the camera images alone, once each of the cameras has been calibrated.

In the UM^{2}AS system, the upper limb is instrumented by using seven body markers. The (x,y,z) coordinates of these body markers are then determined. The (x, y,z) coordinates determine how each body marker is positioned in space. Of course, in a video motion system it is not only necessary to determine how the body markers are moving in space, but also how those markers represent the body being studied, and more importantly, how each segment of that body is moving in relation to the other segments of the body.

2.3 Euler Angles

In the UM^{2}AS system, the relationships between the segments of the upper limb are defined using Euler Angles. Euler Angles, also called Cardan angles [Small, et. al. 1991], have been commonly used to describe rotations occurring in anatomical joints. When using Euler angles, the order of rotation is important in describing the actual position in space. Although other conventions have been detailed [Small, 1991], the method used in the UM^{2}AS system is the same method used by Chao [Chao et al, 1980] and Kinzel[Kinzel, 1983]. This convention conforms to the yaw-pitch-roll system commonly used in flight mechanics.

The order of rotation is first about the z-axis, then about the x′ axis, and finally about the y″ -axis. These rotations can be expressed with standard matrix equations.

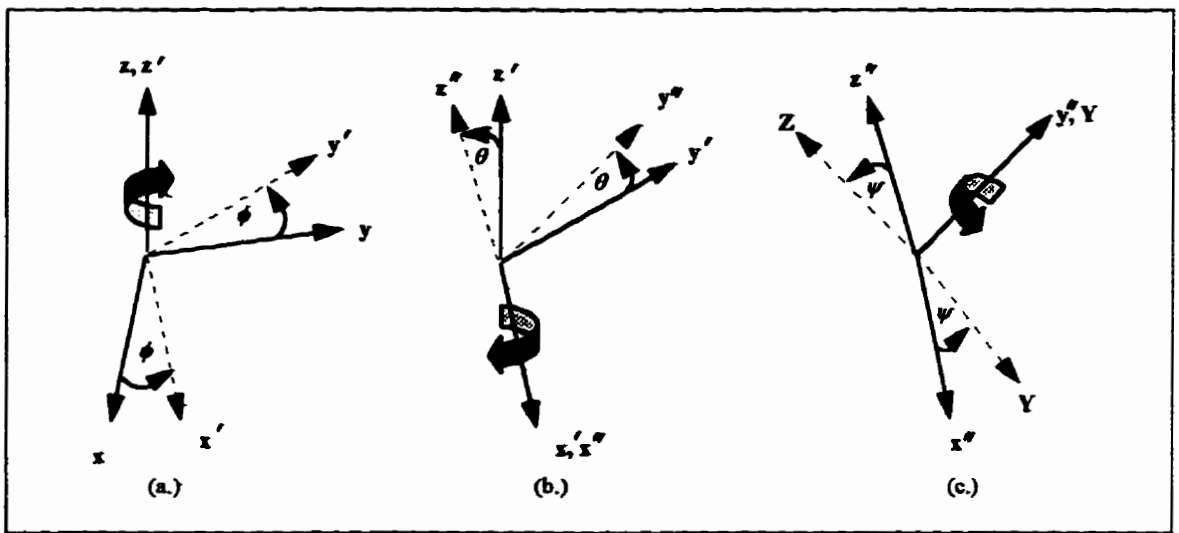


Figure 2.2: The order of rotation used in the UM²AS system

The first rotation, shown in Figure 2.2(a), is a rotation about the z-axis:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \qquad 2.12$$

The second rotation, Figure 2.2(b), is about the x´-axis:

$$\begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \qquad 2.13$$

The final rotation is the rotation about the y˝-axis as indicated in Figure 2.2(c):

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\psi & 0 & -\sin\psi \\ 0 & 1 & 0 \\ \sin\psi & 0 & \cos\psi \end{bmatrix} \begin{bmatrix} x'' \\ y'' \\ z'' \end{bmatrix} \qquad 2.14$$

If the three matrices defined in equations 2.12 through 2.14 are multiplied out, the resultant matrix is the transformation matrix that relates the original (x,y,z) axes to the final (X,Y,Z) axes.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\phi\cos\psi-\sin\phi\sin\theta\sin\psi & \cos\psi\sin\phi+\cos\phi\sin\theta\sin\psi & -\cos\theta\sin\psi \\ -\sin\phi\cos\theta & \cos\phi\cos\theta & \sin\theta \\ \cos\phi\sin\psi+\sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi-\cos\phi\sin\theta\cos\psi & \cos\theta\cos\psi \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.15}$$

Equation 2.15 can be written more compactiy as 2.16 if $[\hat{i},\hat{j},\hat{k}]$ are the unit vectors of the (x,y,z) coordinate system and the $[\hat{I},\hat{J},\hat{K}]$ are the unit vectors of (X,Y,Z).

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \hat{I}\cdot\hat{i} & \hat{I}\cdot\hat{j} & \hat{I}\cdot\hat{k} \\ \hat{J}\cdot\hat{i} & \hat{J}\cdot\hat{j} & \hat{J}\cdot\hat{k} \\ \hat{K}\cdot\hat{i} & \hat{K}\cdot\hat{j} & \hat{K}\cdot\hat{k} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{2.16}$$

Comparing equations 2.15 and 2.16 results in the following relationships between the Euler angles and the unit vectors.

$$\begin{aligned} \theta &= \sin^{-1}(\hat{J}\cdot\hat{k}) \\ \phi &= \cos^{-1}\left(\frac{\hat{J}\cdot\hat{j}}{\cos\theta}\right) \\ \psi &= \cos^{-1}\left(\frac{\hat{K}\cdot\hat{k}}{\cos\theta}\right) \end{aligned} \tag{2.17}$$

If the direction cosines of the unit vectors $\hat{i}$, $\hat{j}$, and $\hat{k}$ in the (X,Y,Z) frame of reference are used in equation 2.17, it is possible to produce equation 2.18 for ϕ, θ, and ψ.

$$\theta = \sin^{-1}[(z_2 - z_1)\cdot\hat{J}]$$

$$\phi = \cos^{-1}[(\frac{y_2 - y_1}{\cos\theta})\cdot\hat{J}]$$

$$\psi = \cos^{-1}[(\frac{z_2 - z_1}{\cos\theta})\cdot\hat{K}]$$

2.18

Equation 2.18 gives a straightforward method to solve for ϕ , θ , and ψ with respect to two defined orthogonal axes. It is now simple to solve for the rotation of each limb segment with respect to the globally defined frame of reference, provided a coordinate axis is defined for each segment.

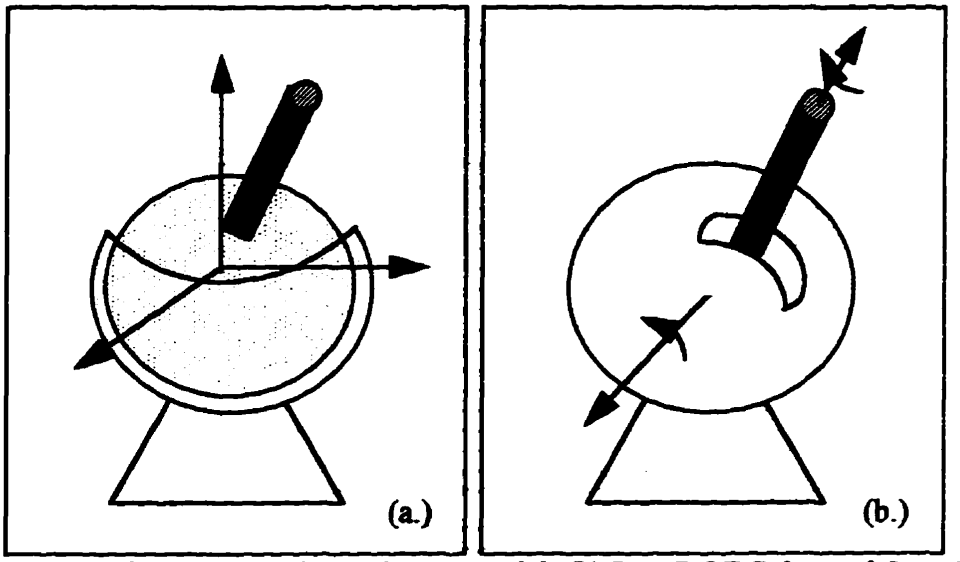


Figure 2.3: (a) Three-DOF Spherical joint model. (b) Two-DOF Spherical Joint Model

In order to apply equation 2.18 to the study of upper limb movement, a kinematic model must be defined for each joint. The wrist and shoulder are modeled as having three degrees of freedom. Since the third rotation in the elbow region is unimportant for the type of studies the UM^{2}AS system was intended for, a two degree of freedom model was chosen for this joint. The models used are the three-DOF spherical joint model, Figure 2.3(a), and the two-DOF spherical joint model, Figure 2.3(b) [Safaee-Rad, 1987].

UM^{2}AS requires that the upper limb be instrumented with a series of seven body markers; these are used to define four sets of orthogonal axes on the upper limb. Figure 2.4(a) illustrates how the markers are placed on the body. The axes are defined such that the adduction-abduction movement at the elbow is omitted [Safaee-Rad, 1987].

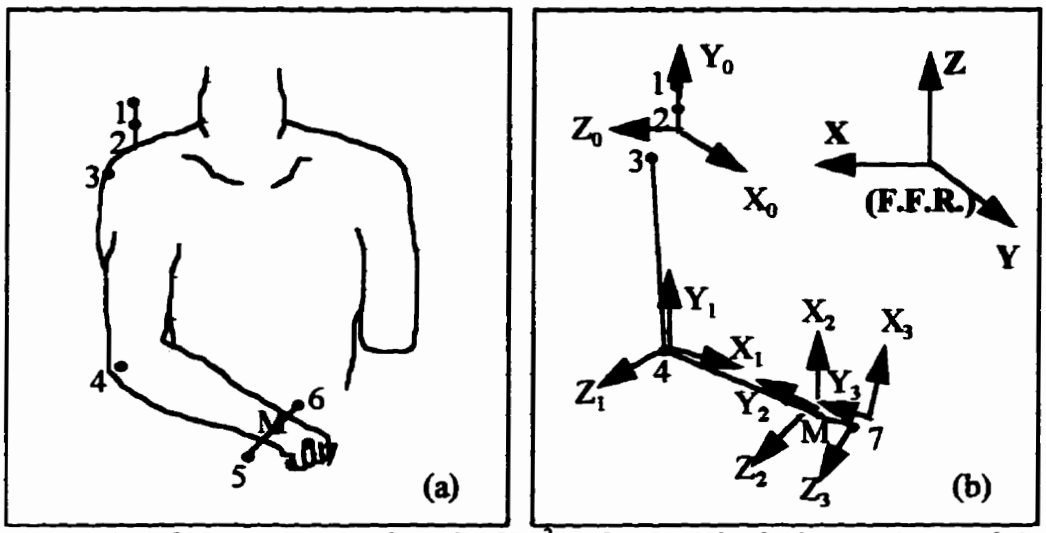


Figure 2.4: (a) Placement of Markers for UM^{2}AS System. (b) Orthogonal Axes defined by Markers.

Using the body markers, the orthogonal axes are defined, as illustrated in Figure 2.4(b). The axes at each joint are defined by equation 2.19 where m_i is the i^{th} marker.

$$\vec{y}_0 = \overrightarrow{m_2m_1}$$
$$\vec{x}_0 = \overrightarrow{m_2m_1} \times \overrightarrow{m_2m_3}$$
$$\vec{z}_0 = \vec{x}_0 \times \vec{y}_0$$

$$\vec{y}_1 = \overrightarrow{m_4m_3}$$
$$\vec{z}_1 = \overrightarrow{m_2m_M} \times \overrightarrow{m_4m_3}$$
$$\vec{x}_1 = \vec{y}_1 \times \vec{z}_1$$

$$\vec{y}_2 = \overrightarrow{m_M m_4}$$
$$\vec{x}_2 = \overrightarrow{m_4 m_6} \times \overrightarrow{m_4 m_5}$$
$$\vec{z}_2 = \vec{x}_2 \times \vec{y}_2$$

$$\vec{y}_3 = \overrightarrow{m_7 m_M}$$
$$\vec{x}_3 = \overrightarrow{m_7 m_5} \times \overrightarrow{m_4 m_5}$$
$$\vec{z}_3 = \vec{x}_3 \times \vec{y}_3$$

2.19

Once the Euler angles are known for each body segment with respect to the global coordinate system, the transformation matrix defined by the equation 2.16 is known. It is then possible to calculate the absolute Euler angles for each body segment.

$$[E_0] = [T_0][E_R]$$
$$[E_1] = [T_1][E_R]$$
$$[E_2] = [T_2][E_R]$$
$$[E_3] = [T_3][E_R]$$

2.20

Finally, the relative rotations between body segments can be found. These relative rotations are assumed to be the rotations that occur in the joints.

$$[E_1] = [T_1][T_0]^{-1}[E_R]$$
$$[E_2] = [T_2][T_1]^{-1}[E_R]$$
$$[E_3] = [T_3][T_2]^{-1}[E_R]$$

2.21

2.4 Summary

Given the basic theory outlined in Section 2.3, the flow of data through the UM^{2}AS system can be described. The first step before every series of recordings, is a camera calibration recording. After this, a series of recordings can then done for a number of subjects in any given test. When the recordings are complete, the data processing begins.

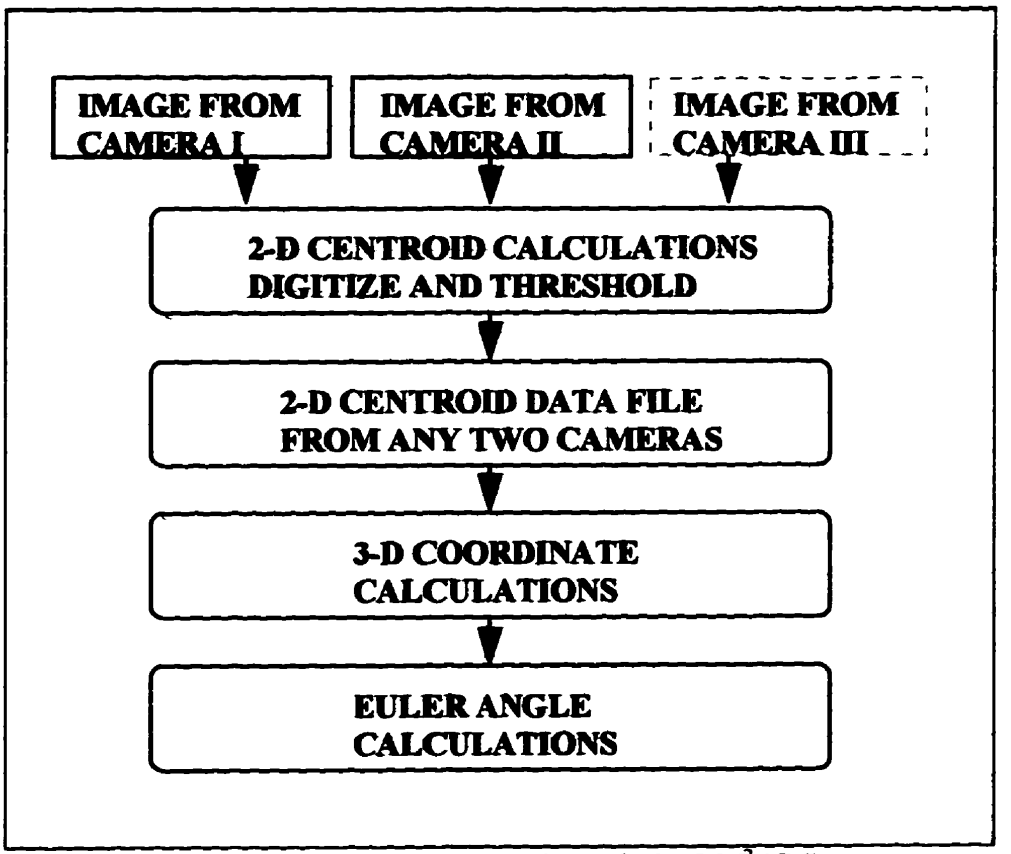


Figure 2.5: Flow Chart of data through the UM^{2}AS System.

Calibration parameters are found by digitizing the images of the calibration markers. The image plane (u,v) coordinates are found for each camera and the calibration parameters are calculated. Once the calibration parameters have been calculated, the rest of the video images are processed and the three dimensional coordinates of each body

marker in the global frame of reference are calculated. These global (x,y,z) coordinates are then subjected to the Euler angle calculation routine which produces the Euler angles at each joint for each set of (x,y,z) coordinates.

The UM^{2}AS system, although effective for its original purpose, it is limited by two factors. First, it assumes that only three joints are being studied. Secondly, it assumes that those particular joints can be effectively modeled by a 3-DOF - 2-DOF - 3-DOF joint model sequence.

Coupled with these assumptions is the necessity that the markers define appropriate orthogonal coordinate systems on the body segments being studied. Thus, the current UM^{2}AS system is, essentially, dedicated to the study of the upper limb, using the joint model implemented. It is therefore very desirable to develop a more generalized system for examining the Euler angles of joint motion.

Chapter Three
Generalized 3-DOF UM²AS System

The limitations of the original UM²AS system was the motivating force for undertaking the present study. If a generalized three degree of freedom model could be developed and implemented, in such a way that it would be possible to select the number of joints studied and the number of DOF examined at each joint, the UM²AS system would be flexible enough to allow many different types of studies.

The current chapter will discuss the development of such a generalized system. Section 3.1 discusses the generalized 3-DOF joint model. Section 3.2 examines the problem of reducing the number of markers required for joint analysis. Section 3.3 presents the rigid body model used in the generalized UM²AS system. The final section, 3.4 discusses the software developed to implement the generalized UM²AS system.

3.1 Generalized 3-DOF Model

The development of a generalized three degree of freedom model has its roots in the standard three degree of freedom joint model used in the original UM²AS system for the upper limb. This model is described by Safaee-Rad and Kinzel and Gutkowski [Safaee-Rad, 1987; Kinzel and Gutkowski, 1983] and can be thought of as a universal joint model.

3.1.1 Generalized Body Motion

The UM²AS system uses Euler angles to quantify the relationship between segments in the upper limb. This technique describes the rotations in the joint and, like all rotational movements, the order of rotations is important. The movement from one position to the next can be described as a rotational transformation.

Equations 2.12-2.14 from the previous chapter show the transformations as the axis of rotation first about the z, the x', and finally about the y"-axis. These rotations are defined by the angles ϕ, θ, and ψ respectively. Therefore the total rotational transformation from the original axis position of (x,y,z) to the final axis position of (X,Y,Z) is shown in Equation 3.1.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \cos\phi\cdot\cos\psi-\sin\phi\cdot\sin\theta\cdot\sin\psi & \sin\phi\cdot\cos\psi+\cos\phi\cdot\sin\theta\cdot\cos\psi & -\cos\theta\cdot\sin\psi \\ -\sin\phi\cdot\cos\theta & \cos\phi\cdot\cos\theta & \sin\theta \\ \cos\phi\cdot\sin\psi+\sin\phi\cdot\sin\theta\cdot\cos\psi & \sin\phi\cdot\sin\psi-\cos\phi\cdot\sin\theta\cdot\cos\psi & \cos\theta\cdot\cos\psi \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{3.1}$$

Assume that the original coordinate system of (X,Y,Z) is fixed and that (x,y,z) is the moving coordinate system (Figure 3.1a), then Equation 3.1 describes the rotational movement between the two coordinate systems. Another way of writing the rotational transformation equations is

$$\begin{aligned} X &= a_{11}x + a_{12}y + a_{13}z \\ Y &= a_{21}x + a_{22}y + a_{23}z \\ Z &= a_{31}x + a_{32}y + a_{33}z \end{aligned} \tag{3.2}$$

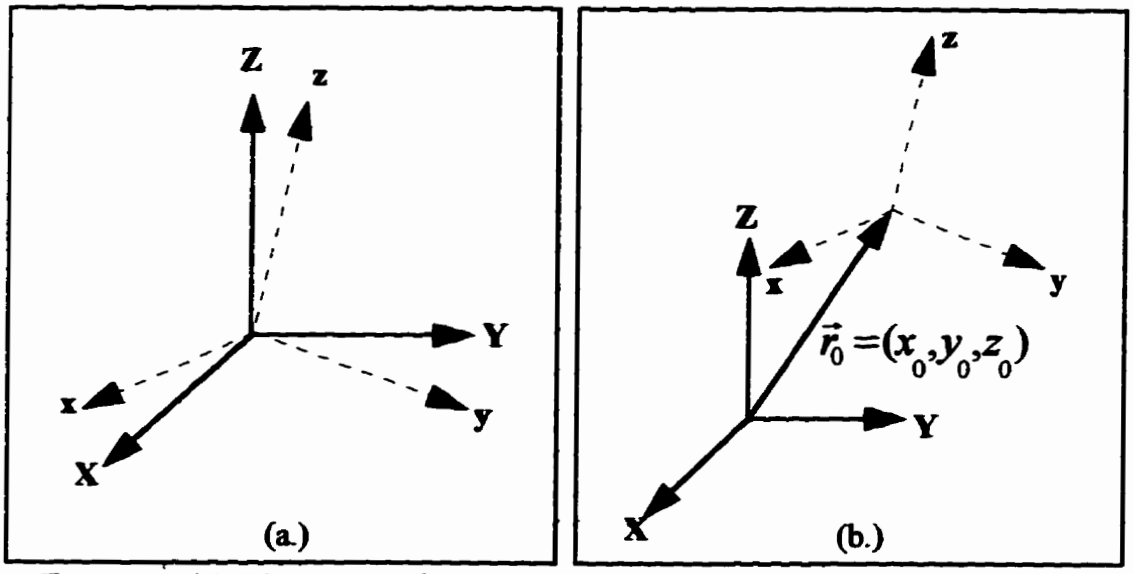


Figure 3.1 (a) Moving coordinate system rotated from the fixed coordinate system. (b) Moving coordinate system displaced and rotated from the fixed coordinate system.

If the origin of the moving coordinate system is moved away from the origin of the fixed coordinates(see Figure 3.1), a displacement vector $\vec{r}_0$ added to the equations. Equation 3.2 becomes 3.3.

$$\begin{aligned} X &= a_{11}x + a_{12}y + a_{13}z + x_0 \\ Y &= a_{21}x + a_{22}y + a_{23}z + y_0 \\ Z &= a_{31}x + a_{32}y + a_{33}z + z_0 \end{aligned} \tag{3.3}$$

If the moving coordinate system is placed on a moving body, such as illustrated in Figure 3.2, the equations are modified slightly. Here the centre of rotation(COR) of the moving body is located at the point COR rather than the origin of the coordinate system. To see what happens to the general equations it is necessary to examine the moving body specifically.

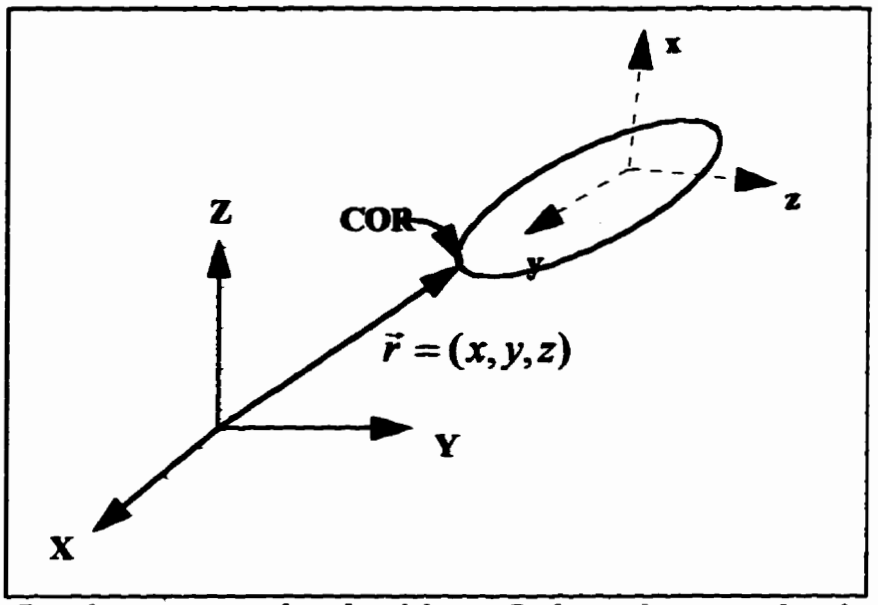


Figure 3.2 Coordinate system placed on Moving Body in a location other than the COR.

Figure 3.3 shows the moving body in position 1 and in position 2. The axis of rotation is the x-axis (the axis coming out of the page). If the distance from the COR to the origin of the coordinate system is R, then the displacement of the origin of the coordinate system due to the rotation at the COR is D. Therefore the displacement can be described by:

$$D = 2R\sin\left(\frac{\theta}{2}\right) \tag{3.4}$$

Figure 3.3 shows that the rotation about the x-axis remains the same at the origin of the coordinate system. Indeed, any point on the moving body would be rotated an equal amount around this axis; the difference in the movement of each point would be a variation in the displacement as a result of the distance of the point from the COR. This is also true if the rotation is about the y or z axes.

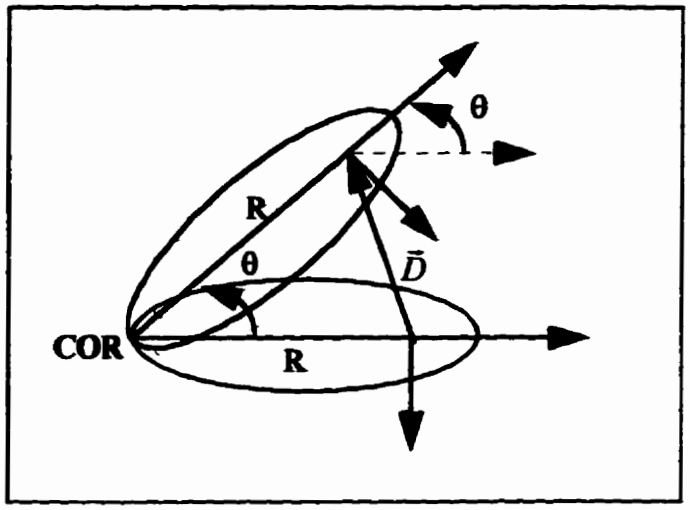


Figure 3.3 Moving Body rotated by an angle θ.

If a displaced COR does not affect the rotation of the moving coordinate system seen by the fixed coordinate system, equation 3.3 remains the same in form. However, the displacement vector $\vec{r}_0$ will have the displacement of the moving coordinate system added to it.

In matrix form Equation 3.3 becomes Equation 3.5.

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & x_0 \\ a_{21} & a_{22} & a_{23} & y_0 \\ a_{31} & a_{32} & a_{33} & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{3.5}$$

Notationally this is:

$$\vec{r}_f = \mathbf{A} \cdot \vec{r}_m \tag{3.6}$$

where **A** is the total transformation matrix. Equation 3.6 can also be expressed notationally as equation 3.7.

$$\vec{r}_f = \mathbf{R} \cdot \vec{r}_m + \vec{r}_0$$

(3.7)

The matrix $\boldsymbol{R}$ in Equation 3.7 is equal to the rotational matrix defined in 3.1. In order to determine the transformation matrix, 3 non-coplanar markers must be used on the moving coordinate system. The marker arrangement defines the axis by placing a marker on each axes. $\mathbf{M}_1$ and $\mathbf{M}_2$ on x, and $\mathbf{M}_3$ on y. If we assume that these markers are located such a manner that each one defines a unit vector, the marker positions with respect to the moving coordinate system is:

$$\vec{r}_m^1 = (1,0,0)^T$$
$$\vec{r}_m^2 = (-1,0,0)^T$$
$$\vec{r}_m^3 = (0,1,0)^T$$

(3.8)

Figure 3.4 shows the marker arrangement on the moving coordinate system. From this figure we can see that the displacement vector $\vec{r}_0$ can be defined as:

$$\vec{r}_0 = \frac{1}{2}\left(\vec{r}_f^1 + \vec{r}_f^2\right)$$

(3.9)

The vector defining the z-axis is cross product of the vectors defining the x-axis and y-axis.

$$\vec{r}_m^4 = \vec{r}_m^1 \times \vec{r}_m^3$$

(3.10)

Therefore Equation 3.7 can be rewritten as:

$$\left(\vec{r}_f^1, \vec{r}_f^3, \vec{r}_f^4\right)^T = \mathbf{R}\left(\vec{r}_m^1, \vec{r}_m^3, \vec{r}_m^4\right)^T + \vec{r}_0$$

(3.11)

Subtracting $\vec{r}_0$ from both sides of the equation results in Equation 3.12.

$$\left(\vec{r}_f^{\,1}-\vec{r}_0,\vec{r}_f^{\,3}-\vec{r}_0,\vec{r}_f^{\,4}-\vec{r}_0\right)^T=\mathbf{R}\left(\vec{r}_m^{\,1},\vec{r}_m^{\,3},\vec{r}_m^{\,4}\right)^T \tag{3.12}$$

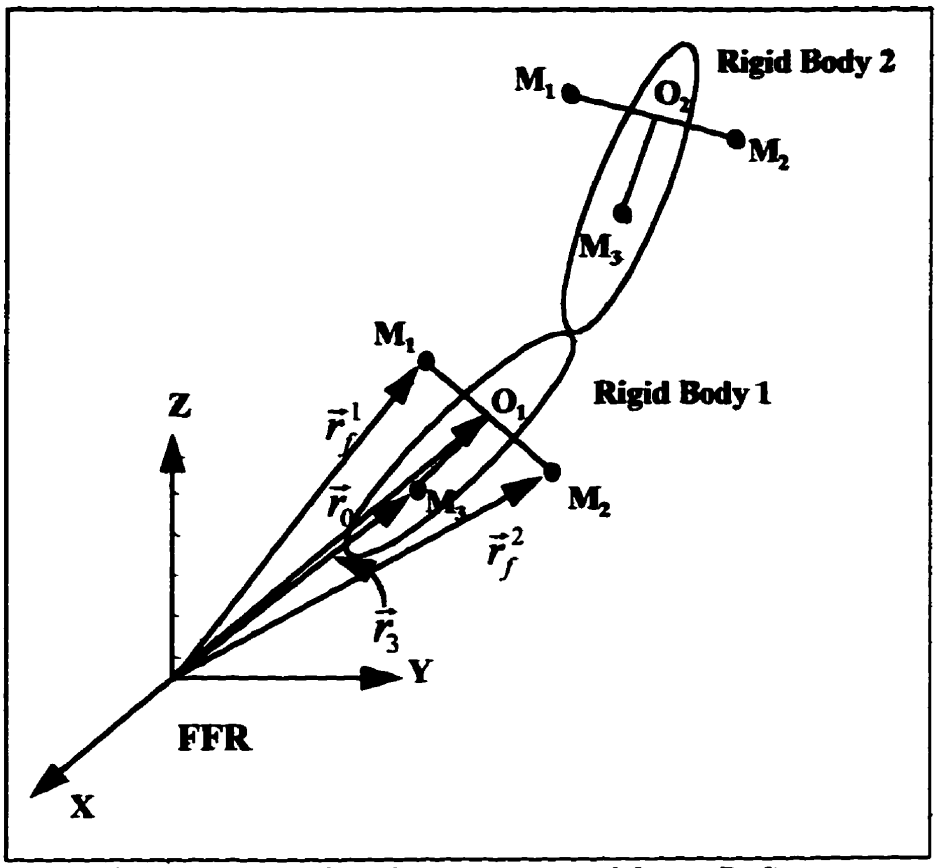


Figure 3.4 Marker placement on two Moving Bodies.

Since the vectors defining the points $\mathbf{M_1}$, $\mathbf{M_2}$, and $\mathbf{M_3}$ are unit vectors, the right-hand side of equation 3.13 can be written as:

$$\mathbf{R}(\vec{r}_m^{\,1},\vec{r}_m^{\,3},\vec{r}_m^{\,4})^T=\mathbf{R}\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}=\mathbf{RI} \tag{3.13}$$

Therefore the rotation matrix, **R**, is simply

$$\mathbf{R} = \left(\vec{r}_f^{\,1} - \vec{r}_0, \vec{r}_f^{\,3} - \vec{r}_0, \vec{r}_f^{\,4} - \vec{r}_0\right)^T = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \tag{3.14}$$

Using equation 3.1 and 3.14, it is possible to solve for unique values of ϕ, θ, and ψ.

$$\theta = \sin^{-1}(a_{23}) \tag{3.15}$$

$$\psi = \sin^{-1}\left(\frac{a_{13}}{-\cos\theta}\right)$$

$$\psi = \cos^{-1}\left(\frac{a_{33}}{-\cos\theta}\right) \tag{3.16}$$

$$\phi = \sin^{-1}\left(\frac{a_{21}}{-\cos\theta}\right)$$

$$\phi = \cos^{-1}\left(\frac{a_{22}}{-\cos\theta}\right) \tag{3.17}$$

Equations 3.15, 3.16, and 3.17 allow the calculation of the rotations between the moving and the fixed coordinate system. Since the displacement vector can be calculated using equation 3.11 and the rotations, the total transformation matrix, A, is defined. This matrix enables the calculation of the Euler angles.

3.1.2 Calculating the Euler Angles

The purpose of calculating the Euler angles is to describe the motion between two moving bodies. In the case of joint motion, the moving coordinate system is attached to

the body segment on one side of the joint and the fixed coordinate is assumed to be attached to the body segment on the other side of the joint. The rotations at the joint can now be calculated using equation 3.1 and 3.5. If a series of body segments are to be studied, the purpose is to describe the motion of each body segment relative to the segment it is attached to.

Euler angles are calculated in much the same way as described in Chapter Two for the original UM^{2}AS system. If three body segments are being studied, as in the upper limb, a coordinate system would be placed on each segment. The transformation, **A**, would then be calculated for each body segment with respect to the fixed coordinate system.

$$\begin{aligned} \vec{r}_f &= \mathbf{A}_1 \vec{r}_m^2 \\ \vec{r}_f &= \mathbf{A}_2 \vec{r}_m^2 \\ \vec{r}_f &= \mathbf{A}_3 \vec{r}_m^3 \end{aligned} \tag{3.18}$$

In order to find the relative motion between each body segment, it is necessary to find a transformation matrix from one moving coordinate system to the next. If this transformation is defined as $\mathbf{B}_{ij}$, where *i* represents the proximal segment and *j* the distal, segment, Equation 3.19 results for three body segments.

$$\begin{aligned} \vec{r}_{m2} &= \mathbf{B}_{23} \vec{r}_{m3} \\ \vec{r}_{m1} &= \mathbf{B}_{23} \vec{r}_{m2} \\ \vec{r}_f &= \mathbf{B}_{01} \vec{r}_{m1} \end{aligned} \tag{3.19}$$

By substituting the equations in 3.18 into 3.19, $\mathbf{B}_{ij}$ is defined as in equation 3.20.

$$\mathbf{B}_{23} = \mathbf{A}_2^{-1}\mathbf{A}_3$$
$$\mathbf{B}_{12} = \mathbf{A}_1^{-1}\mathbf{A}_2$$
$$\mathbf{B}_{01} = \mathbf{A}_1 \tag{3.20}$$

In general $\mathbf{B}_{ij}$ is defined by:

$$\mathbf{B}_{ij} = \mathbf{A}_i^{-1}\mathbf{A}_j \tag{3.21}$$

Once the transformation matrices, from moving coordinates to fixed coordinates are defined, it is possible to calculate the Euler angles in each joint being examined. The sequence of rotations for calculating Euler angles can be defined in various ways. However, in order to have the rotations about the axes of the moving coordinate system represent anatomical angles, the moving coordinate systems must be placed on the body so that the axes of the coordinate system are valid axes of rotation in the joint being examined.

3.2 Reduction of markers

Although the preceding methodology is a completely generalized method of describing the motion between two rigid bodies, in practical terms it is not necessarily viable. In theory three non-collinear markers are required on each rigid body segment to uniquely define the three dimensional motion between two rigid bodies [Youm and Yoon, 1979]. This includes a definition of translation as well as a definition of rotation. However, it is impossible to get an accurate definition of translation without knowing the centre of rotation of the moving body.

Calculating the centre of rotation is the subject of a great deal of discussion in the literature [Blacharski, et al, 1975; Soudan, et al, 1979; Spiegelman and Woo, 1987; Holzreiter, 1991; Crisco, et al, 1994; Panjabi, et al, 1982]. However there does not seem to be a good method developed for calculating the centre of rotation in joint models other than the planar model. Thus, most discussion is limited to the 2-dimensional case with no obvious way to handle a 3-dimensional environment. Indeed, Soudan, et al, state that the application of the Reuleaux method in three dimensions, as done by Blacharski, et al, is not even possible and leads to inaccurate results. Fortunately, anatomical translations are typically very small and can be safely ignored in the types of studies the generalized UM^{2}AS system is intended for.

Of course, if translational motions are ignored, it becomes possible to define the rotations between two body segments with less than 3 markers. This is a very desirable endeavor, as the most time consuming aspect of the UM^{2}AS system is the marker digitization process. Although this process is automated to a certain extent, it still requires the intervention of an operator when marker positions are close together.

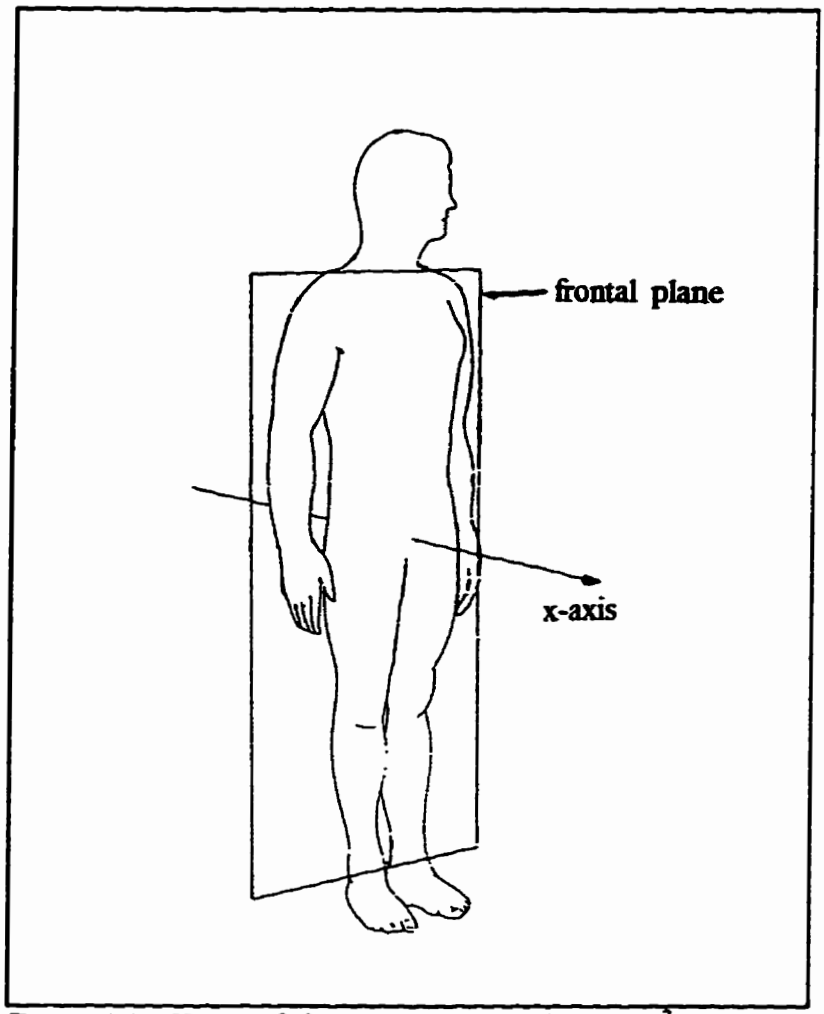


Figure 3.5: X-Axis definition for generalized UM²AS system.

The proposed reduced marker system would place two markers on each rigid body segment(with some exceptions). It is required that the markers define a single axis of the coordinate system. The axis chosen is the x-axis. This axis is defined to run parallel to the floor and be oriented perpendicular to the frontal plane of the body, if the subject is standing with hands at sides as depicted in Figure 3.5. This orientation coincides with Safaee-Rad's [Safaee-Rad, 1987; Safaee-Rad, 1990] coordinate orientation choice for the

shoulder, elbow, and wrist. The markers are placed at the distal end of each rigid body segment as shown in Figure 3.6, the axes of each coordinate system are then defined as:

$$\begin{aligned} \vec{x}_i &= \vec{m}_i - \vec{M}_i \\ \vec{y}_i &= \overrightarrow{M_i M_{i-1}} \\ \vec{z}_i &= \vec{x}_i \cdot \vec{y}_i \end{aligned} \tag{3.22}$$

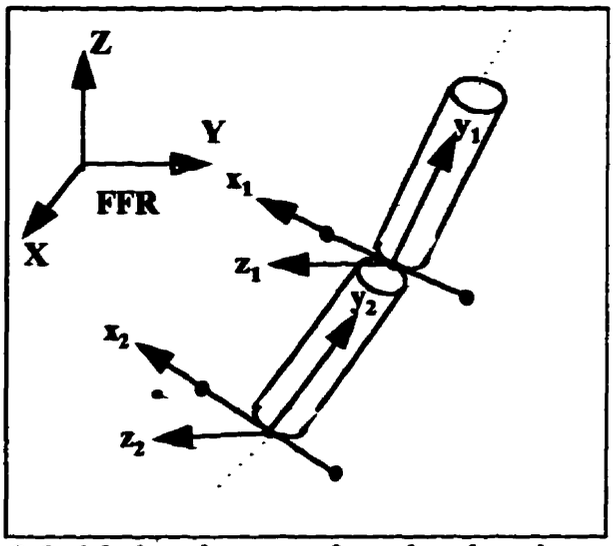


Figure 3.6: Marker placement for reduced marker system.

The marker placement demonstrated in Figure 3.6 is most representative of the lower and upper limb. When marker placements on the torso and head are considered, certain adjustments must be made to the marker placements.

3.3 Rigid Body Model used in the generalized UM²AS system

In order to consider the correct marker placements, the model used for generalizing the UM²AS system must be considered. The human body is modeled as a system of 15 rigid bodies attached by a system of 14 3-DOF spherical joints. However,

since only one side of the body can be examined at a time, in reality the model is a series of 8 joints (C7-T1, Lumbar-Sacral, Hip, Knee, Ankle, Shoulder, Elbow, and Wrist) connecting 9 rigid body links (Head/Neck, Thorax, Pelvis, Thigh, Calf, Foot, Upper Arm, Forearm, and Hand).

When considering the model used for the human body, it becomes clear that the marker system depicted in Figure 3.6 cannot be used for every body segment in the model. Therefore the proposed marker system is a two marker unit placed on the foot, calf, thigh, hand and forearm. Three marker units are used for the head, thorax, pelvis and upper arm. Proposed marker placements are illustrated in Figure 3.7.

It is not necessary for the marker units to be of the same length, however it is important that the x-axis markers be centered on the body segment for those marker systems whose mid-point is used to define the y-axis of other body segments. This proposed reduced marker system is the marker system assumed by the generalized UM^{2}AS system discussed in the following section.

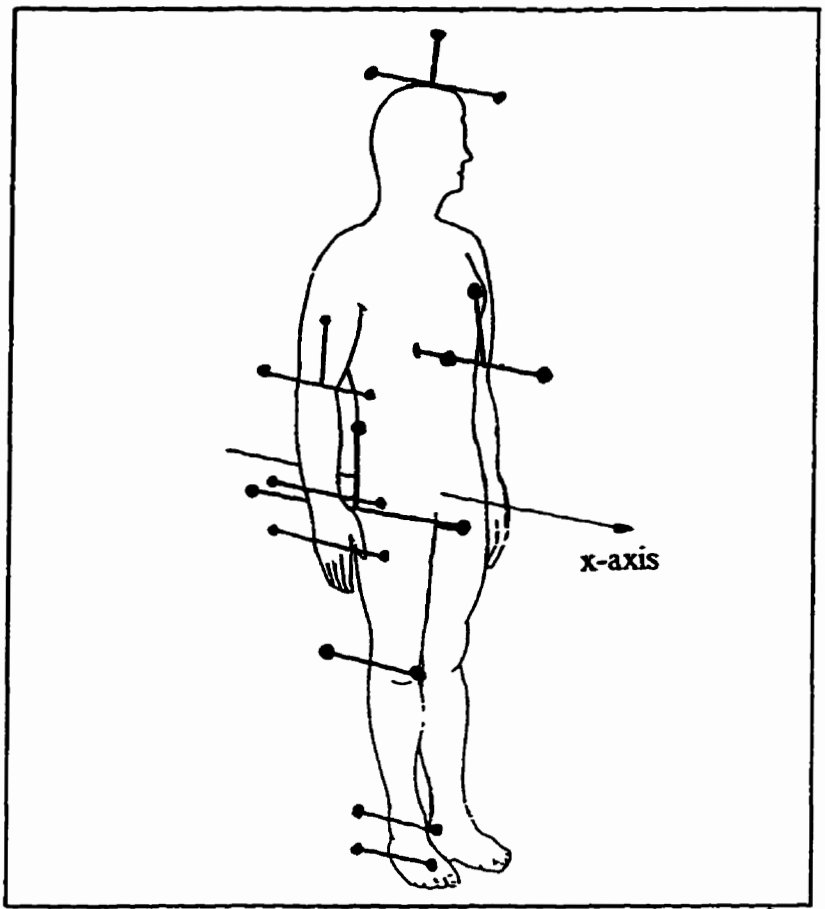


Figure 3.7: Proposed marker placements for the generalized UM²AS system.

3.4 Generalized UM²AS Software

The software developed for the generalized UM²AS system was written in Borland C++ for Windows. This software is contained in three main files, UM2ASG, USERINPT, and CALCANGL. The three underlying assumptions of the generalized UM²AS are: (1) The marker system used is the system proposed in the previous section. (2) The markers for each coordinate system are digitized in a prescribed order. The marker placed on the x-axis on the "positive"(the front marker) side first, the "negative" x-marker second, and lastly the y-marker (if it is needed for the coordinate system). (3) The coordinate systems

for each frame are digitized in order of Head, Thorax, Pelvis, Thigh, Calf, Foot, Arm, Forearm, and lastly Hand. If not all joints are to be studied, the appropriate coordinate systems may not be used, but those required must be digitized in the specified order.

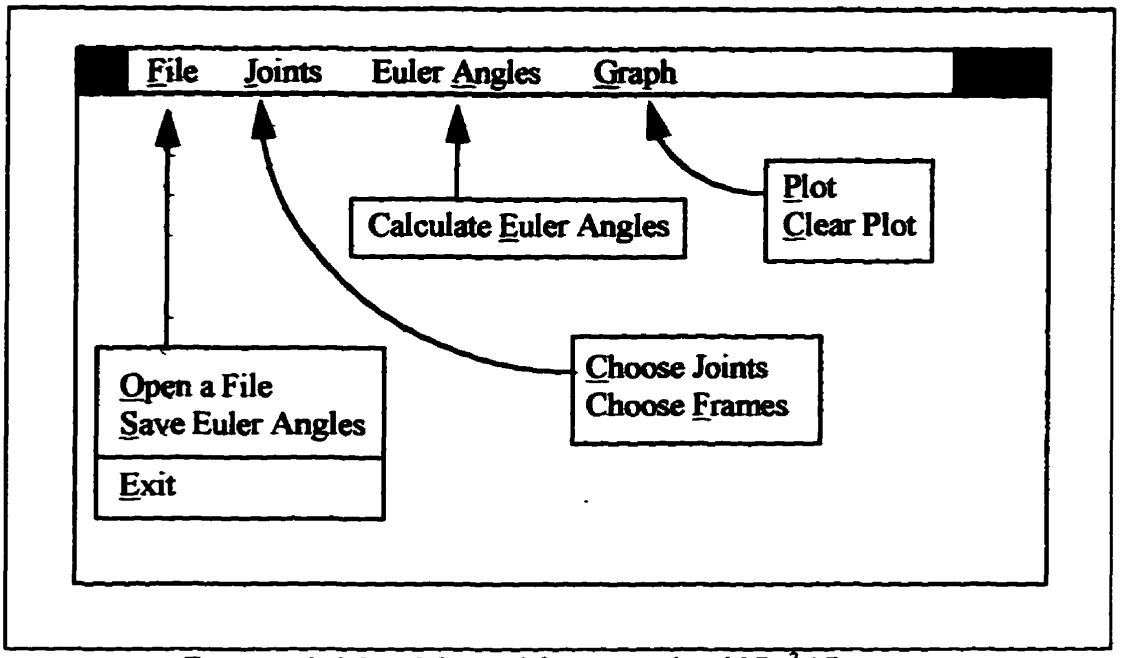


Figure 3.8 Main Menu of the generalized UM²AS system.

The file UM2ASG contains the software that manages the presentation and control of the system. This section of the program is used to present the main menu in a Windows environment and allow the user to select the tasks to be performed. The UM2ASG main menu is presented in Figure 3.8. The FILE option allows the user to select the three dimensional coordinate data files for analysis. Included in this pop-up menu item is a file save option to allow the user to save the calculated angles in a filename of choice and the EXIT option used for quitting the system. Angle data are saved in an ASCII format in columns representing the joint and the ϕ, θ, and ψ angles. A small sample of a saved file is shown in Figure 3.9. The data files can easily be imported into EXCEL or a statistical package for ease of further analysis.

C7 Angles			LumSac Angles			Hip
Phi	Theta	Psi	Phi	Theta	Psi	Phi
-4.999	-7.000	2.995	2.997	1.000	-5.002	1.011
-7.999	-12.000	7.999	7.998	4.000	-7.998	4.003
-11.002	-17.000	13.001	12.999	7.000	-11.000	7.000
-14.000	-22.000	18.001	18.000	10.000	-14.001	10.000
-17.001	-27.000	22.999	23.000	13.000	-17.001	12.998
-20.001	-32.000	28.000	28.000	16.000	-20.000	15.999
-22.999	-37.000	33.000	33.000	19.000	-23.001	19.001

Figure 3.9: Example of calculated Euler Angle file.

The "Joint" option in the main menu, has two choices in its pop-up menu, "Choose Joints" and "Choose Frames". If "Choose Frames" is selected, a Dialog Box appears that requests the input of the number of frames to be analyzed. When "Choose Joints" is selected, a Dialog Box appears, as shown in Figure 3.10. This Dialog Box lists the possible joint choices, each with an associated Check Box. A joint can be selected by selecting the associated Check Box. Once the user chooses "OK", a Dialog Box for each selected joint appears in succession. Each joint's Dialog Box allows the user to select the rotations to be analyzed for the current joint. If less than three rotations are selected for each joint, the method of calculating the joint rotations is modified as explained later.

The third menu option, "Calculate Angles", has only one item in its pop-up menu. This option "Calculate Euler Angles" calls the Euler angle calculation routines. It requires the name of the TDC file to be analyzed, the name of the file chosen to store calculated angle data to, the number of frames to be analyzed, and the joints selected to be input by the user prior to selecting "CalculateEulerAngles". (These functions are contained in the

file CALCANGL.) If any of the four pieces of information required by the data processing functions have not yet been selected, a message appears in the Status Bar at the bottom of the screen. This message informs the user what information is missing. At this point the user may select the missing information and then choose "Calculate Euler Angles" again.

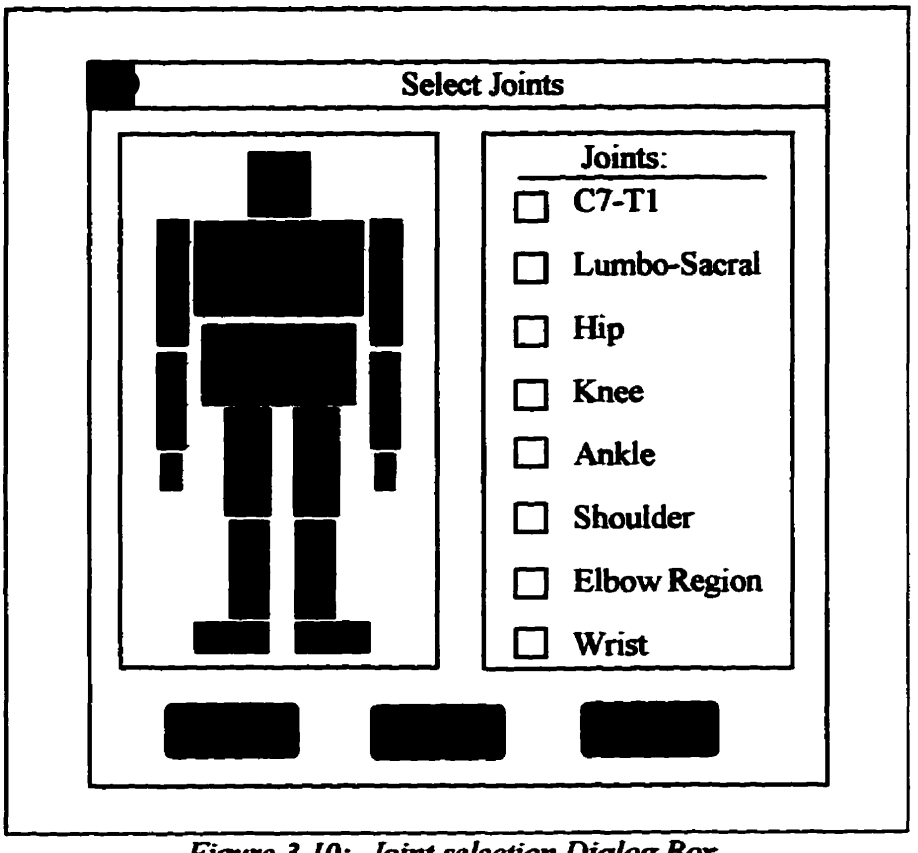


Figure 3.10: Joint selection Dialog Box.

The final menu item in the main menu is the selection "Plot". The pop-up menu for this option contains two items, "Plot" and "Clear Plot". If the option "Plot" is selected, a Dialog Box similar to the one used for selecting the joints for analyses appears. This Dialog Box allows the user to select the joint angles that are to be to plotted on the

screen. After the joint is selected, the ϕ, θ, and ψ angles for that joint are plotted to the screen. The "Clear Plot" selection clears the current plot from the screen.

The second software file, USERINPT, contains the Dialog Box controls for the Dialog Boxes that are used by the main menu options. This file contains the functions that process the user's selections from the Dialog Boxes and returns the necessary information back to the main program control, contained in the UM2ASG file.

The final software file, CALCANGL, encompasses all of the functions used for the calculation of the Euler Angles. The program control throughout the Euler angle calculations is much more complex than in the previously described files. The flowchart, illustrated in Figure 3.11, gives a general overview of the method used for calculating the Euler angles in the generalized UM^{2}AS system. The early parts of the flowchart are quite straight forward, the section following the decision box "All rotations selected in each joint?" requires more explanation. If all rotations are not selected in a joint, a reduced version of Equation 3.1 is used for finding the rotations. There are 6 options for this, x-rotation only, y-rotation only, z-rotation only, x and y rotations, x and z rotations, and y and z rotations. The modified matrices are listed in Appendix A.

If less than three rotations are selected, the method of solving for the rotations is more straightforward. In each of the options, the rotation matrix has a cosine and a sine of each angle to be solved for, in isolation. This allows for an easy method of deciding the appropriate quadrant for the calculated angle. This decision is important as computers solve the inverse of sine between $\pi/2$ and $-\pi/2$, and inverse cosine between 0 and π.

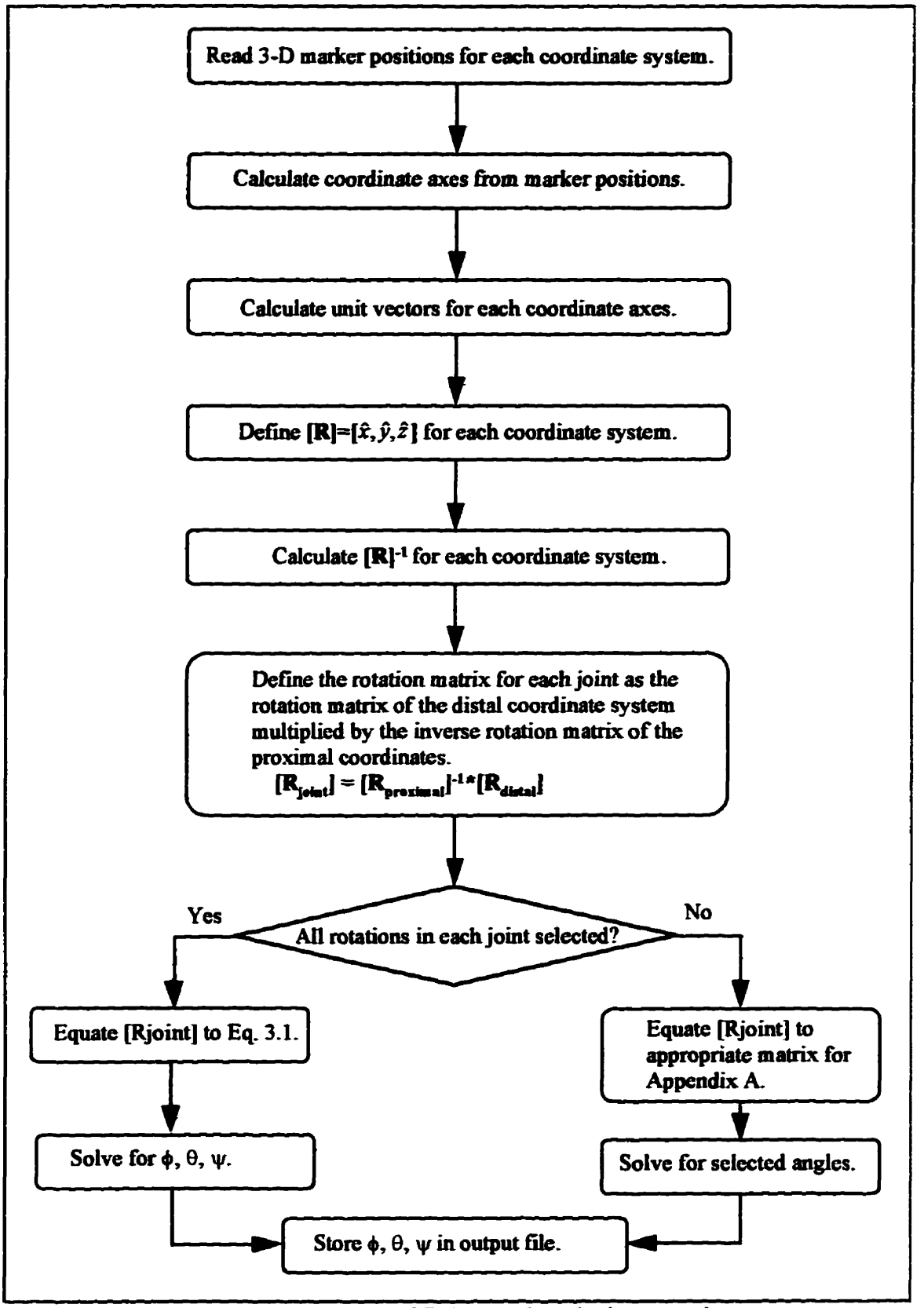


Figure 3.11: Flowchart of Euler Angle calculation technique.

When all three angles for a joint are selected the method of solving for the angles is not as straight forward. In this case, the complete rotation matrix of Equation 3.1 must be used. Since only $sin(\theta)$ is isolated in this matrix, the method of solving for ϕ, θ, ψ is much more complex. In general, θ is solved using Equation 3.15. A second value of θ is found as $(\pi-\theta)$ if θ is positive and $(-\pi-\theta)$ if θ is negative. Using these two values of θ, the corresponding values of ϕ and ψ are calculated using Equations 3.16 and 3.17. The values of ϕ, θ, ψ are compared to the previous values, and the set of angles with the smallest difference from the previous set of angles is chosen as the correct set of angles.

In the case of $\theta = 90$, there is no solution to Equation 3.1. When this occurs, the values of ϕ and ψ are solved using a two point linear extrapolation.

$$\phi_i = \phi_{i-1} + (\phi_{i-1} - \phi_{i-2}) \tag{3.23}$$

Appendix B contains flowcharts for each function used for calculating Euler angles in the generalized UM^{2}AS system. Also included are flowcharts of functions included in the UM2ASG file and the USERINPT file.

Calculating the Euler angles of each joint is the ultimate function of the generalized UM2AS system. Testing of this system was performed using a computer program to generate marker locations. The testing procedure is discussed in the following chapter.

Chapter Four
Testing and Validation

In order to assess the effectiveness of the generalized UM^{2}AS system, it was necessary to develop a testing procedure. It is relatively difficult to compare the results from various joint analyses systems [Small et al, 1992; Kinzel and Gutkowski, 1982] due to the variety of joint models and measurement systems available. Therefore, human testing was not considered very desirable and a computer simulation testing method was decided upon. The advantage of this method is the ability to know exactly what results should be expected. It is therefore easy to assess the size of the error made by the system.

A program was written, in Borland C++ for Windows, to generate potential marker positions. This program, Gen_Mark, allows the user to select the joints to simulate and store the generated positions to a TDC file, as required by the generalized UM^{2}AS software. The relative error between the calculated Euler angles and the angles used to produce the marker positions was computed.

4.1 Development of the Testing Method

The input for testing the generalized UM^{2}AS software is the 3-D marker positions. Ideally these marker coordinates closely match potential coordinates produced by human movement.

Since the UM^{2}AS system has been used primarily to examine upper limb motion in human subjects, the development of the marker generation theory will be explained using a

model that approximates the upper limb. A three segment rigid body model, such as the model shown in Figure 4.1, is used.

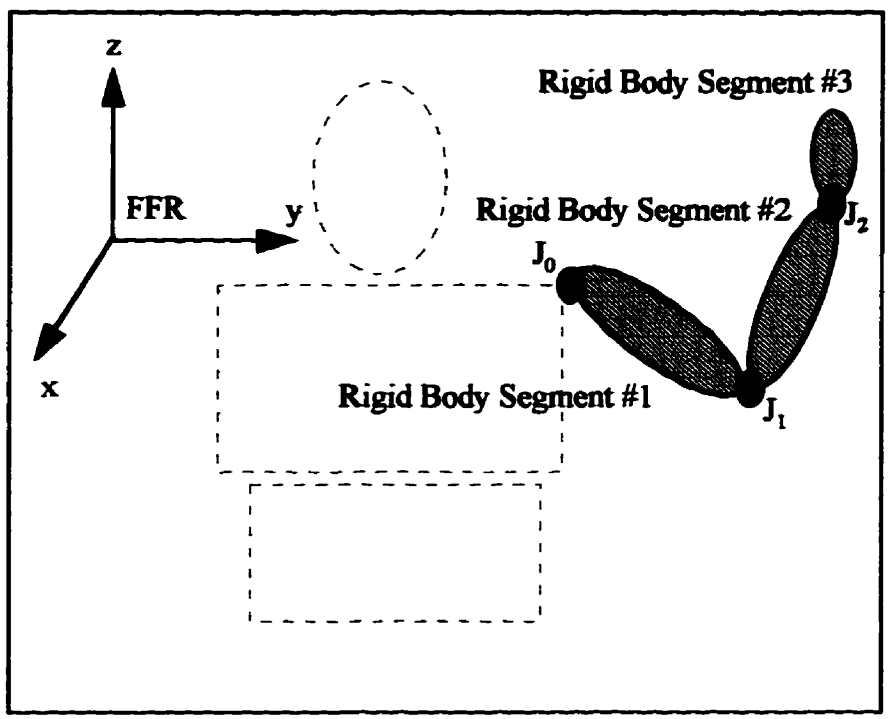


Figure 4.1: Rigid Body model used for developing the testing model.

The Gen_Mark program is required to generate the marker positions of the coordinate systems placed on each rigid body segment. It has been shown (in Figure 3.4) that the location of the coordinate systems do not have to be located at the centre of rotation of the moving segment in order to determine valid rotations for the moving segment. However, the generalized UM^{2}AS system does assume marker placement as described in Section 3.4 of the previous chapter and this is the marker placement used by the Gen_Mark program.

For any set of rigid bodies attached together by joints, the rotations of the more proximal joints such as J_0 and J_1, affect the marker positions of the coordinate system placed on a more distal body segment. Therefore the markers on MB_2 and MB_3 will move

if a rotation occurs in J_0. On the other hand, a rotation in J_2 will not affect the location of markers placed on MB_2 and MB_1. It is more efficient, from a programming point of view, to first rotate markers on the most distal rigid body segment and then rotate successive proximal segments.

Table 4.1: Nomenclature used for discussing the development of the testing program.

SYMBOL	DESCRIPTION
J_i	:Joint number *i*
$\left[C_i\right]_j$	:3 by 3 matrix of the markers of the coordinate system on MB_i w.r.t. the coordinate system on MB_j ($[x^+_i, x^-_i, y_i]_j$)
$\vec{D}_i$	:displacement vector from the origin of the centre of rotation on MB_i to the coordinate system placed on it
$\left[R_n\right]$	:rotation matrix representing the rotations that occur in joint number *n*
$\vec{r}_i$	:vector from the origin of the FFR to the origin of the coordinate system on MB_i

If MB_3 is rotated at J_2 by rotation matrix $[R_2]$ the resulting marker positions for coordinate system three are represented as $[C_3']_3$. Since the orientation of $\vec{D}_3$ is affected by the rotation in J_2, $\vec{D}_3$ must also be multiplied by $[R_2]$.

$$\left[\mathbf{C_3'}\right]_3 = \left[\mathbf{R_2}\right] \cdot \left[\mathbf{C_3}\right]_3 \qquad 4.1$$

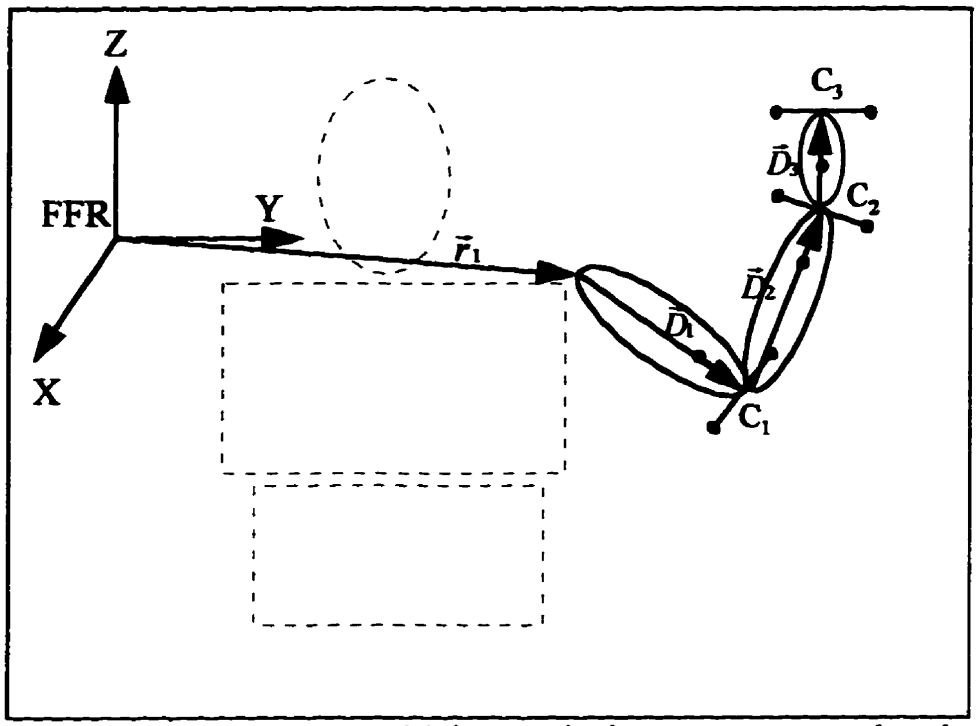


Figure 4.2: A three-segment model showing displacement vectors and markers.

To represent the coordinates of MB_3 with respect to the coordinate system on MB_2, the displacement vector $\vec{D}_3$ must be added to the rotated coordinates $[C'_3]_3$.

$$[C_3]_2 = [C_3']_3 + [R_2] \cdot \vec{D}_3 \qquad 4.2$$

The rotation in J_1 acts on both the markers places on MB_2 and MB_3. The resulting positions of the coordinates on MB_2 are:

$$[C_2']_2 = [R_1][C_2]_2 \qquad 4.3$$

The positions of MB_3 markers with respect to the coordinate system on MB_2 after rotation at J_1 is the marker positions multiplied by the rotation matrix representing the rotation in J_1.

$$[C_3']_2 = [R_1][C_3]_2 \qquad 4.4$$

The rotations occurring in J_0 affect the coordinate systems on all three body segments. Markers placed on MB_3 are known (Eq. 4.4) with respect to MB_2. Therefore they are known with respect to MB_1 by adding the displacement vector $\vec{D}_2$. Here the displacement vector must be rotated by the rotation in J_0.

$$[C_3]_1 = [C_3']_2 + [R_1] \cdot \vec{D}_2 \tag{4.5}$$

Similarly, markers on MB_2 are known with respect to the coordinate system on MB_1 by adding the rotated displacement vector $\vec{D}_2$.

$$[C_2]_1 = [C_2']_2 + [R_1] \cdot \vec{D}_2 \tag{4.6}$$

Finally, after the rotation of J_0 occurs, the markers for all three moving bodies can be expressed, with respect to a fixed frame of reference placed on the trunk, as follows:

For MB_3:

$$[C_3']_{FFR} = [R_0]\,[C_3]_1 + [R_0] \cdot \vec{D}_1 \tag{4.7}$$

For MB_2:

$$[C_2']_{FFR} = [R_0]\,[C_2]_1 + [R_0] \cdot \vec{D}_1 \tag{4.8}$$

For MB_1:

$$[C_1']_{FFR} = [R_0]\,[C_1]_1 + [R_0] \cdot \vec{D}_1 \tag{4.9}$$

It would be possible to express the marker coordinates in terms of another, non-body based, fixed frame of reference. This task would be accomplished by adding the

vector $\vec{r}_1$ to all the marker positions. However, since all the marker positions would be modified by the same vector, this final step was deemed unnecessary for the testing procedure. Therefore, Equations 4.7 - 4.9 represent the marker positions of each moving body, with respect to the fixed frame of reference, after the rotations in each joint have occurred. These are the marker positions required by the generalized UM^{2}AS system.

4.2 Expanding a 3-Segment System to the Whole Body

In order to adequately test the generalized UM^{2}AS system, it is necessary to model the entire body. Since the generalized UM^{2}AS assumes that the body is modeled as a system of eight joints: C7-T1, Lumbar-Sacral, Hip, Knee, Ankle, Shoulder, Elbow, and Wrist, it is necessary to expand the marker generation program to produce markers representing nine rigid body segments. This expansion is not a simple matter of increasing the number of rigid body segments in the chain. The model used must represent the possible connection in the human body.

From Figure 4.3 it is possible to see that the movement of the head and neck due to a rotation occurring in C7-T1, does not affect the position of any other rigid body segment. However, a rotation occurring in the Lumbar-Sacral joint will move the body segments of Pelvis, Thigh, Calf, and Foot, provided no other rotations occur in the Hip, Knee or Ankle joints. Thus it is necessary to treat the upper limb as a 3-segment unit, the head/neck as a single segment unit, and the lower limb, including the pelvis, as a 4-segment unit. The thorax is assumed to be fixed and all other rigid body segments move relative to the thorax.

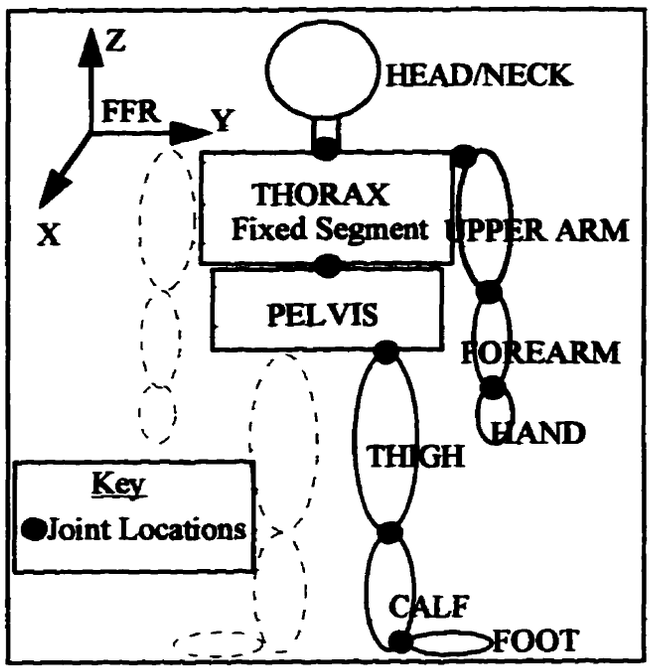


Figure 4.3 Whole Body model used by Gen_Mark for generating marker positions.

4.3 Gen_Mark Software

The Gen_Mark program is written in Borland C++ for Windows. This software requires an input file containing the number of frames of markers to generate, the number of joints that will be modeled, and the joint angles for each joint. The sequence of input for the joint angles is ϕ, θ, and ψ. The order each joint is expected is C7-T1, Lumbar-Sacral, Hip, Knee, Ankle, Shoulder, Elbow and Wrist. Any joints not required would be omitted, but the order would be preserved for the remaining joints. Figure 4.4 shows the beginning of an input file required by the Gen_Mark program.

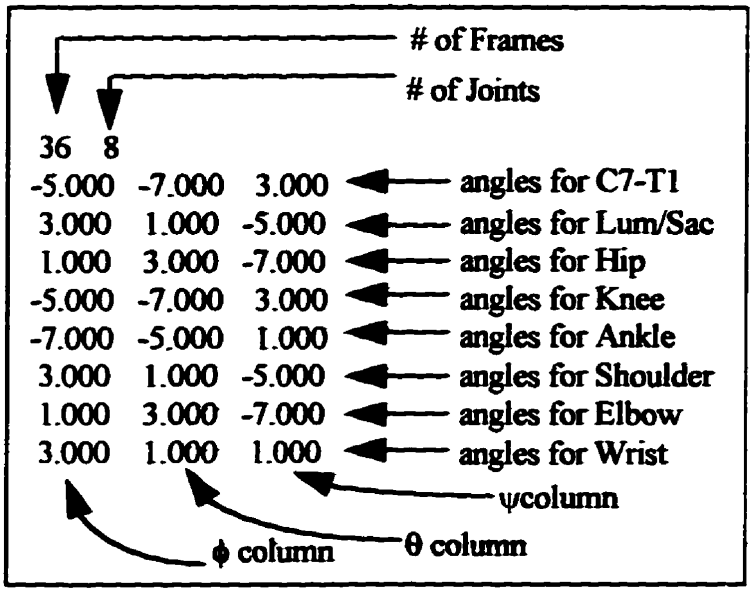


Figure 4.4 Angle file for Gen_Mark program.

The main menu in the Gen_Mark program is shown in Figure 4.5. Under "File" there are three options. "Open Angle File" allows the selection of the file containing the angles used to generate the marker positions. "Save TDC Data" is used for selecting a filename for saving the generated marker positions. If the file selected already exists, it will be written over by the new marker positions. The "Exit" option is used to exit the Gen_Mark program and return to the Windows Program Manager.

The second menu option, "Joints", has just one selection in its pop-up menu. "Choose Joints" creates a Dialog Box listing the possible joint choices. Joint selections are made by selecting the Check Box associated with the joint. The number of joints selected must match the number of joints indicated in the angles input file.

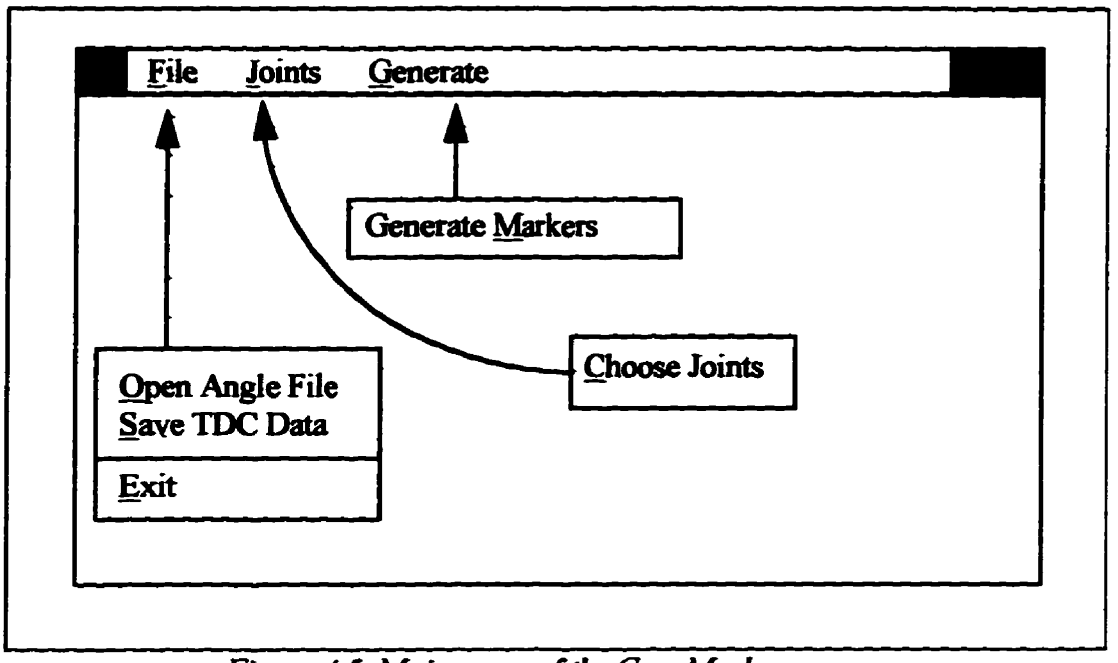


Figure 4.5 Main menu of the Gen_Mark program.

"Generate" is the final menu item. It also has one selection in its pop-up menu. "Generate Markers" requires that the joints be selected and the names of the angle file and the TDC file be chosen. If one of these items is not selected, a message is posted to the Status Bar at the bottom of the screen. The user is then allowed to choose the missing information and then reselect "Generate Markers". A flowchart representing the general method of calculating the marker positions is represented in Figure 4.6

After a test file has been created by Gen_Mark and saved as a TDC file, it can be used for input to the generalized UM^{2}AS software. The error between the calculated Euler angles and the original angles can be calculated. This testing method is discussed in the following section.

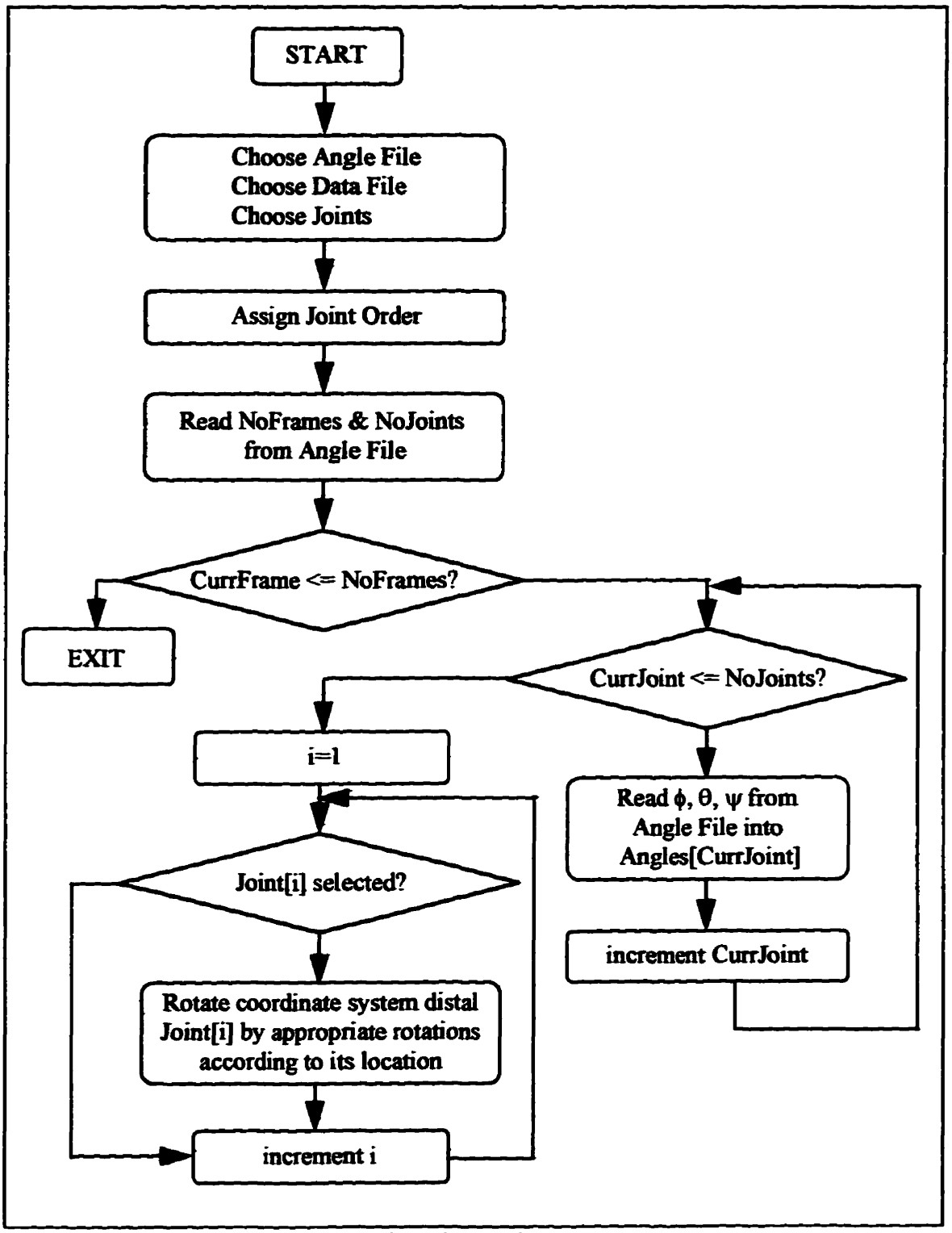


Figure 4.6 Flowchart for marker generation routine.

4.4 Testing the generalized UM²AS software

The generation of test marker coordinates is the first step in testing the generalized UM²AS software. In order to do this a file containing the angle positions for all eight joints was created. Initial angles were set to -2.0°, and the value of each angle was either incremented or decremented by 3° or 5° for each frame. Thirty-six frames were produced for testing purposes. Those angles that were modified by 5° for each frame, passed through a full 180° in a 36 frame cycle.

Three angles, for each of the eight possible joints, were generated using a simple program called TEST8. TEST8 also saves the angles in a second file, in a column format, for easy use with spreadsheet or statistical packages.

The angle file, generated by TEST8, was used as input to the Gen_Mark program. The resulting TDC file became the input file to the generalized UM²AS program. The calculated Euler angles were compared to the angles used to generate the marker coordinates. The relative error was calculated for each angle, at each frame.

$$Rel.Err = \left(\frac{(calc.value - corr.value)}{corr.value} \right) \cdot 100\% \qquad 4.10$$

The resulting relative error values ranged from -2.5% - 1.1% with a mean value of -0.001 and a standard deviation of 0.105.

The relative error plot of C7-T1 is presented in Figure 4.7. This plot is typical of the plot that results at other joints as well. Appendix D contains a complete presentation of the relative error plots, as well as diagrams indicating the range of each joint angle for the test file.

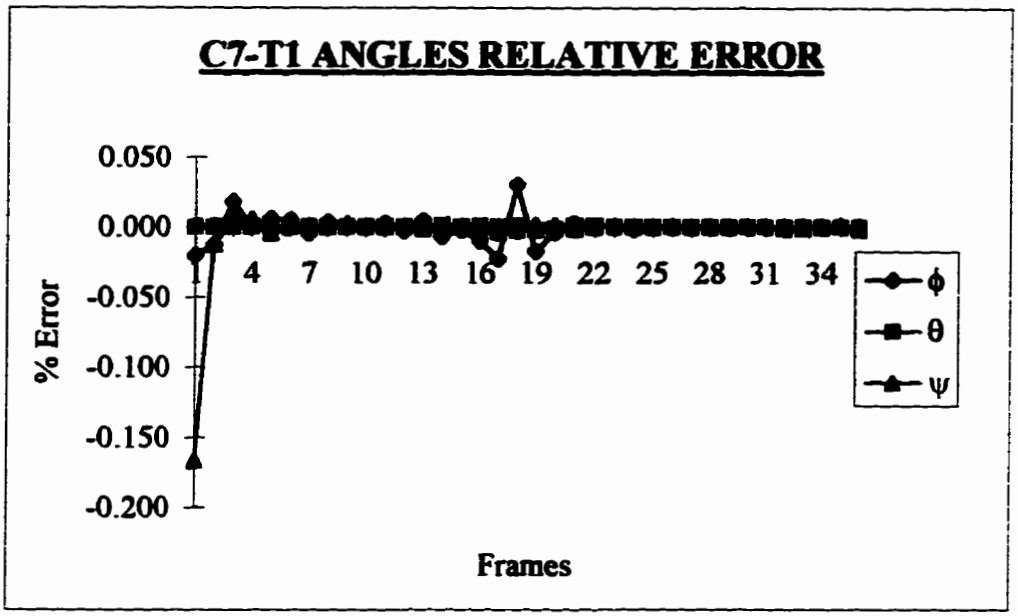


Figure 4.7 Relative Error plot for C7-T1 angles.

Since this testing method utilizes a computer model, it is expected that the errors be very low, which they are. These errors can easily be accounted for by the round-off error inherent in computer calculations.. However, since ψ generates a higher error than θ and ϕ in each joint, the reason for this was examined. The answer appears to lie in the order of calculation. When marker locations are generated, the rotation about the y-axis, ψ, is the last rotation performed. Any round-off errors already inherent at this point are compounded by the final rotation. Thus, the y-axis, rotation is most affected by the error. Additionally, the Euler angle ψ is not calculated in isolation as θ is. Therefore the round-off error resulting from the calculation of θ and $cos(\theta)$, is included in the calculation of both ϕ and ψ, increasing their errors.

4.5 Potential Experimental Problems

The testing performed in the preceding section is essential an idealized situation. Marker locations, in experimental situations, can be affected by a number of factors. Sometimes the calculated centroids are distorted due to a portion of the marker being obscured in one or the cameras. Centroids can also be distorted by reflections off another object in the recording environment. Every effort is made to minimize these factors during experiments, but there is usually at least several frames in each experiment that will be affected by such factors.

Even if marker centroids were always found perfectly, there are several other factors that will affect the calculated three-dimensional coordinates. Giles [Giles, 1993] measured a relative error as high as -1.18% in the calculated distances between two markers spaced approximately 30.5 cm. apart. This was termed to be camera error and is a result of space quantization, lens distortion and marker distortion combined. It can be expected that these kinds of errors will also be inherent in the 3-D marker coordinates eventually used by the generalized UM^2AS system.

The final potential source of error that can be considered is error due to skin motion over the bony masses in the body. The motion required to be measured is the rotations in the joints. However, the motion measured is the motion of the skin of the body, since this is the surface to which marker systems are attached. Therefore, whenever the skin does not move simultaneously with the bony masses, there will result an error due to skin movement.

Chapter Five
Summary and Conclusions

The original UM^{2}AS system has been a very useful tool for the study of upper limb movement. However, since its purpose was intended to be for the study of upper limb movements, it is practically impossible to use this tool in other functional motion studies. For this reason, the goal of this study was the development of a generalized joint motion analysis system. The generalized system was to allow the examination of joint motion of the whole body. Because the system is to be used in clinical studies, the number of markers needed to be reduced as much as possible.

In answer to these goals, a generalized motion analysis system has been developed. The software for this analysis system is written in Borland C++ for Windows and runs in a Windows environment. This analysis system allows the study of 8 joints in the body, or effectively one side of the body. It is capable of the examination of three rotations in each joint, using a spherical joint model, although fewer rotations may be selected if desirable. Because the software is written in a standard Windows format, it is user friendly and easily learned.

In addition to the generalized UM^{2}AS system, a program for generating three dimensional marker coordinates was also developed in Borland C++ for Windows. This marker generation program simulates marker positions expected by the UM^{2}AS system. It was developed for use in testing the generalized UM^{2}AS system. Marker positions were produced by the marker generation program and run through the generalized UM^{2}AS system. The generalized UM^{2}AS system was able to reproduce the Euler angles, used to

produce the marker positions, to within a relative error of -2.4% to 1.1%. The marker generation program is also based in a standard Windows format and is easily and intuitively used.

With the idealized marker positions produced by the Gen_Mark program, the generalized UM^{2}AS software seems to produce very valid results. Based on these tests, it is safe to assume that for valid input, the output from this system is valid. However normal clinical input is not usually idealized in this manner. It would be useful to run an experiment on a subject and try regular clinical input on the generalized UM^{2}AS system. Indeed, this would be a very desirable endeavour prior to its use in a clinical study.

In addition to testing the generalized UM^{2}AS system in a clinical setting, there are several other recommendations for improvement to this system. The first of these would be concerned with the usefulness of the software. The software currently produces time (or frames) versus angle plots. It could be useful for clinicians if these plots could be sent to the printer as well as the screen. Secondly, stick-figure representations of the upper limb motion was included in the very first version of the UM^{2}AS software. These stick-figures were plotted in each of the three body based planes, frontal, sagittal, and horizontal. Such animated diagrams could be a useful tool in a clinical setting and it would be useful to add this feature to the generalized software.

In addition to the mentioned improvements of this software, is the consideration of the state of currently available technology. The software is written for a Windows environment on a PC. If the UM^{2}AS system is to stay in a PC based environment, the software needs to be converted to a Windows 95 type environment. This task may be

relatively straightforward if newer versions of Borland C++ are downwardly compatible. On the other hand, if the UM^{2}AS system moves to an Apple environment, to take advantage of the integrated image digitization capabilities of an Apple system, most of the user-interfacing sections of this software will have to be re-written.

In conclusion, the software developed for this project gives very good results for the idealized data on which it was tested. The goal of the project, the development of a generalized joint motion analysis system, has been fulfilled. The software is user friendly and easily learned as it has a standard Windows, mouse/menu driven user interface. The testing software, Gen_Mark produces marker positions for testing this generalized UM^{2}AS software, but could be modified to test other marker configurations as well. Both the generalized UM^{2}AS system and the Gen_Mark program have the potential to become useful clinical tools.

REFERENCES:

[1.] Blacharski, P. A., Somerset, J. H. and Murray, D. G., A three-dimensional study of the kinimatics of the human knee. J. of Biomechanics, Vol. 8, pp 375-384, 1975.

[2.] Chao, Edmund Y. S., Justification of triaxial goniometer for the measurment of joint rotation. J. of Biomecanics, Vol. 13, pp 989-1006, 1980.

[3.] Crisco III, Joseph J., Chen, Xingben, Panjabi, Manohar M. and Wolfe, Scott W., Optimal Marker Placement for Calculating the Instantaeous Centre of Rotation. J. of Biomechanics, Vol. 27, No. 9, pp 1183-1187, 1994.

[4.] Giles, John M. Development of and Integrated Environment for Upper Limb Motion Analysis Studies., M.Sc. Thesis, University of Manitoba, 1993.

[5.] Grood, E. S. and Suntay, W. J. A joint coordinate system for the clinical description of three-dimensioanl motions: Application to the knee. J. of Biomechanical Engineering, Vol. 105, pp 136-144, 1983.

[6.] Holzreiter, St. Calculation of the instantaneous centre of rotation for a rigid body. J. of Biomechanics, Vol. 24, No. 7, pp 643-647, 1991.

[7.] Kinzel, G. L. and Gutkowski, L. J. Joint models, degrees of freedom, and anatomical motion measurement. J. of Biomechanical Engineering, Vol. 105, pp 55-62, 1983.

[8.] Kinzel, G. L., Hall, A. S. and Hillberry, B. M. Measurement of the total motion between two body segments-I. Analytical Developement. J.of Biomechanics, Vol. 5. pp. 93-105, 1972.

[9.] Panjabi, M. M., Centers and angles of rotation of body joints: a study of errors and optimization., J. of Biomechanics, Vol. 12, pp 911-920, 1979.

[10.] Panjabi, Manohar M. and Goel, Vijay K., Errors in kinematic parameters of a planar joint: Guidelines for optimal experimental design., J. of Biomechanics, Vol. 15, No.7, pp 537-544, 1982.

[11.] Pennock, G. R. and Clark, K. J., An anatomy-based coordinate system for the description of the kinematic displacements in the human knee. J. of Biomechanics, Vol. 23, No. 12, pp 1209-1218, 1990.

[12.] Safaee-Rad, Reza, Shwedyk, Edward, Quanbury, Arthur O. and Cooper, Juliette E. Normal functional range of motion of the upper limb joints during performance of three feeding activities. Arch. Phys. Med. Rehahil., Vol. 71, June 1990.

[13.] Safaee-Rad, Reza, Functional Human Arm Motion Study with a New 3-D Measurement System (VCR-PIPEZ-PC), M.Sc. Thesis, University of Manitoba, 1987.

[14.] Schalkoff, Robert J., Digital Image Processing and Computer Vision, John Wiley and Sones Inc., New York, 1989.

[15.] Small, C. F., Bryant, J. T. and Pichora, D. R., Rationalization of kinematic descriptors for three-dimensional hand and finger motion. J. of Biomedical Engineering, pp. 133-141, 1992.

[16.] Small, C.F., Pichora, D. R., Bryant, J. T. and Griffiths, P. M. Precision and accuracy of bone landmarks in characterizing hand and wrist poisition. J. of Biomedical Engineering, pp. 371-378, 1994.

[17.] Soudan, Karel, Van Audekercke, Remi and Martens, Marc. Methods, difficulties and inaccuracies in the study of human joint kinematics and pathokinematics by the instant axis concept. Example: The knee joint. J. of Biomechanics, Vol. 12, pp 27-33, 1979.

[18.] Spiegelman, Jeffery J. and Woo, Savio L. Y., A rigid-body method for finding centers of rotation and angular displacements for planar joint motion. J. of Biomechanics, Vol. 20, No. 7, pp 715-721, 1987.

[19.] Wang, Mei, Bryant, J. Tim, and Dumas, Genevieve A. A new technique for in vitro measurment of small 3-D joint motion. Engineering Systems Design and Analysis, Vol 4, 1994.

[20.] Youm,Y. and Yoon, Y. S., Analytical developement in investigation of wrist kinematics. J. of Biomechanics, Vol. 12, pp. 613-621, 1979.

Appendix A

Alternative Rotation Matrices For Less Than Three Rotations

There are six possible alternative rotation matrices used when less than 3 rotations are selected in a joint The correct matrix is chosen by the generalized UM^{2}AS program based on the rotations selected in each joint.

Equation A.1 is used if only the x-rotation is selected.

$$[\mathbf{R}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \qquad \text{A.1}$$

Equation A.2 is used if only the y-rotation is selected.

$$[\mathbf{R}] = \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix} \qquad \text{A.2}$$

Equation A.3 is used if only the z-rotation is selected.

$$[\mathbf{R}] = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \text{A.3}$$

Equation A.4 is used if the x and z rotation are selected.

$$[\mathbf{R}] = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$[\mathbf{R}] = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi \cdot \cos\theta & \cos\theta \cdot \cos\phi & \sin\theta \\ \sin\phi \cdot \sin\theta & -\sin\theta \cdot \cos\phi & \cos\theta \end{bmatrix} \qquad \text{A.4}$$

Equation A.5 is used if the y and x rotations are selected.

$$[\mathbf{R}] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix}$$

$$[\mathbf{R}] = \begin{bmatrix} \cos\psi & \sin\theta \cdot \sin\psi & -\sin\psi \cdot \cos\theta \\ 0 & \cos\theta & \sin\theta \\ \sin\psi & -\sin\theta \cdot \cos\psi & \cos\psi \cdot \cos\theta \end{bmatrix}$$

A.5

Equation A.6 is used if the y and z rotations are selected.

$$[\mathbf{R}] = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\psi & 0 & \sin\psi \\ 0 & 1 & 0 \\ -\sin\psi & 0 & \cos\psi \end{bmatrix}$$

$$[\mathbf{R}] = \begin{bmatrix} \cos\psi \cdot \cos\phi & \sin\phi \cdot \cos\psi & -\sin\psi \\ -\sin\phi & \cos\phi & 0 \\ -\sin\psi \cdot \cos\phi & -\sin\psi \cdot \sin\phi & \cos\psi \end{bmatrix}$$

A.6

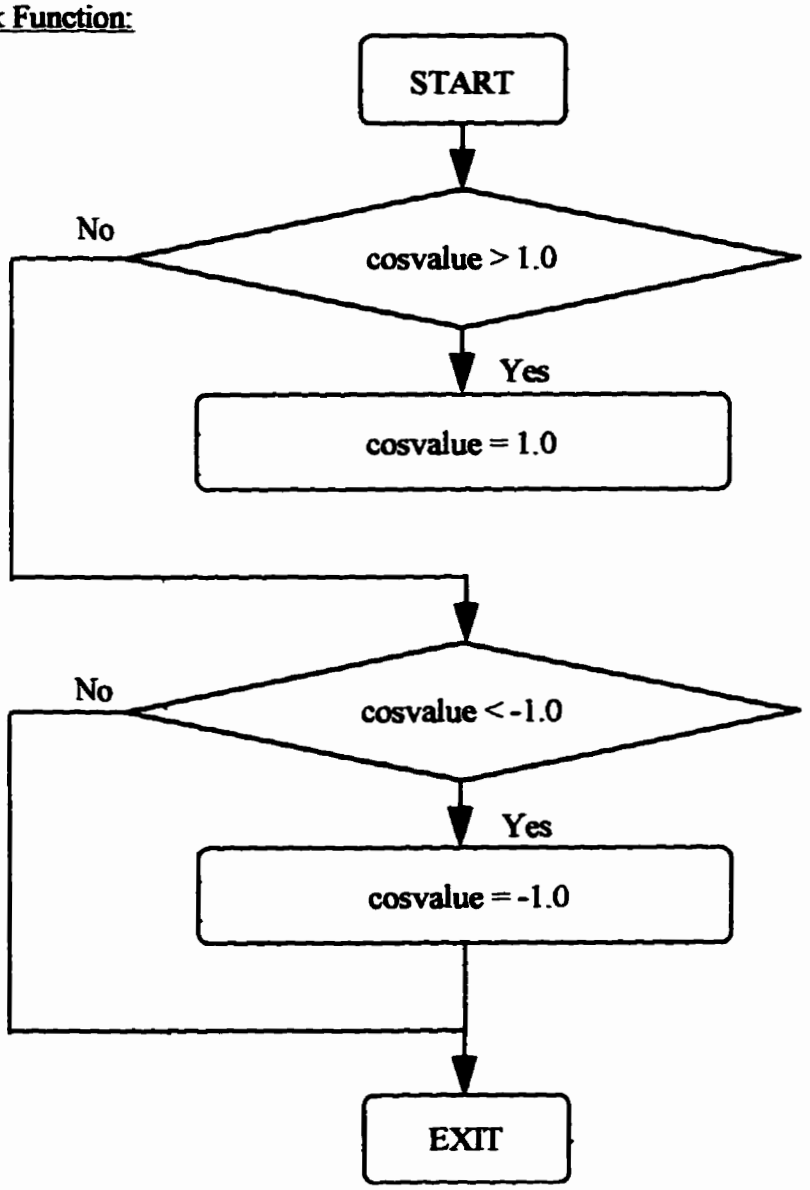
Appendix B

Flowcharts for Generalized UM^{2}AS Software

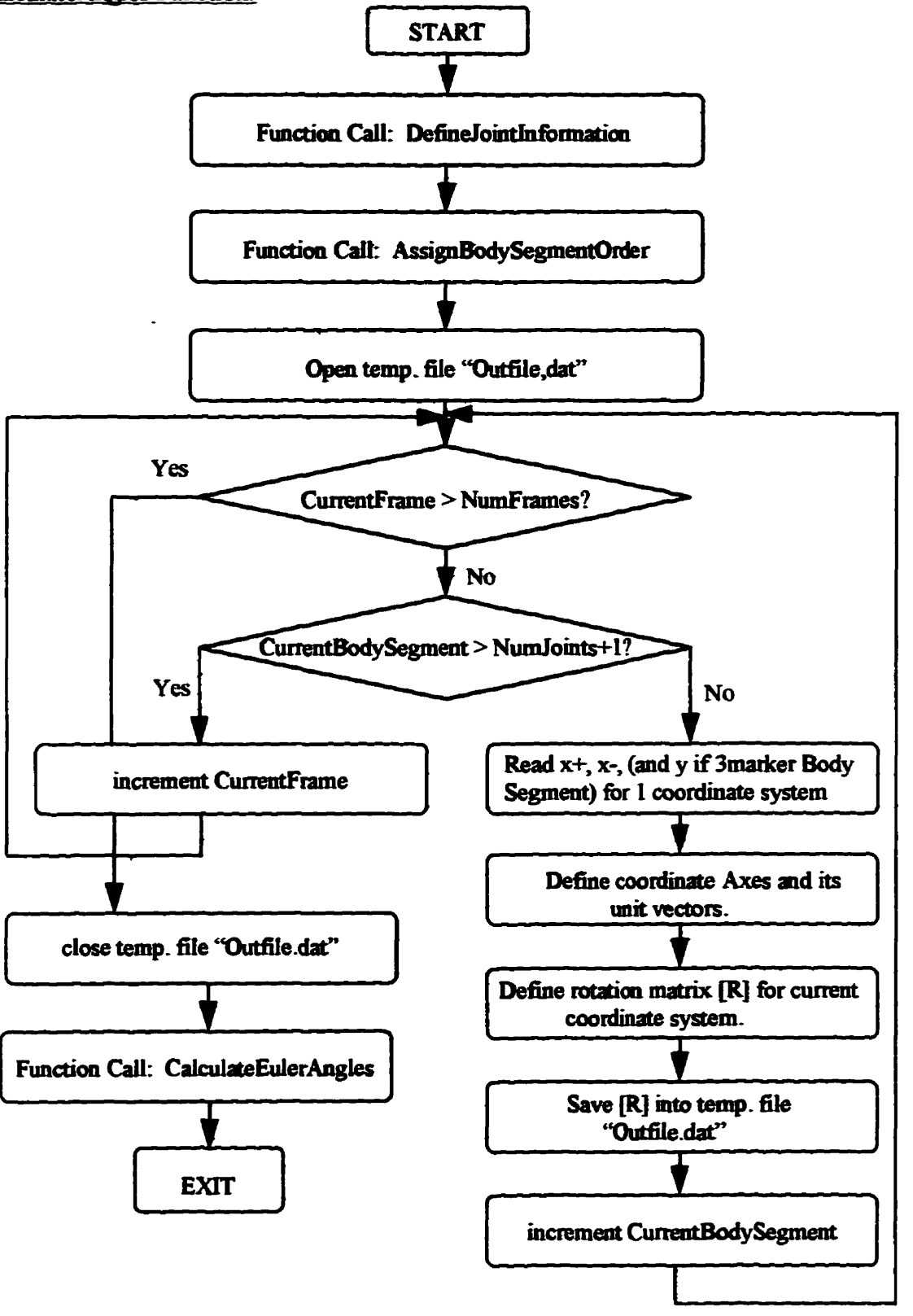
CALCANGL FUNCTIONS:

The following functions are contained in the CALCANGL file of the UM²AS software. This file contains the functions used to calculate the Euler angles.

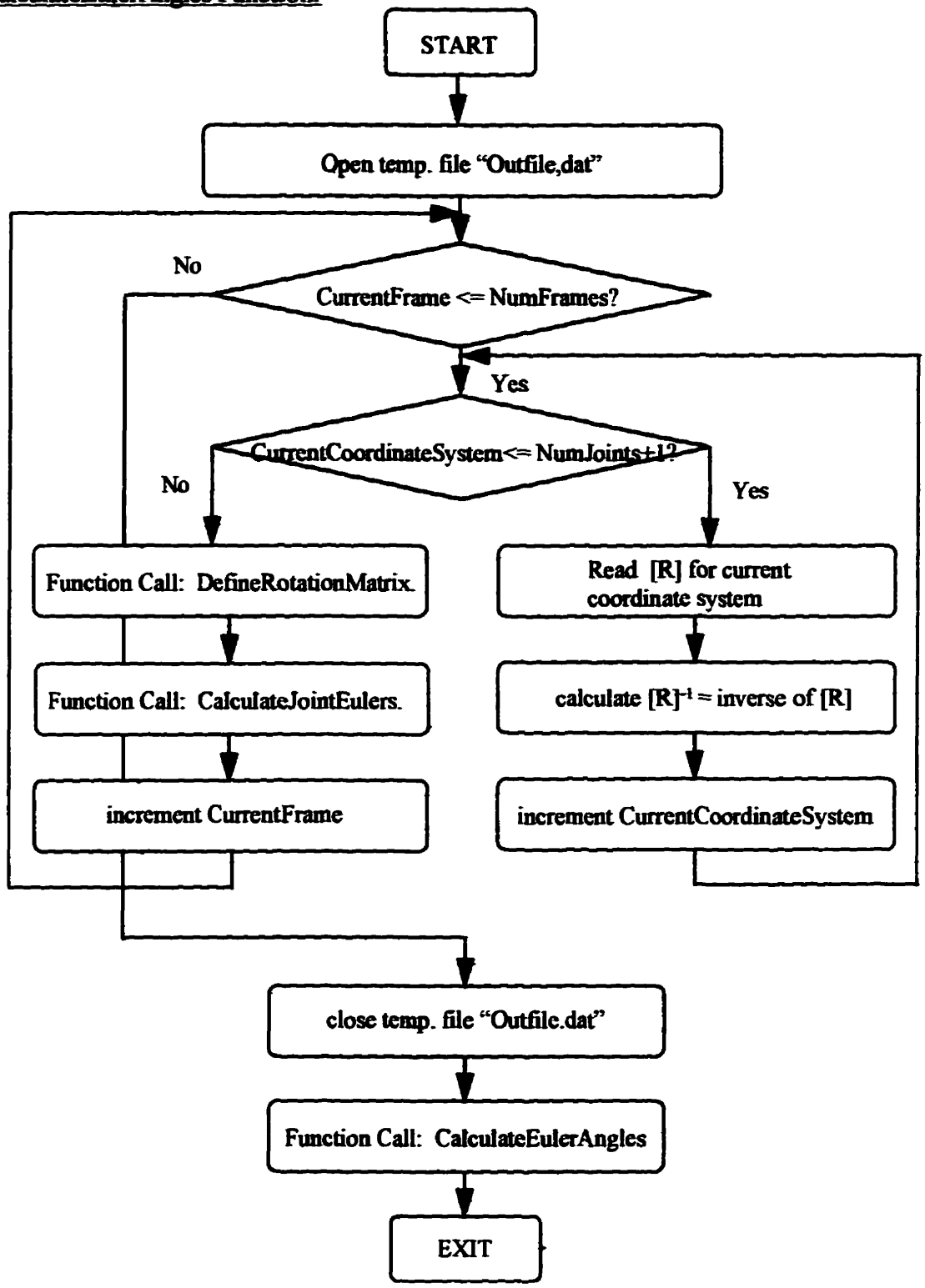
DomainCheck Function:



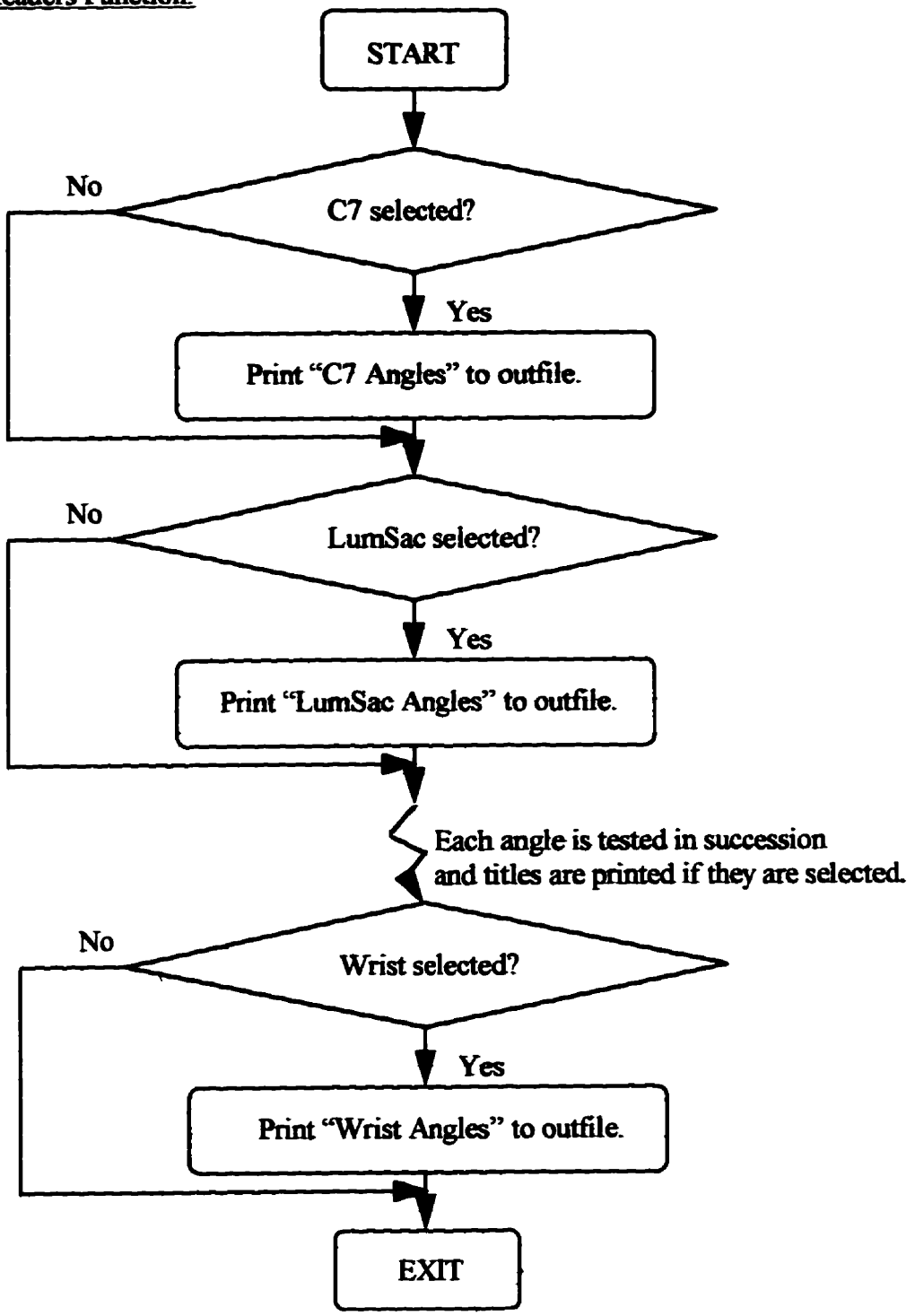
CalculateAngles Function:

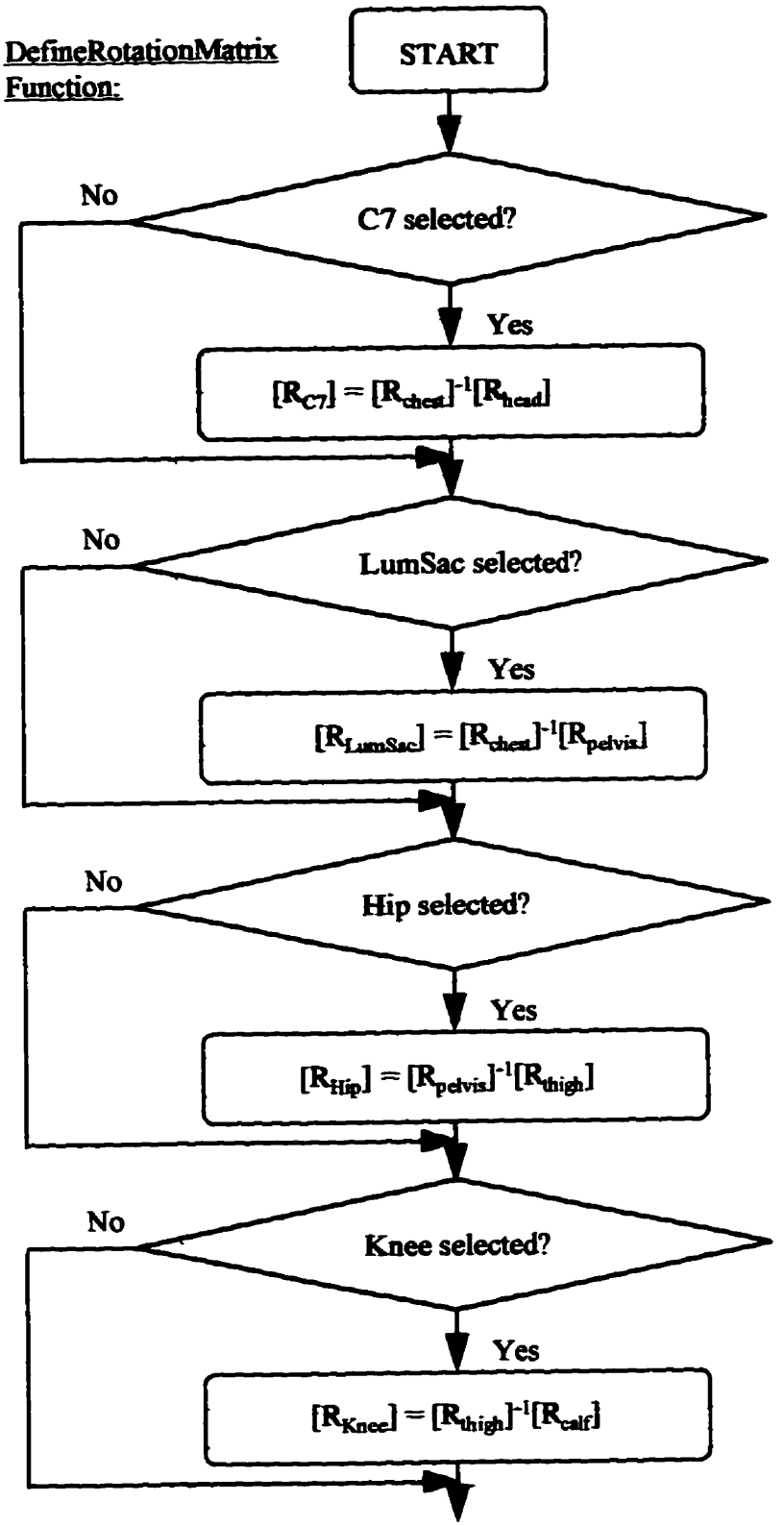


CalculateEulerAngles Function:



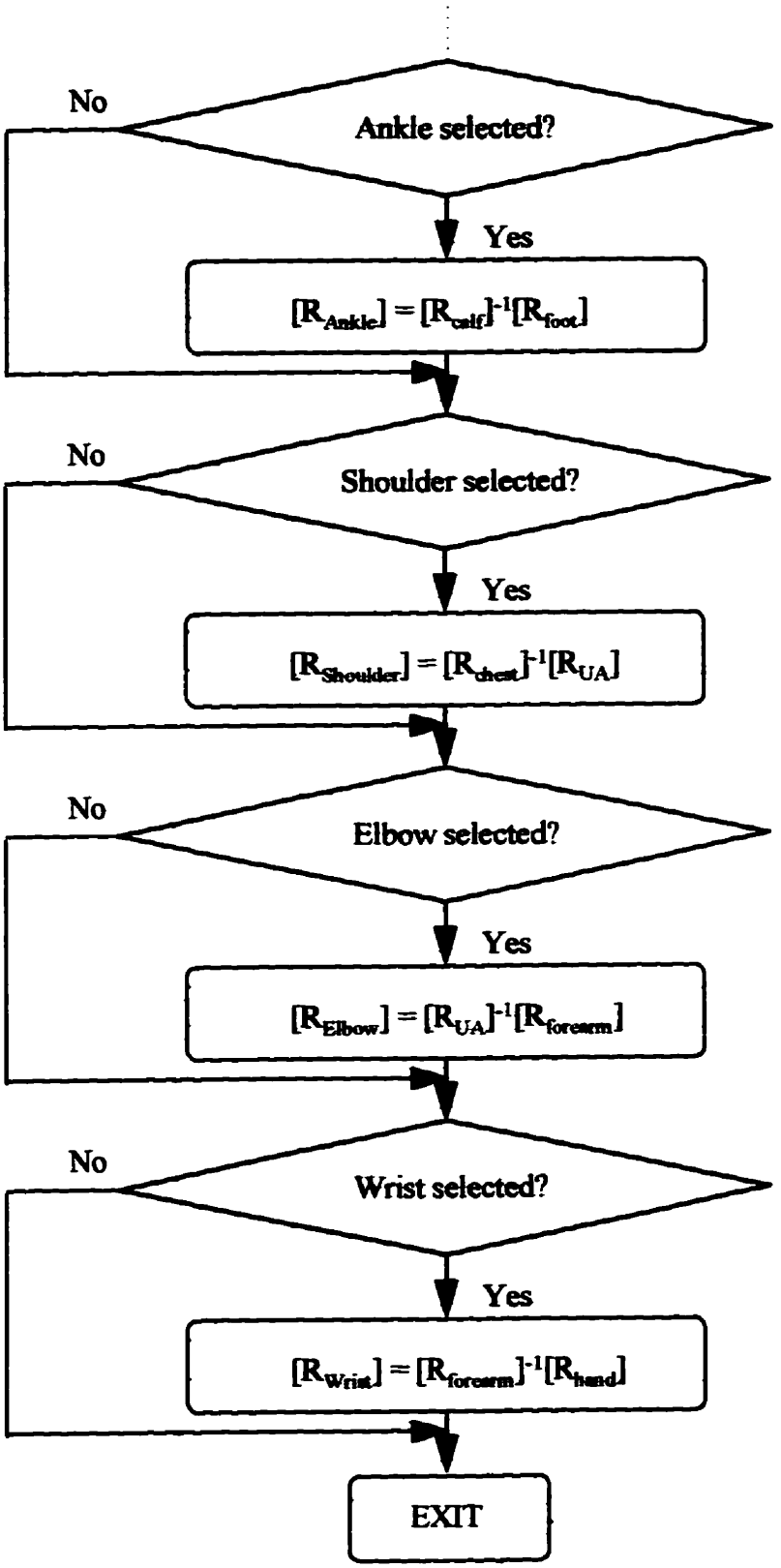
PrintHeaders Function:

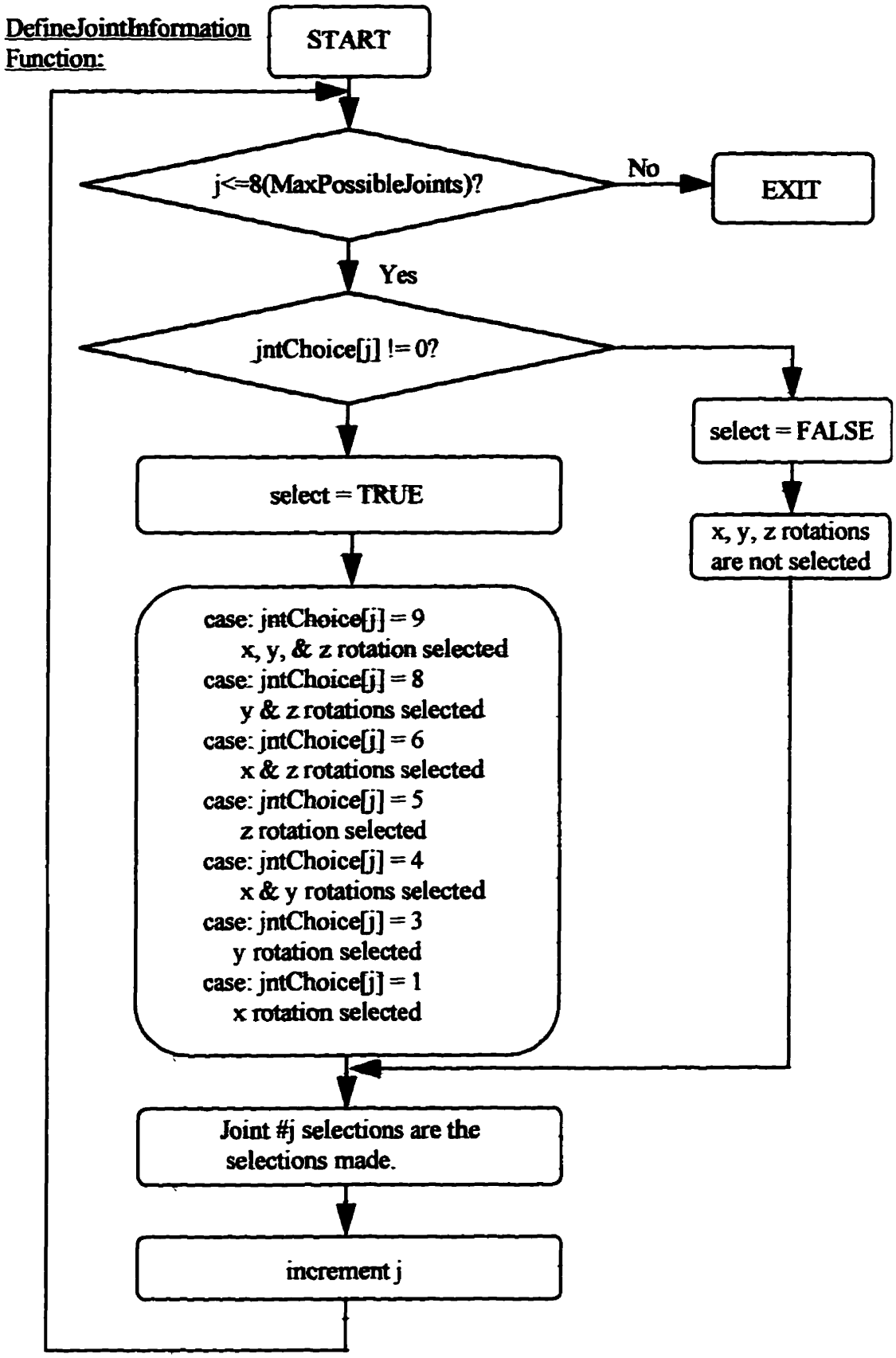




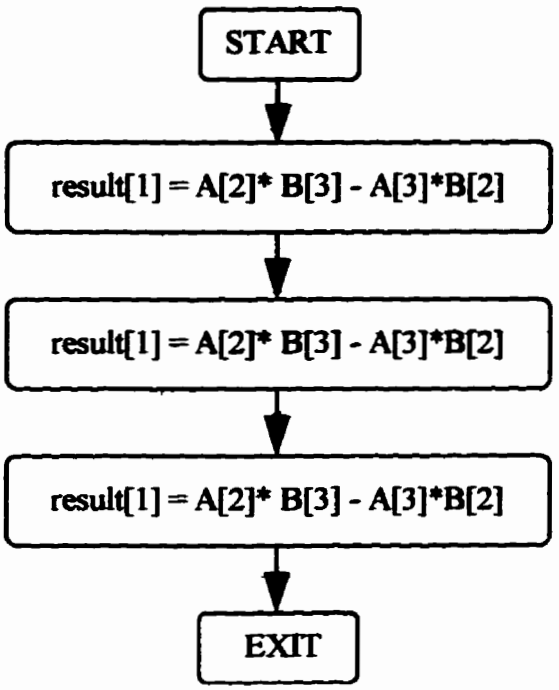
continued on next page

continued from previous page

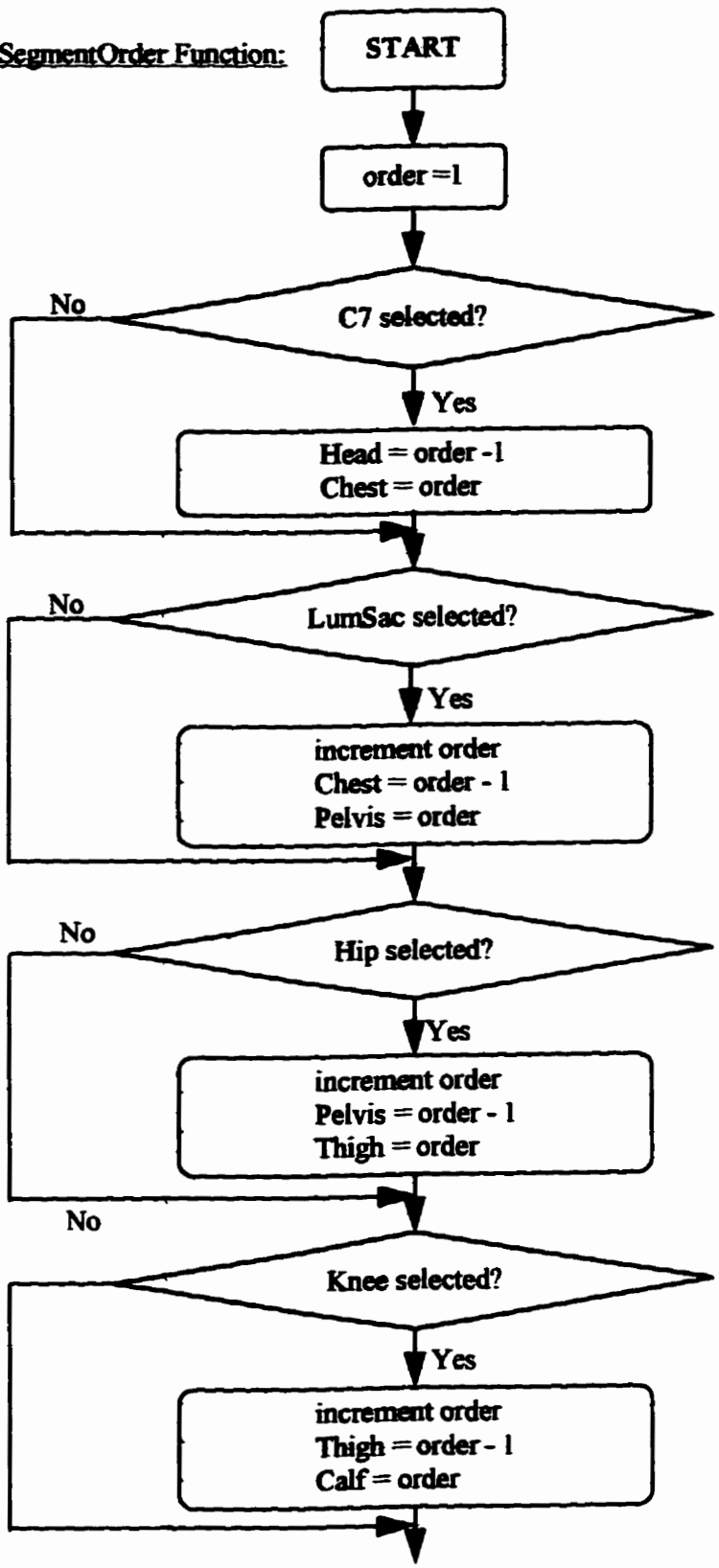



DefineJointInformation Function:
START
j<=8(MaxPossibleJoints)?
No
EXIT
Yes
jntChoice[j] != 0?
select = FALSE
x, y, z rotations are not selected
select = TRUE
case: jntChoice[j] = 9
x, y, & z rotation selected
case: jntChoice[j] = 8
y & z rotations selected
case: jntChoice[j] = 6
x & z rotations selected
case: jntChoice[j] = 5
z rotation selected
case: jntChoice[j] = 4
x & y rotations selected
case: jntChoice[j] = 3
y rotation selected
case: jntChoice[j] = 1
x rotation selected
Joint #j selections are the selections made.
increment j

CrossProduct Function:

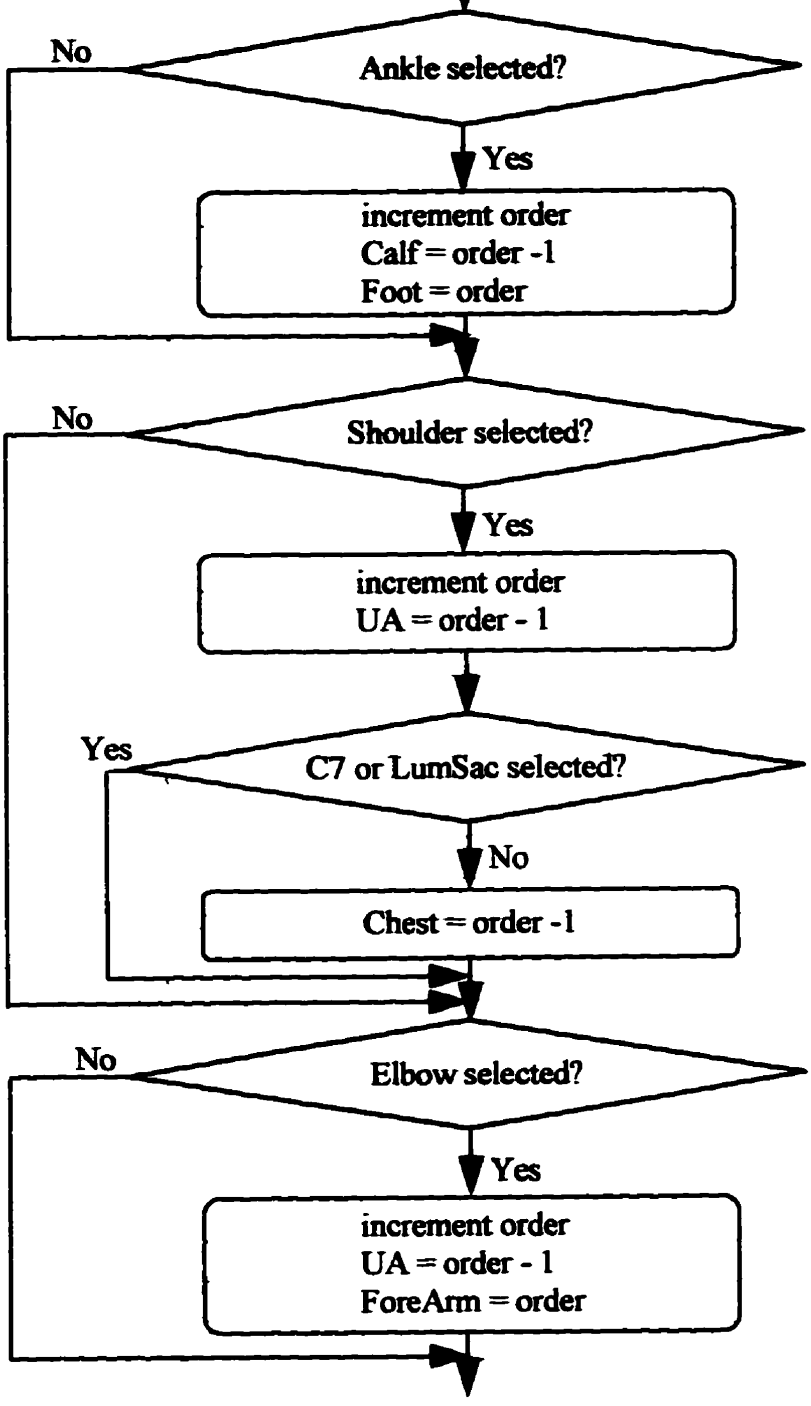


AssignBodySegmentOrder Function:

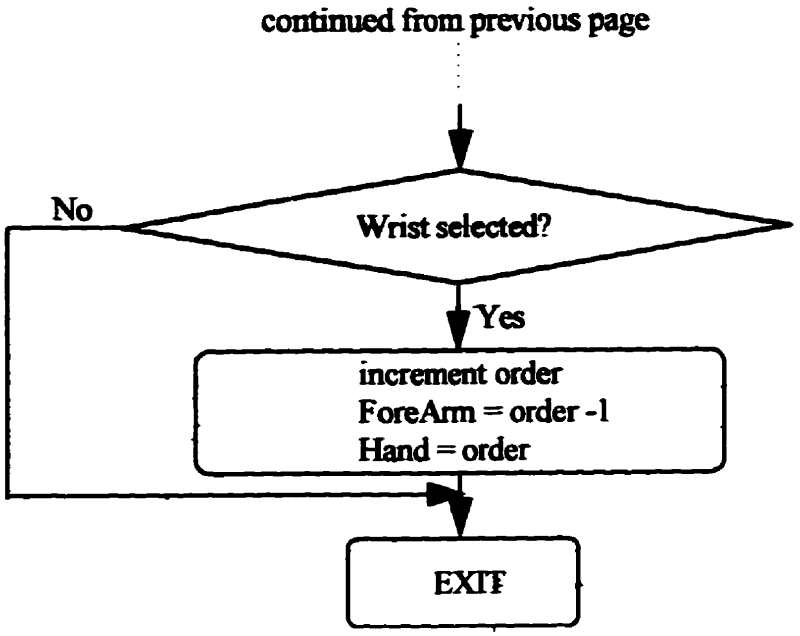


continued on next page

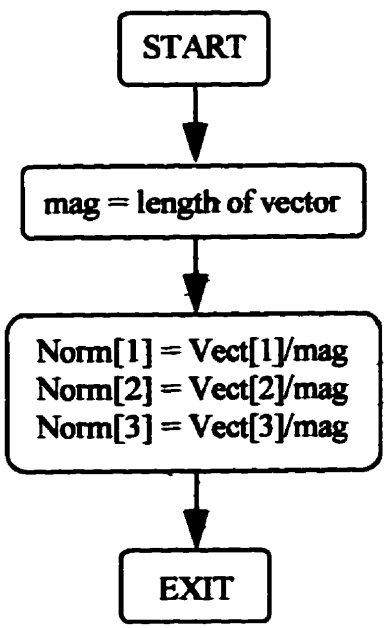
continued from previous page

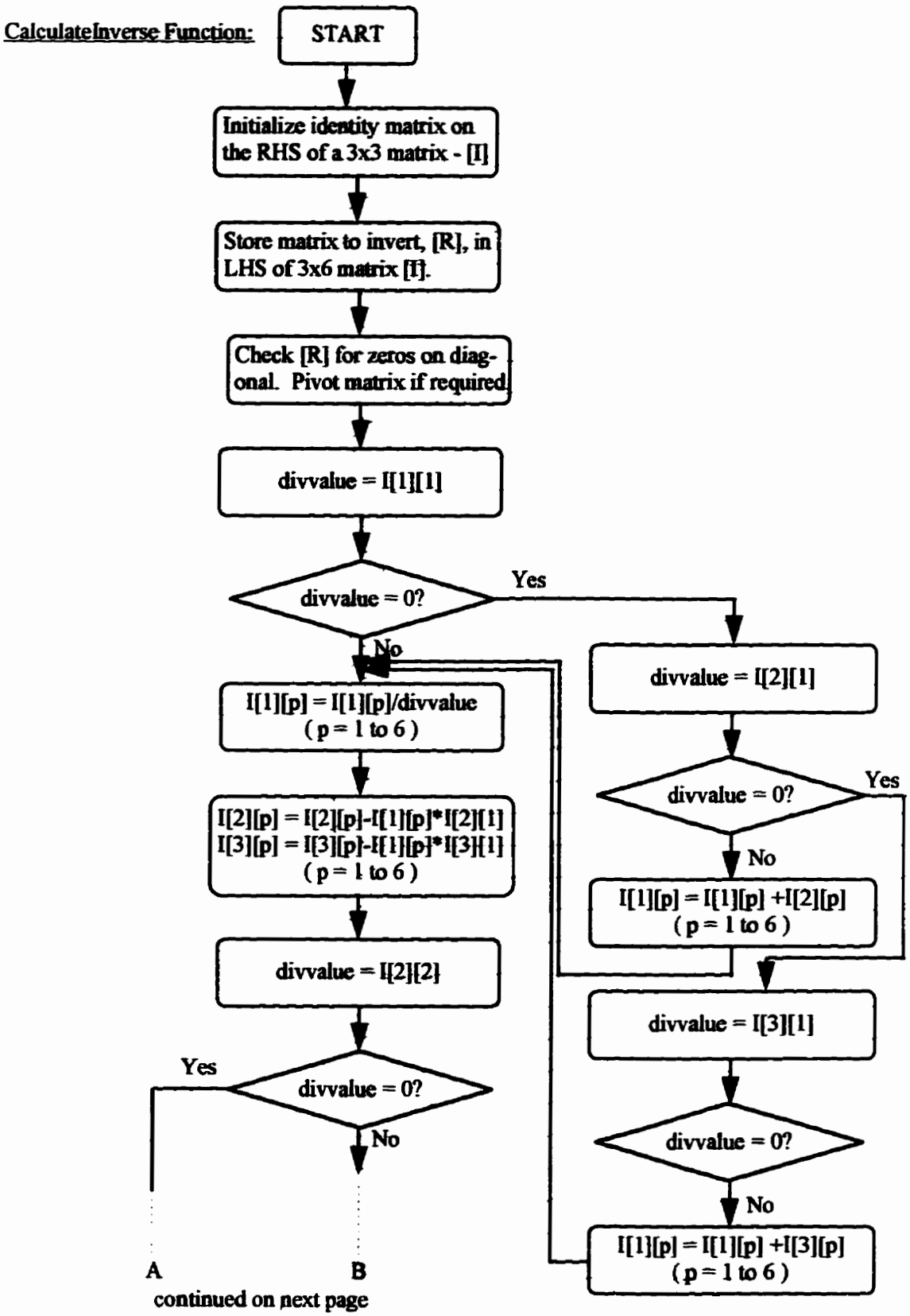


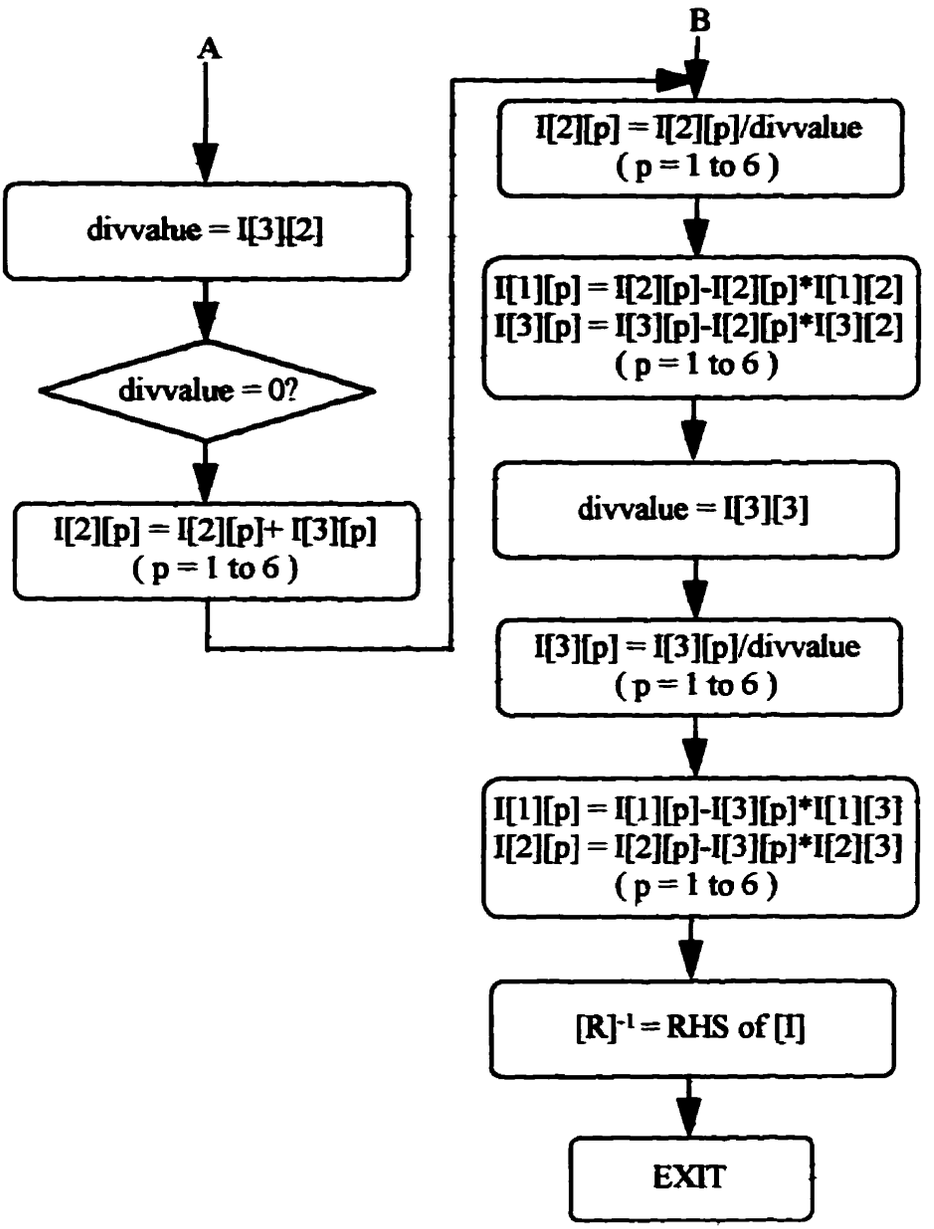
continued on next page



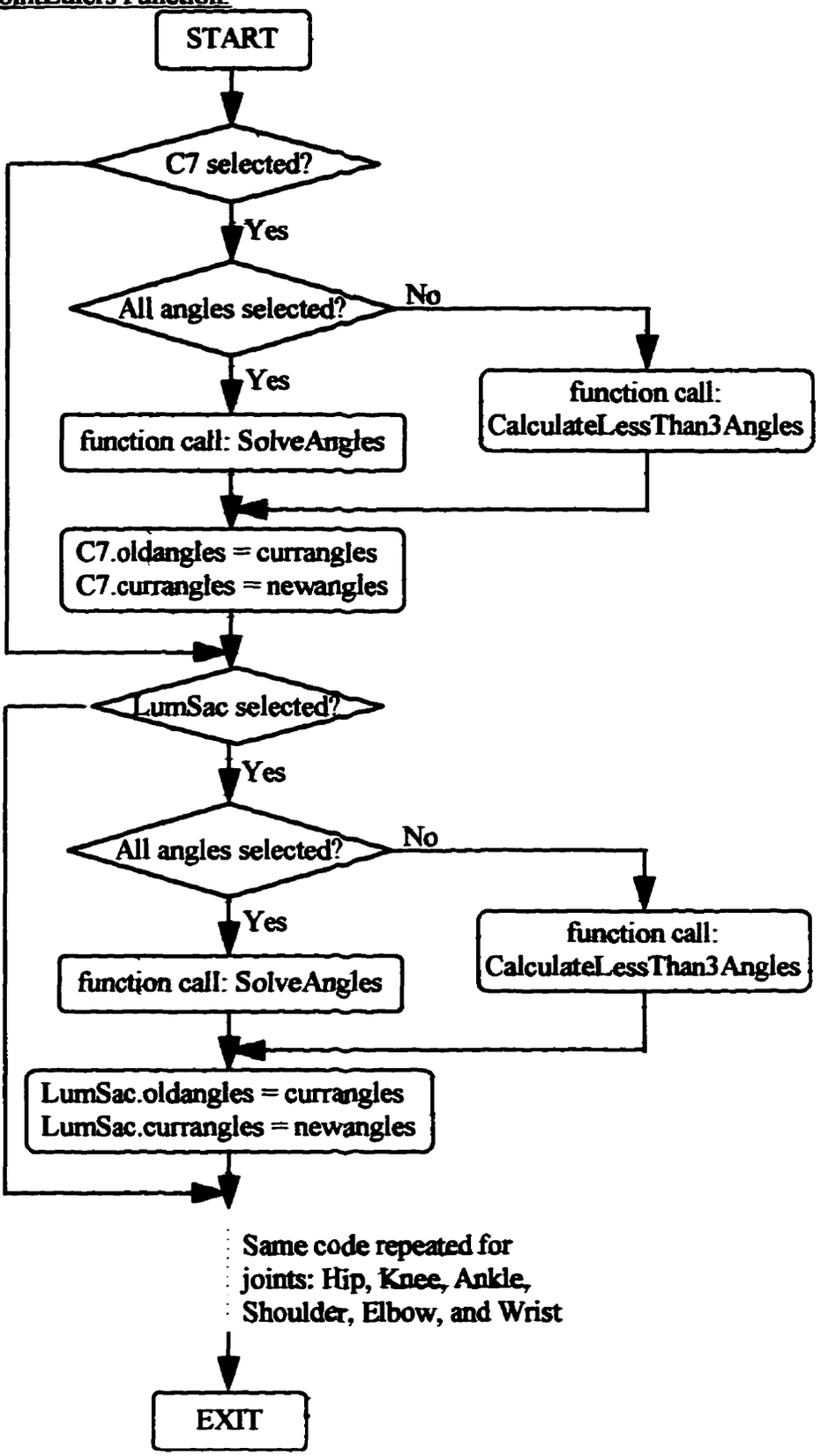
NormalizeVector Function:



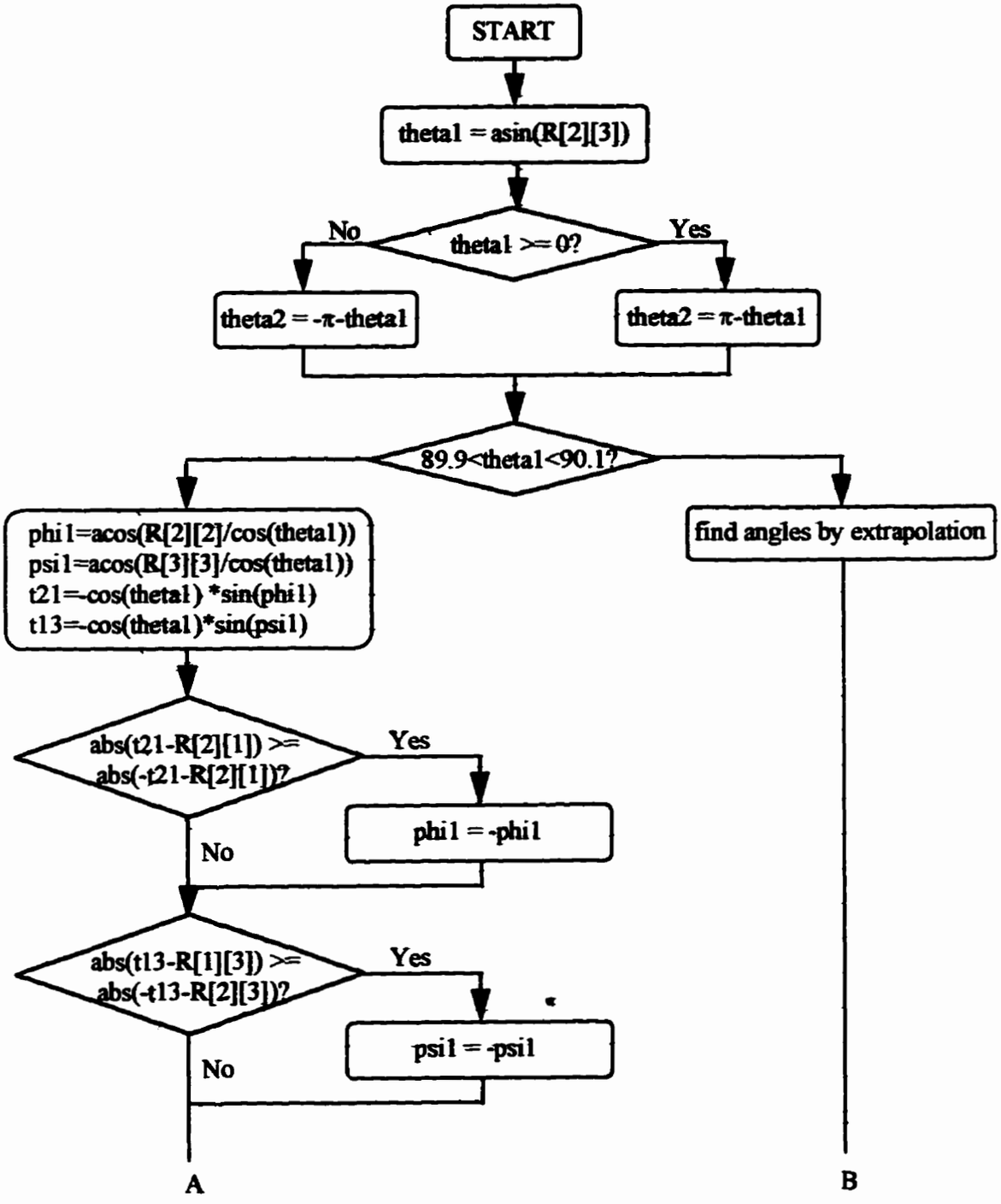

CalculateInverse Function:
START
Initialize identity matrix on the RHS of a 3x3 matrix - [I]
Store matrix to invert, [R], in LHS of 3x6 matrix [I].
Check [R] for zeros on diagonal. Pivot matrix if required
divvalue = I[1][1]
divvalue = 0?
Yes
No
I[1][p] = I[1][p]/divvalue
(p = 1 to 6)
I[2][p] = I[2][p]-I[1][p]*I[2][1]
I[3][p] = I[3][p]-I[1][p]*I[3][1]
(p = 1 to 6)
divvalue = I[2][2]
divvalue = 0?
Yes
No
A
B
continued on next page
divvalue = I[2][1]
divvalue = 0?
Yes
No
I[1][p] = I[1][p] +I[2][p]
(p = 1 to 6)
divvalue = I[3][1]
divvalue = 0?
No
I[1][p] = I[1][p] +I[3][p]
(p = 1 to 6)


A
divvalue = I[3][2]
divvalue = 0?
I[2][p] = I[2][p]+ I[3][p]
(p = 1 to 6)
B
I[2][p] = I[2][p]/divvalue
(p = 1 to 6)
I[1][p] = I[2][p]-I[2][p]*I[1][2]
I[3][p] = I[3][p]-I[2][p]*I[3][2]
(p = 1 to 6)
divvalue = I[3][3]
I[3][p] = I[3][p]/divvalue
(p = 1 to 6)
I[1][p] = I[1][p]-I[3][p]*I[1][3]
I[2][p] = I[2][p]-I[3][p]*I[2][3]
(p = 1 to 6)
[R]-1 = RHS of [I]
EXIT

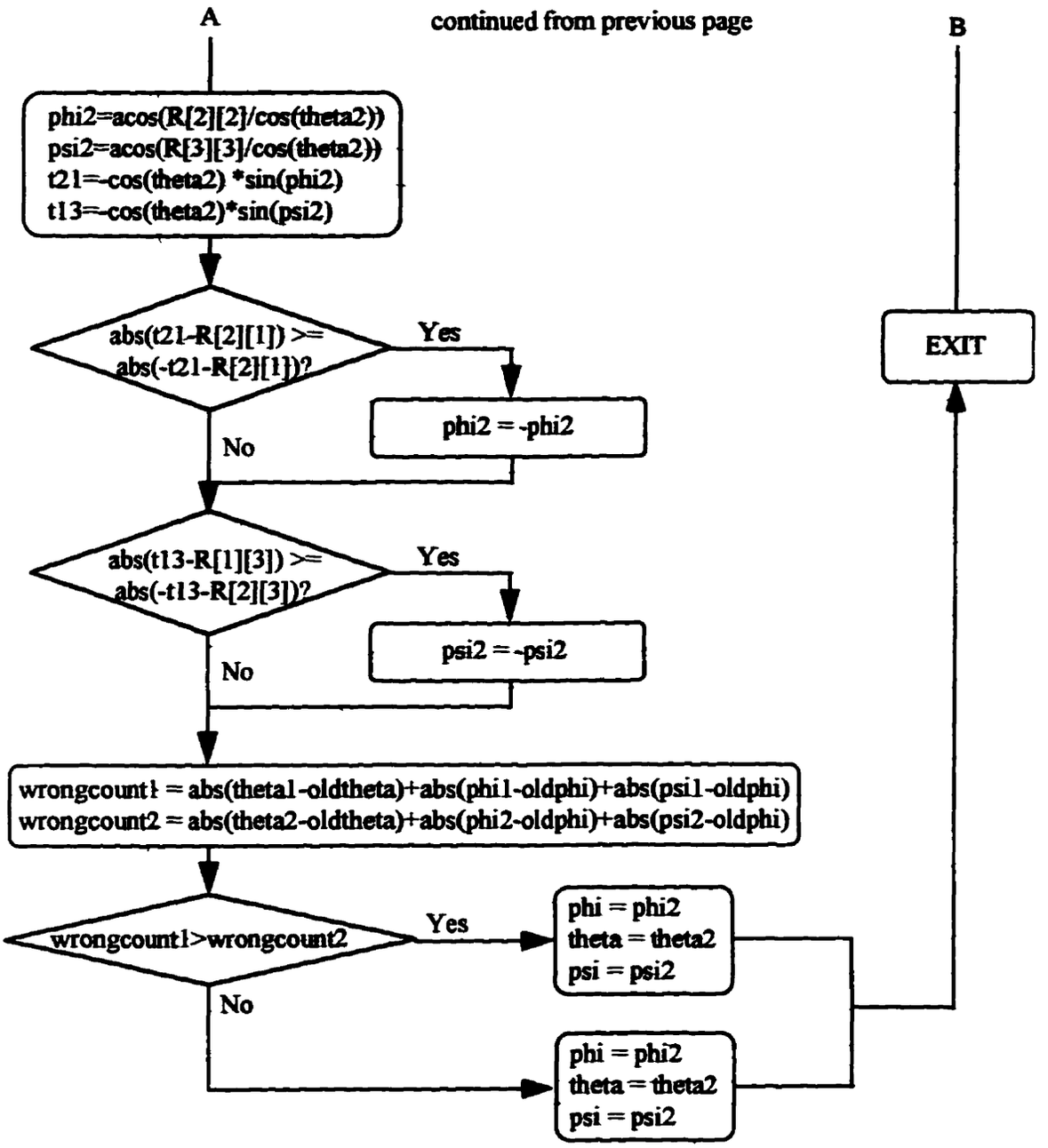
CalculateJointEulers Function:



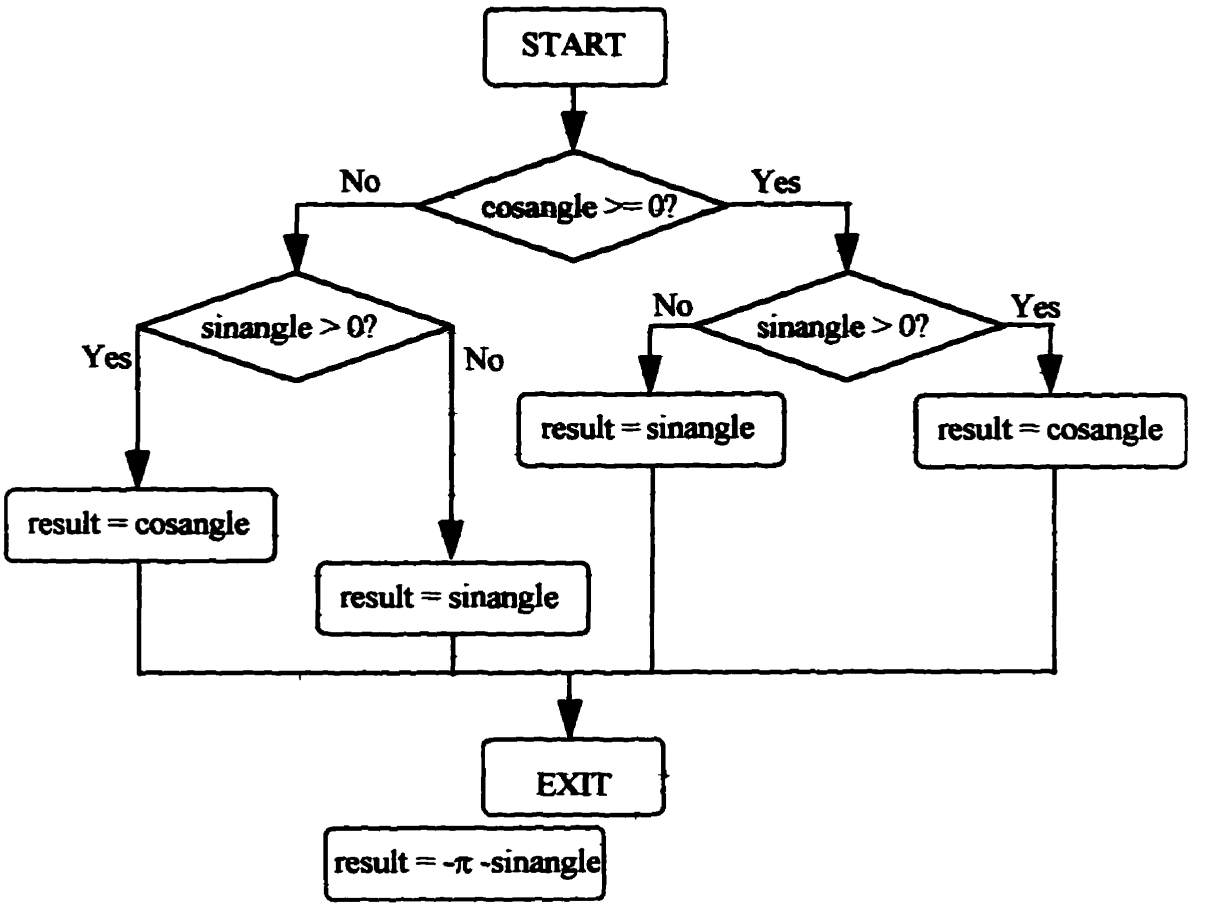
SolveAngles Function:

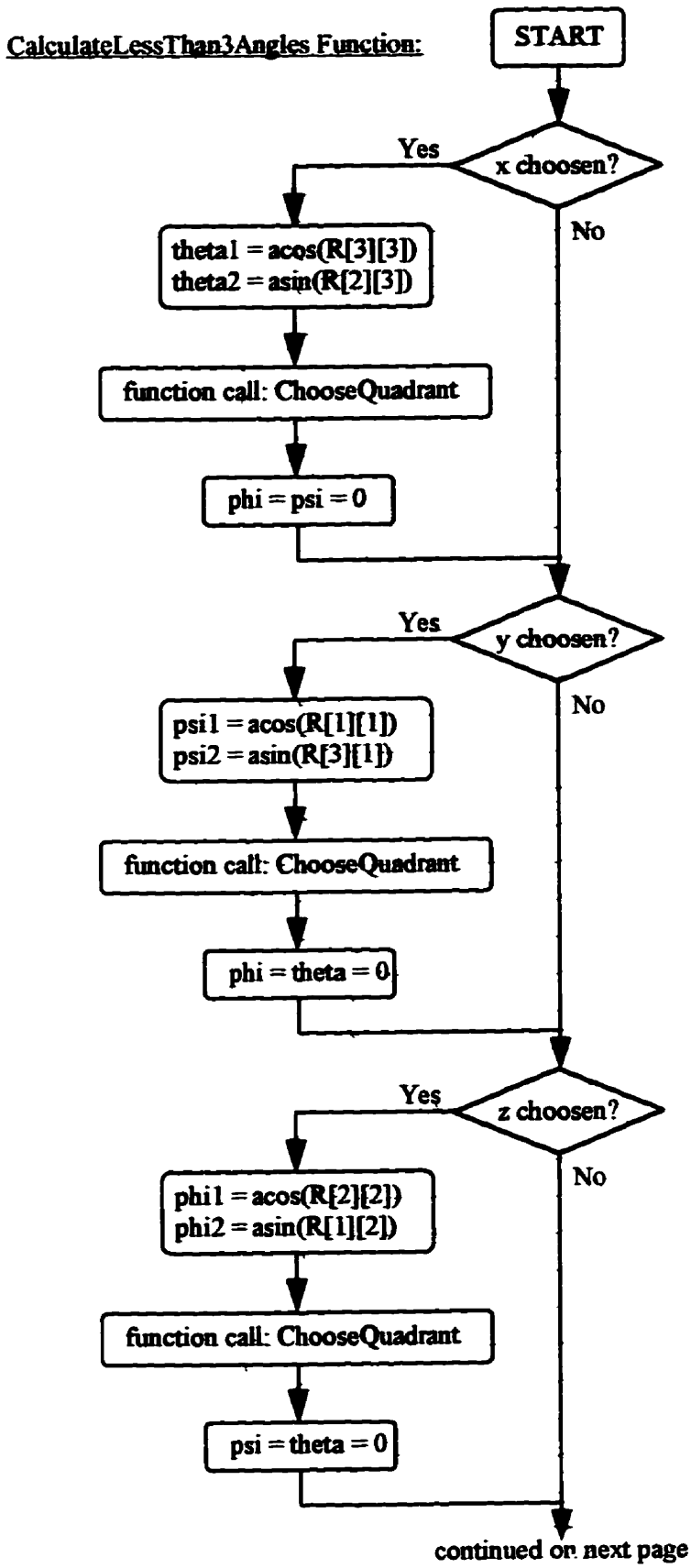


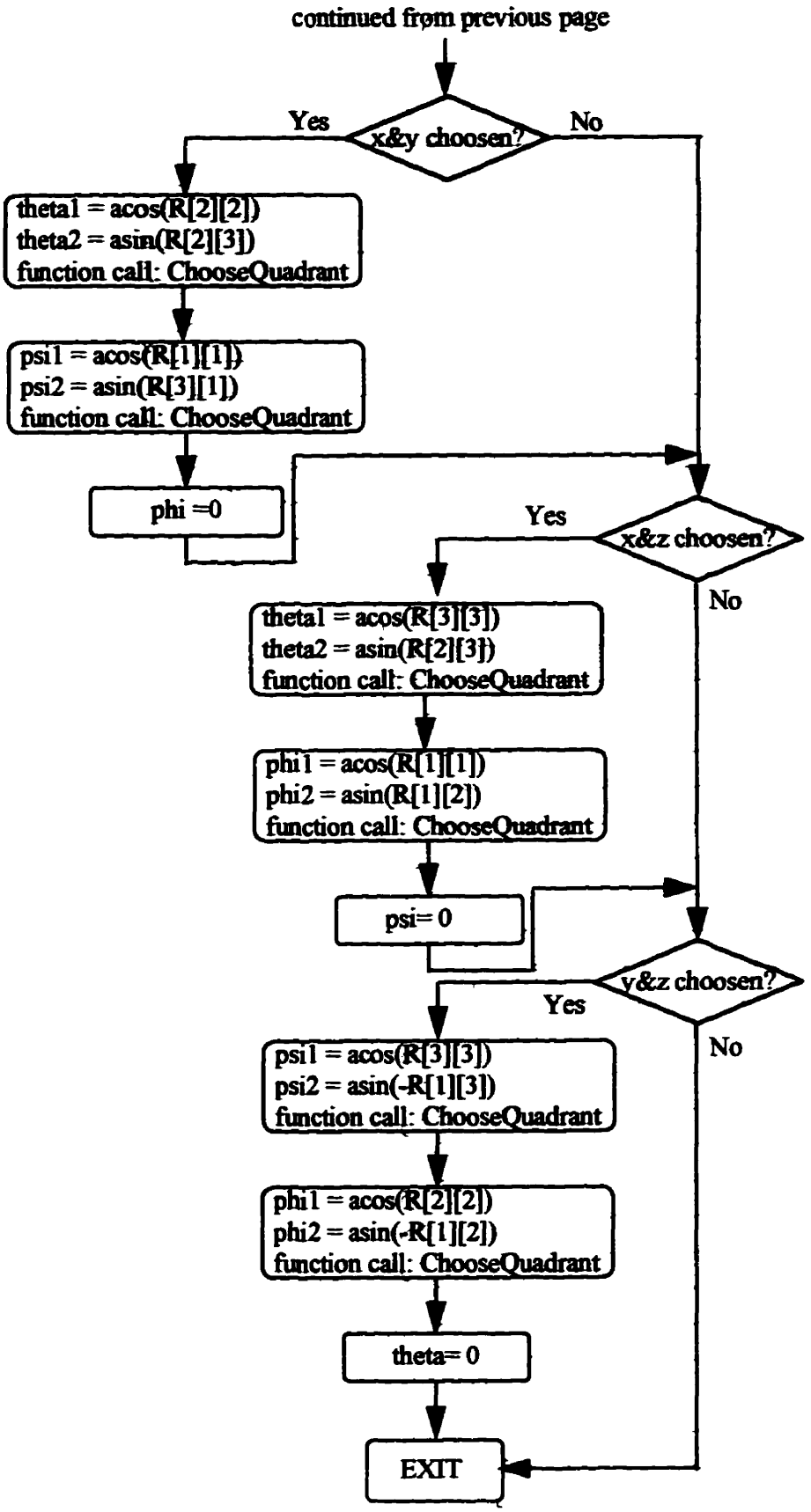
continued on next page


A
continued from previous page
B
phi2=acos(R[2][2]/cos(theta2))
psi2=acos(R[3][3]/cos(theta2))
t21=-cos(theta2) *sin(phi2)
t13=-cos(theta2)*sin(psi2)
abs(t21-R[2][1]) >=
abs(-t21-R[2][1])?
Yes
No
phi2 = -phi2
abs(t13-R[1][3]) >=
abs(-t13-R[2][3])?
Yes
No
psi2 = -psi2
wrongcount1 = abs(theta1-oldtheta)+abs(phi1-oldphi)+abs(psi1-oldphi)
wrongcount2 = abs(theta2-oldtheta)+abs(phi2-oldphi)+abs(psi2-oldphi)
wrongcount1>wrongcount2
Yes
No
phi = phi2
theta = theta2
psi = psi2
phi = phi2
theta = theta2
psi = psi2
EXIT

ChooseQuadrant Function:



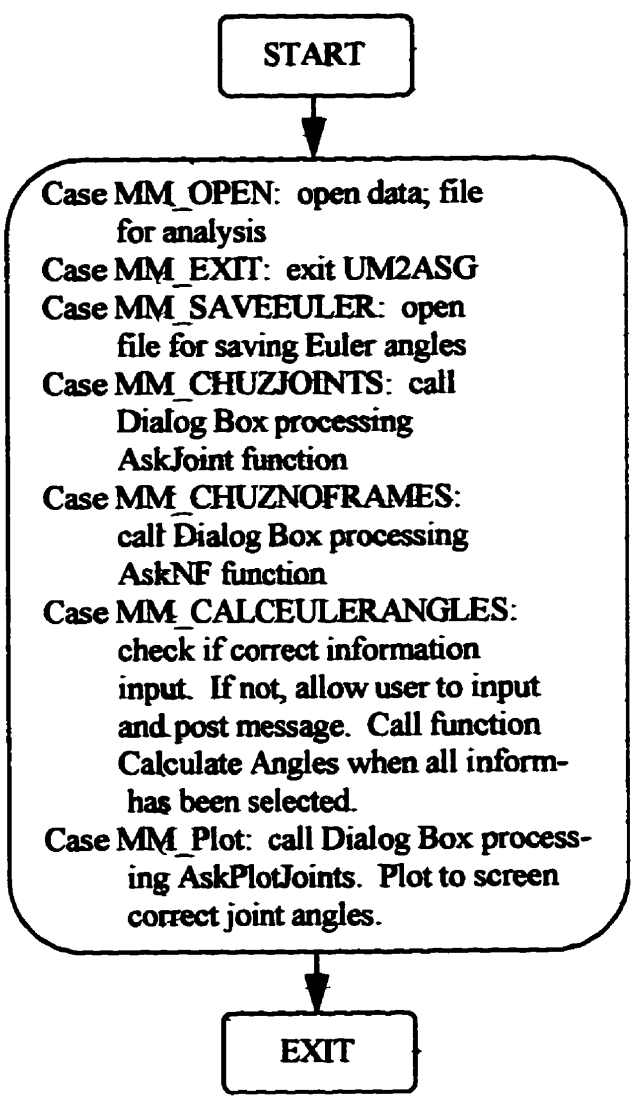

CalculateLessThan3Angles Function:
START
x choosen?
Yes
No
theta1 = acos(R[3][3])
theta2 = asin(R[2][3])
function call: ChooseQuadrant
phi = psi = 0
y choosen?
Yes
No
psi1 = acos(R[1][1])
psi2 = asin(R[3][1])
function call: ChooseQuadrant
phi = theta = 0
z choosen?
Yes
No
phi1 = acos(R[2][2])
phi2 = asin(R[1][2])
function call: ChooseQuadrant
psi = theta = 0
continued on next page


continued from previous page
x&y choosen?
Yes
No
theta1 = acos(R[2][2])
theta2 = asin(R[2][3])
function call: ChooseQuadrant
psi1 = acos(R[1][1])
psi2 = asin(R[3][1])
function call: ChooseQuadrant
phi =0
x&z choosen?
Yes
No
theta1 = acos(R[3][3])
theta2 = asin(R[2][3])
function call: ChooseQuadrant
phi1 = acos(R[1][1])
phi2 = asin(R[1][2])
function call: ChooseQuadrant
psi= 0
y&z choosen?
Yes
No
psi1 = acos(R[3][3])
psi2 = asin(-R[1][3])
function call: ChooseQuadrant
phi1 = acos(R[2][2])
phi2 = asin(-R[1][2])
function call: ChooseQuadrant
theta= 0
EXIT

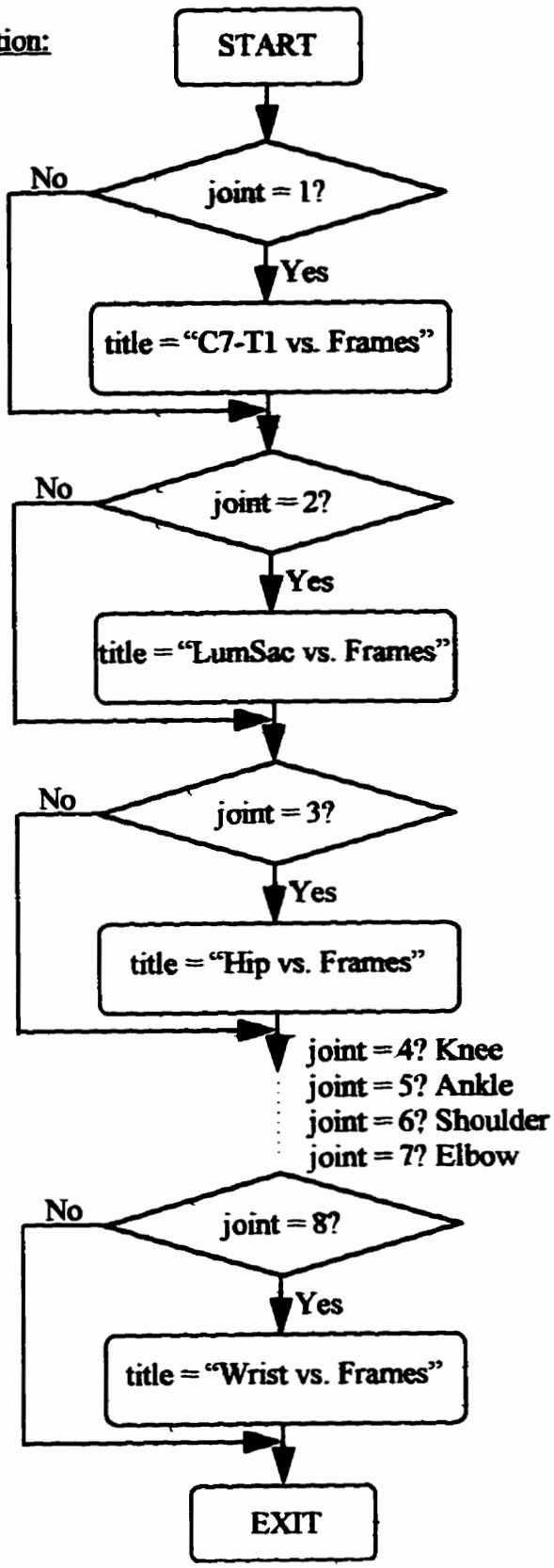
UM2ASG FUNCTIONS

The following flowcharts are for functions found in the file UM2ASG. This file contains the main menu and manages the Windows control of the the software.

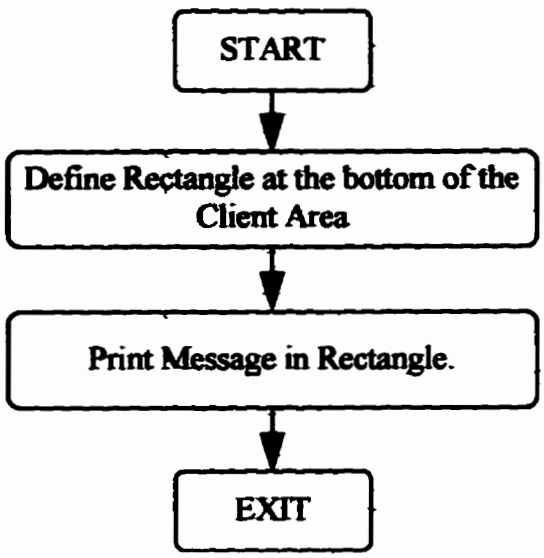
WndProc Function:



GetPlotTitle Function:



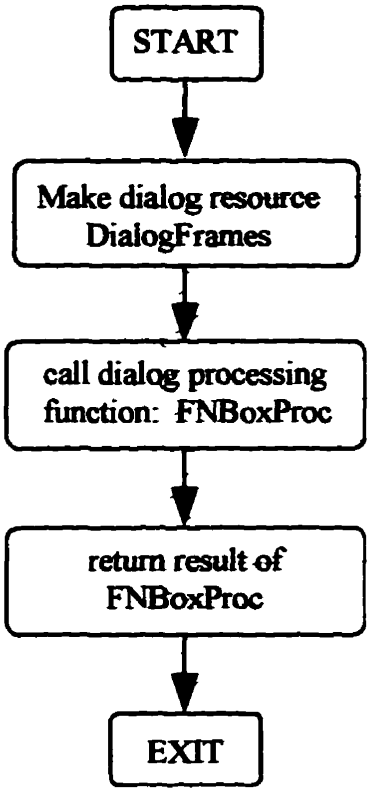
StatusBar Function:



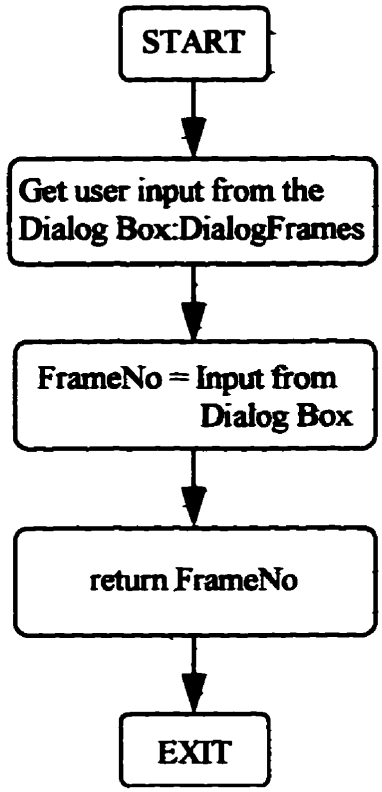
USERINPT FUNCTIONS

The following functions are contained in the USERINPT file. This file contains the functions used to control and manage the resources, such as Dialog Boxes, used by the generalized UM^{2}AS system.

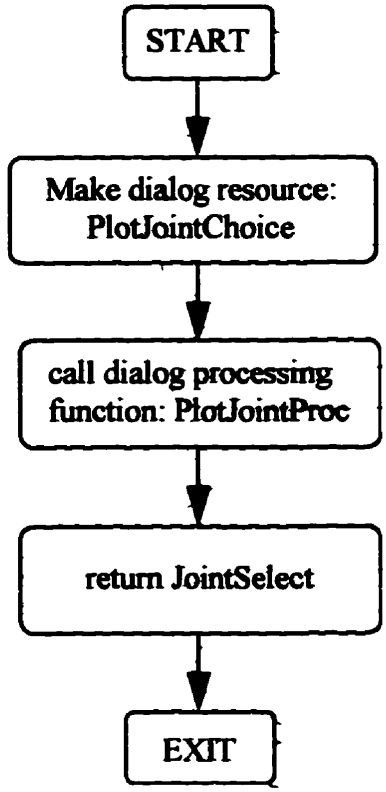
AskNF Function:



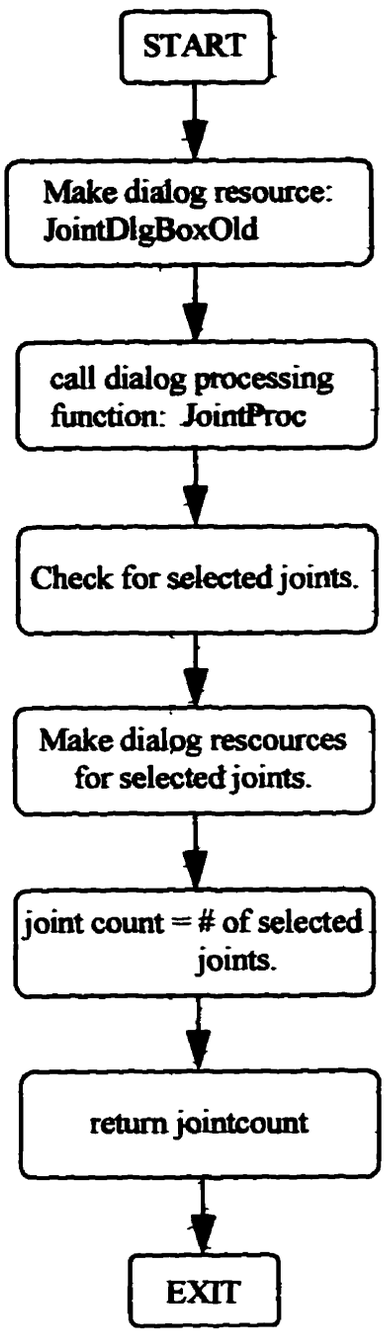
FnBoxProc Function:



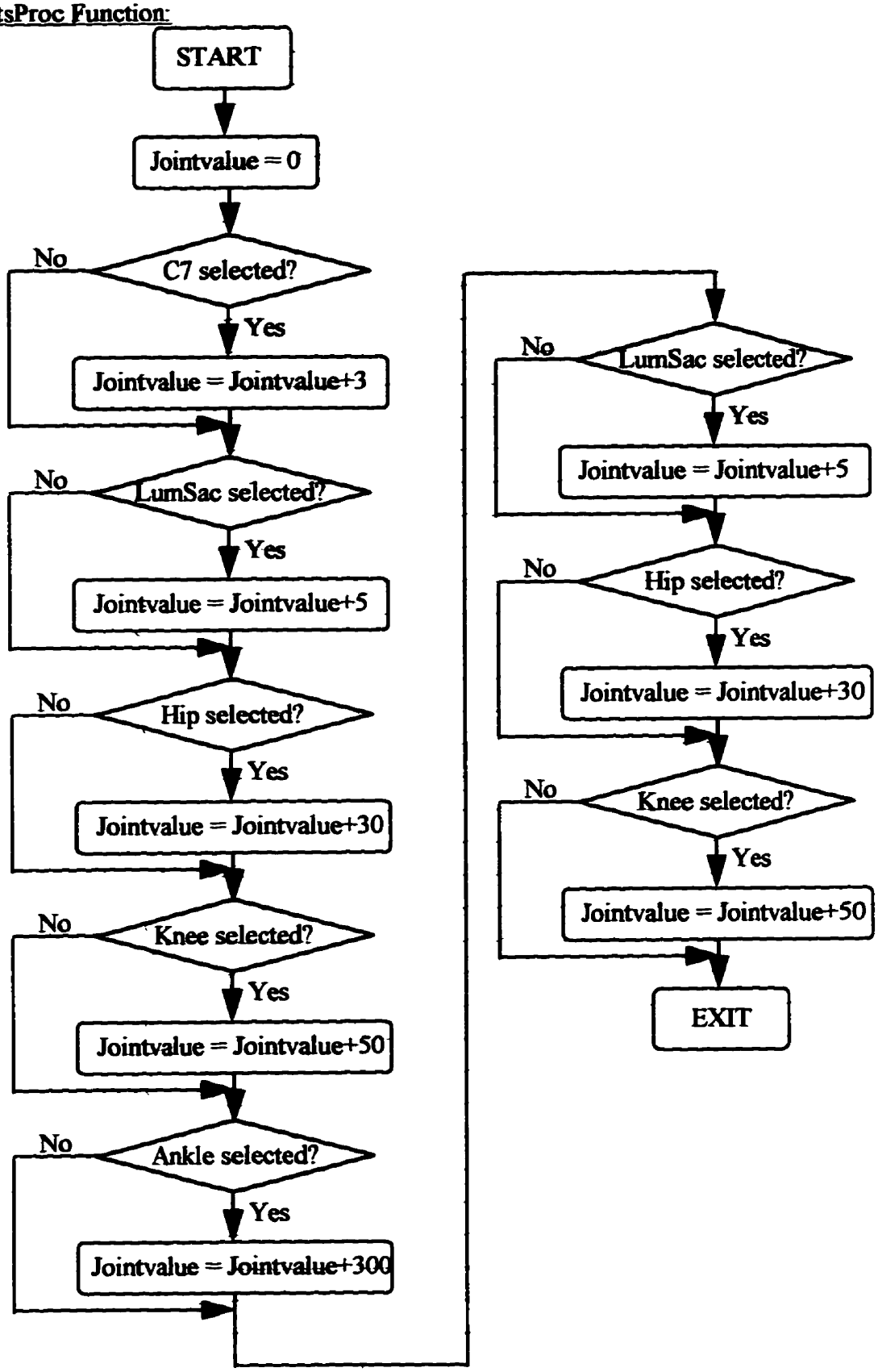
AskPlotJoints Function:



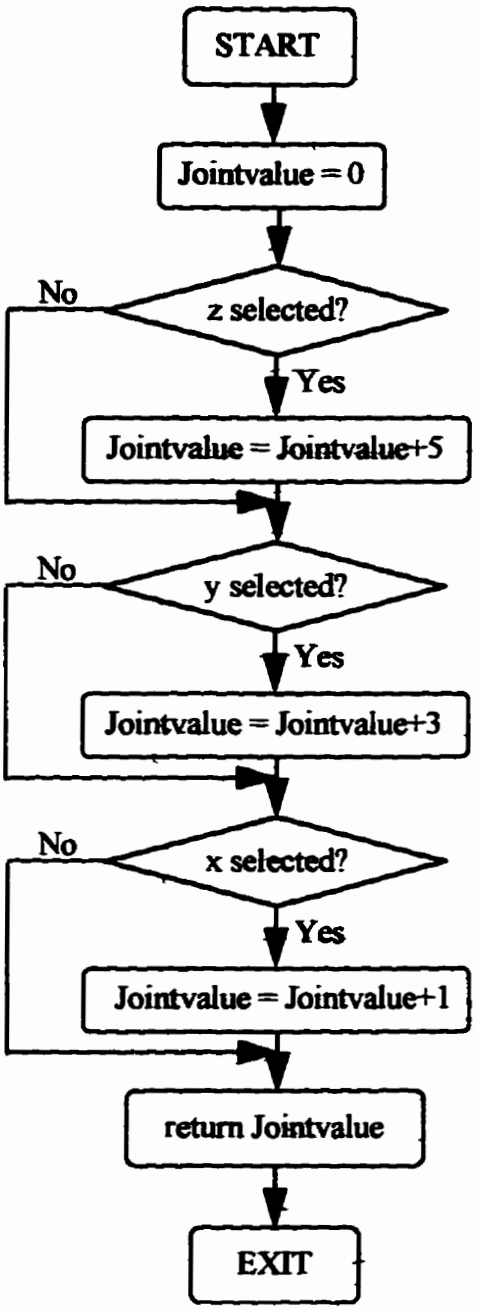
AskJoints Function:



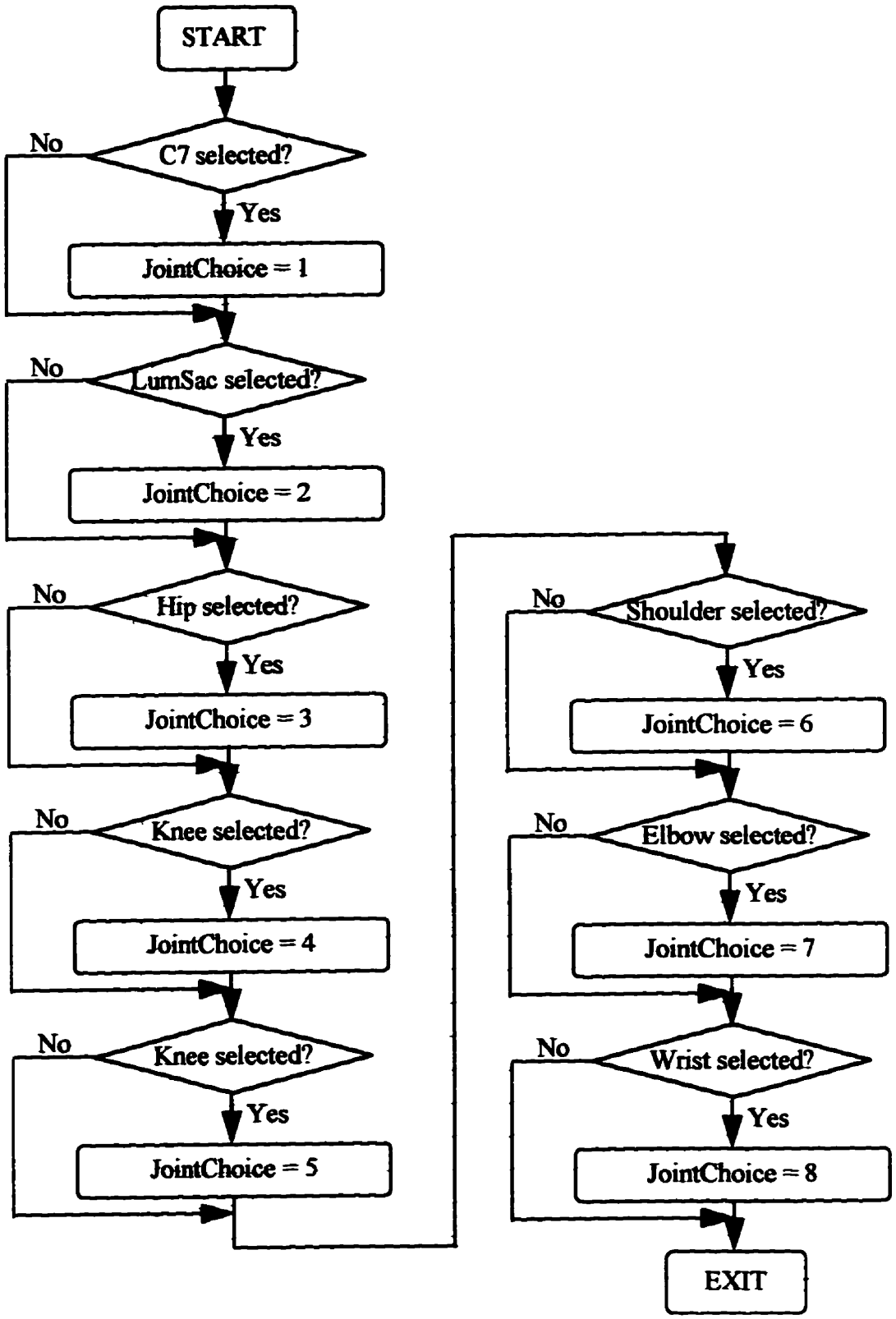
JointsProc Function:



Flowchart for C7Proc, LumSacProc, HipProc, KneeProc, AnkleProc, ShoulderProc, ElbowProc, and WristProc Functions:



PlotJointsProc Function:



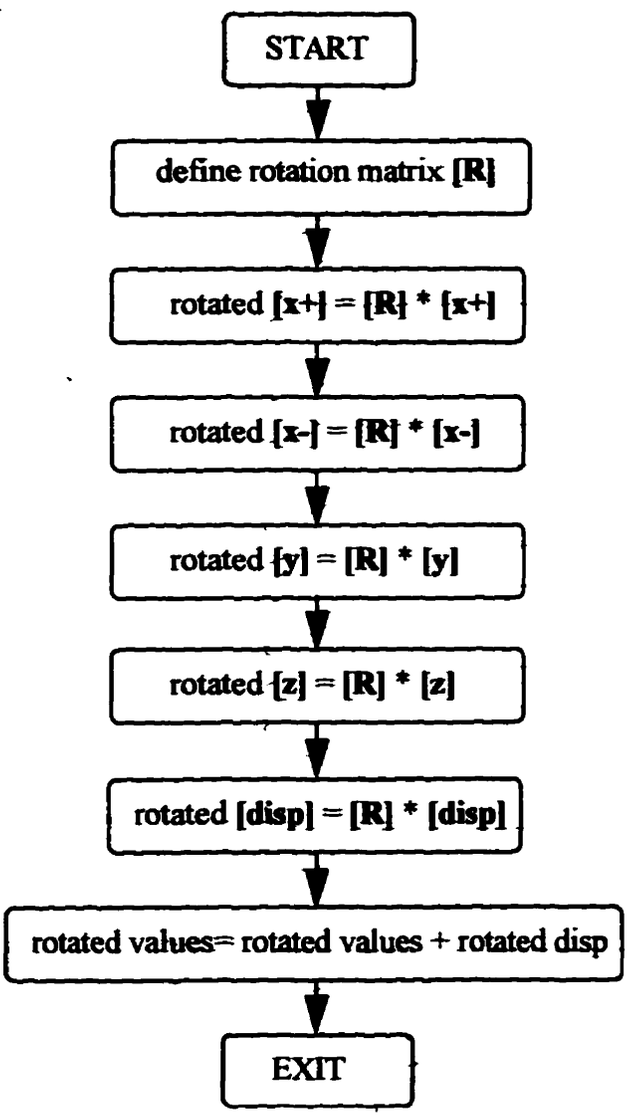
Appendix C

Flowcharts for Gen_Mark Software

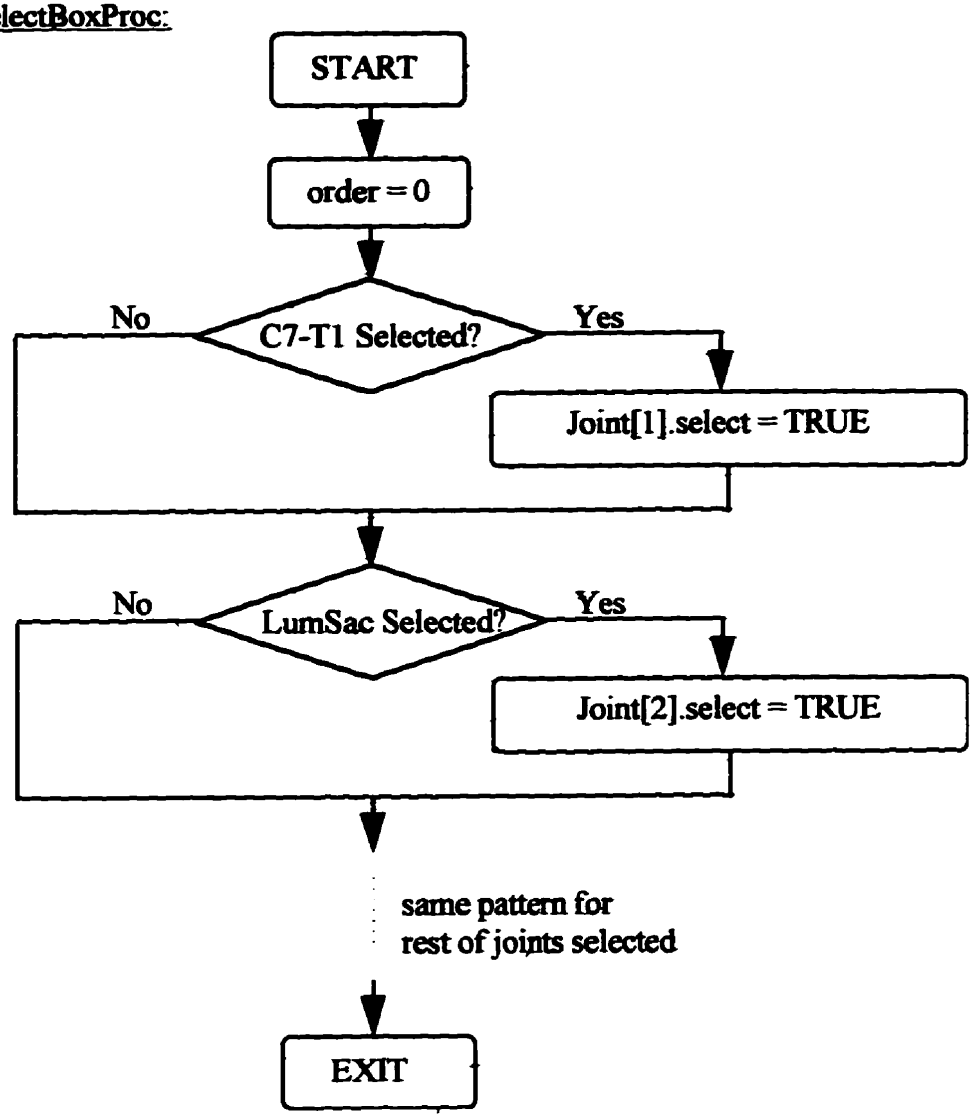
Gen_Mark FUNCTIONS:

The following functions are contained in the Gen_Mark file. This program is used to generate theoretical marker positions.

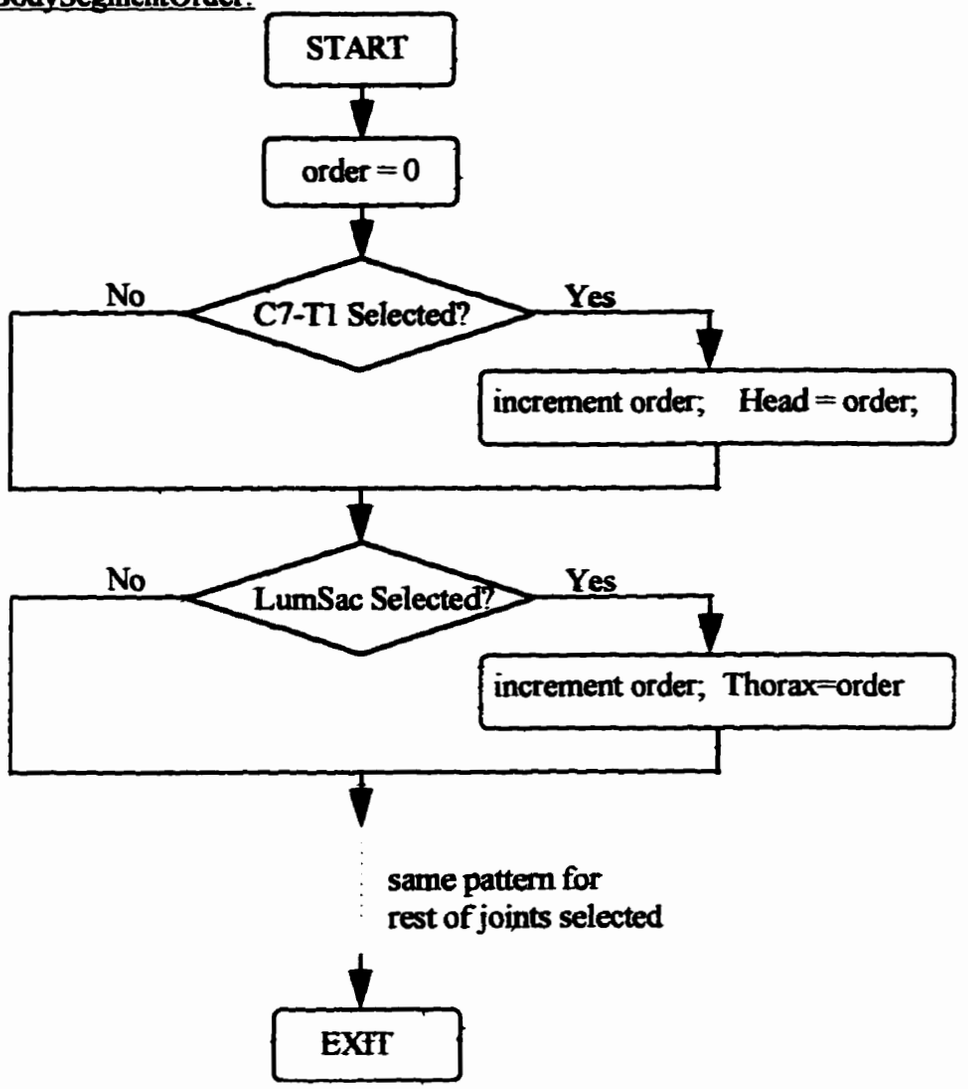
MultiplyByRotation:



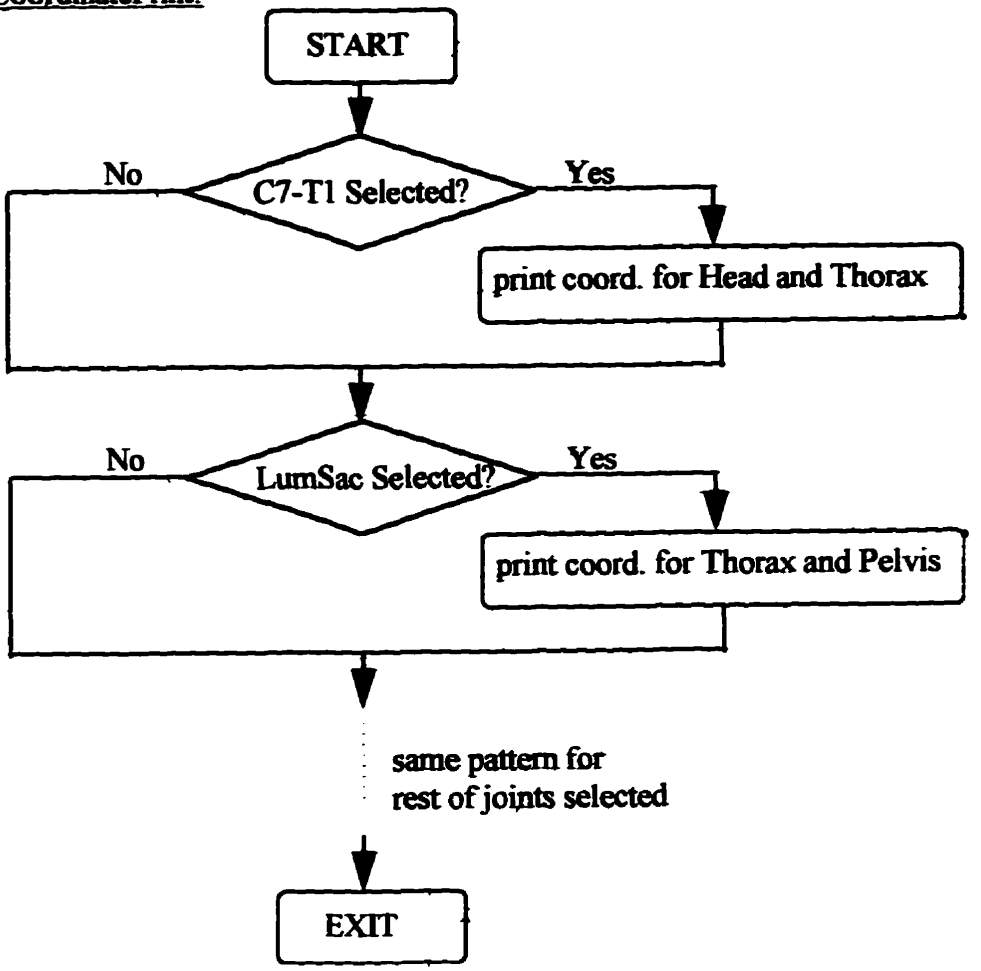
JointSelectBoxProc:



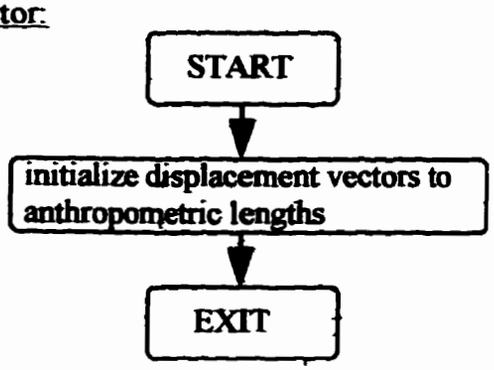
AssignBodySegmentOrder:



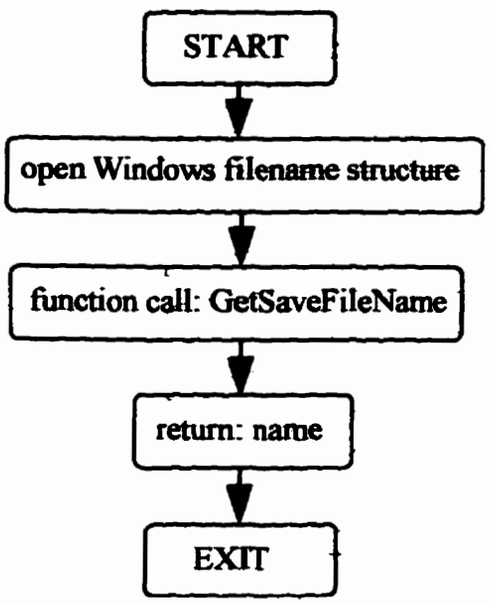
DecideCoordinatePrint:



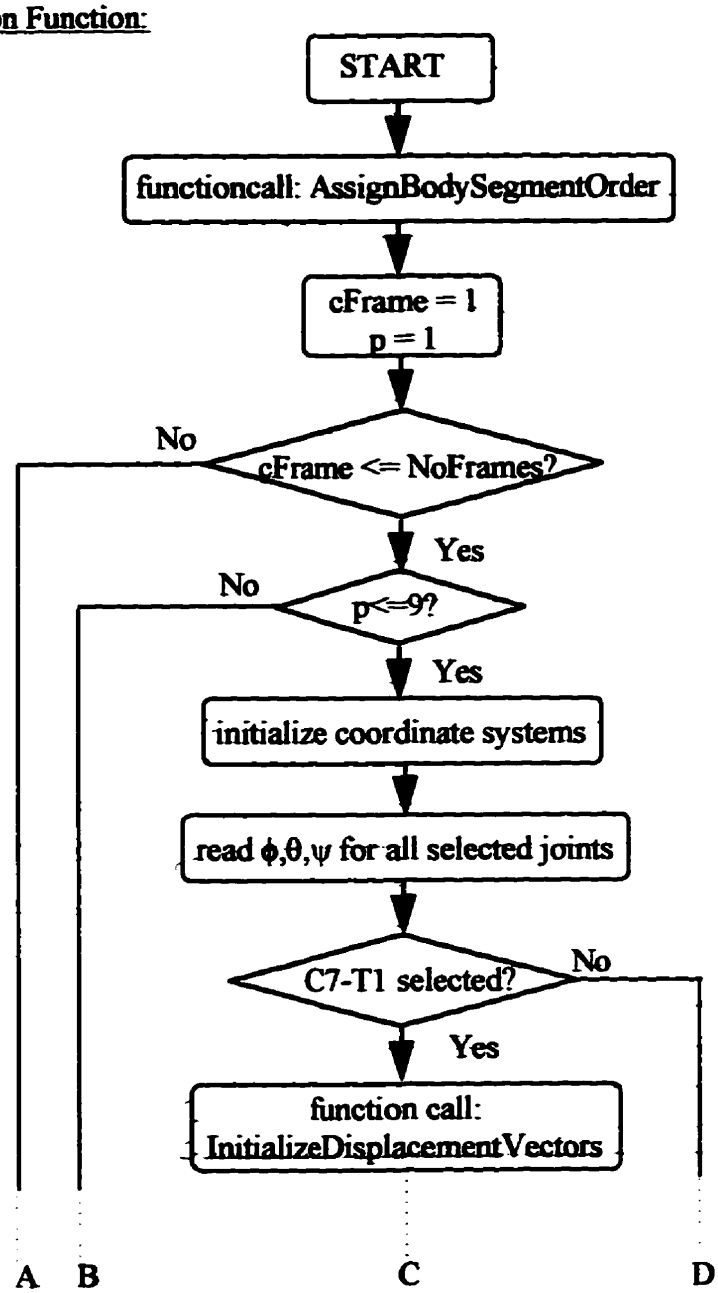
InitializeDisplacementVector:



GetSaveMarkName:

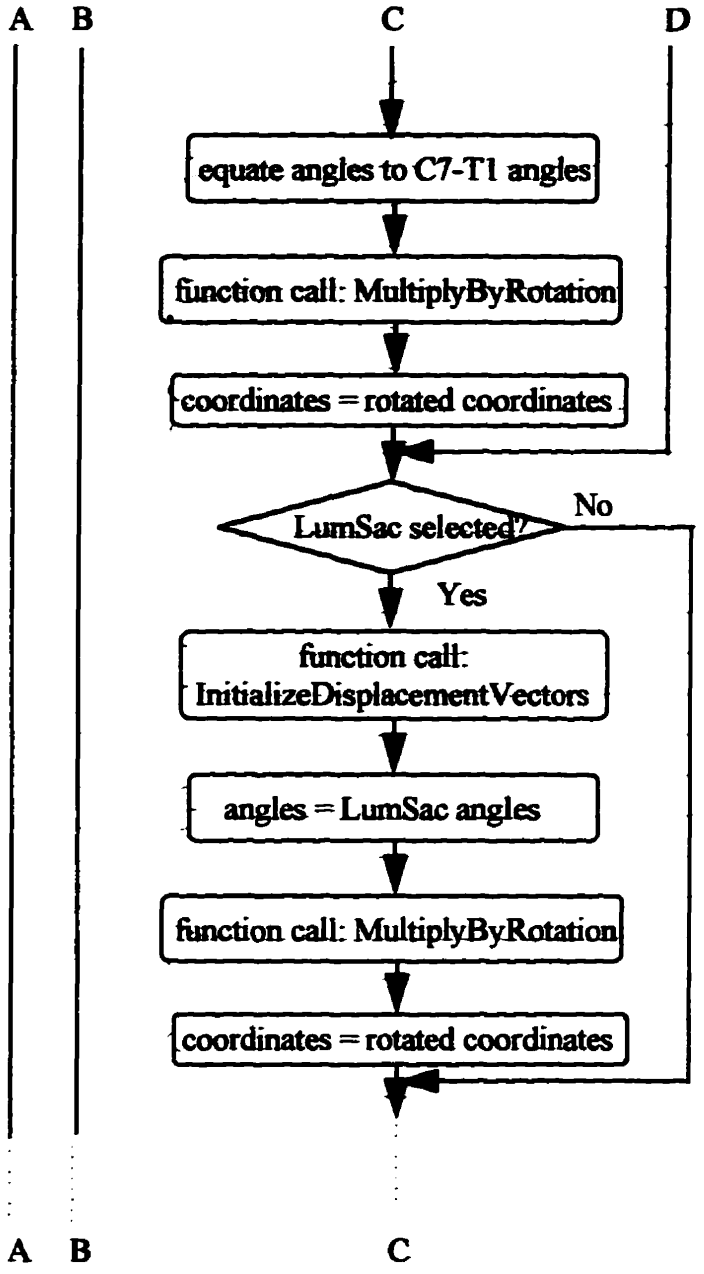


MarkerGeneration Function:

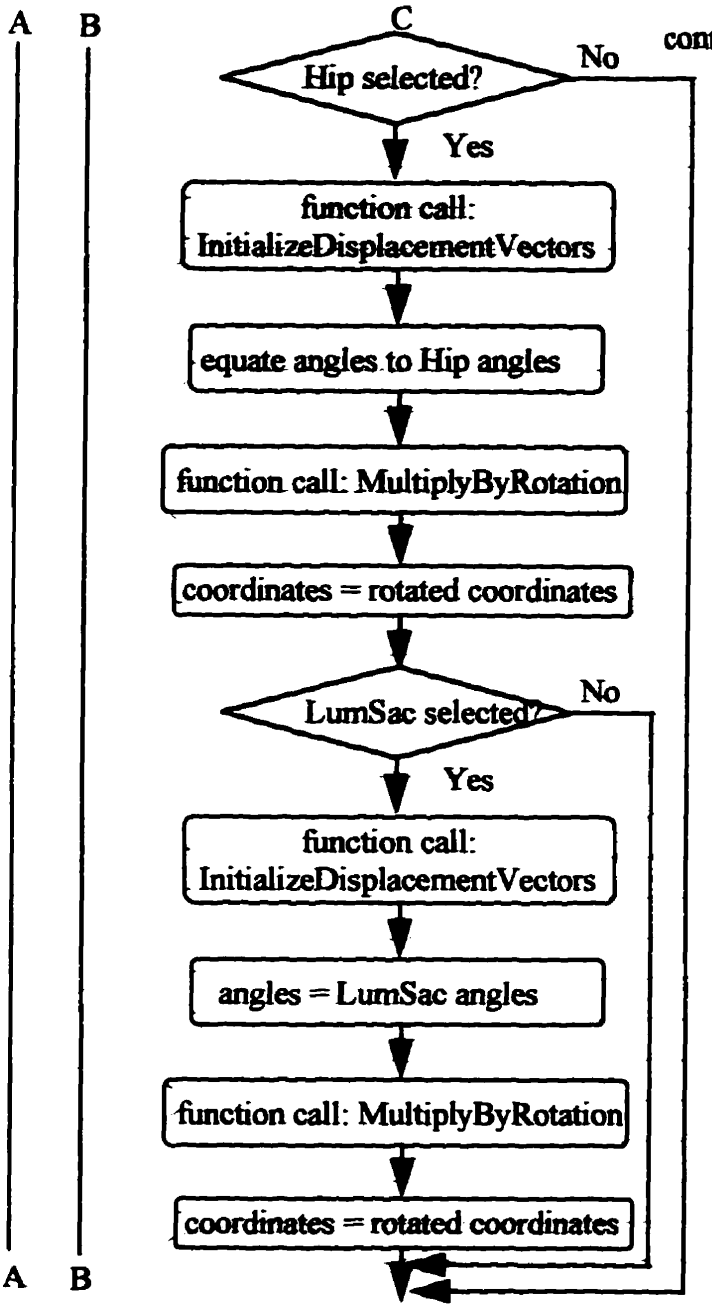


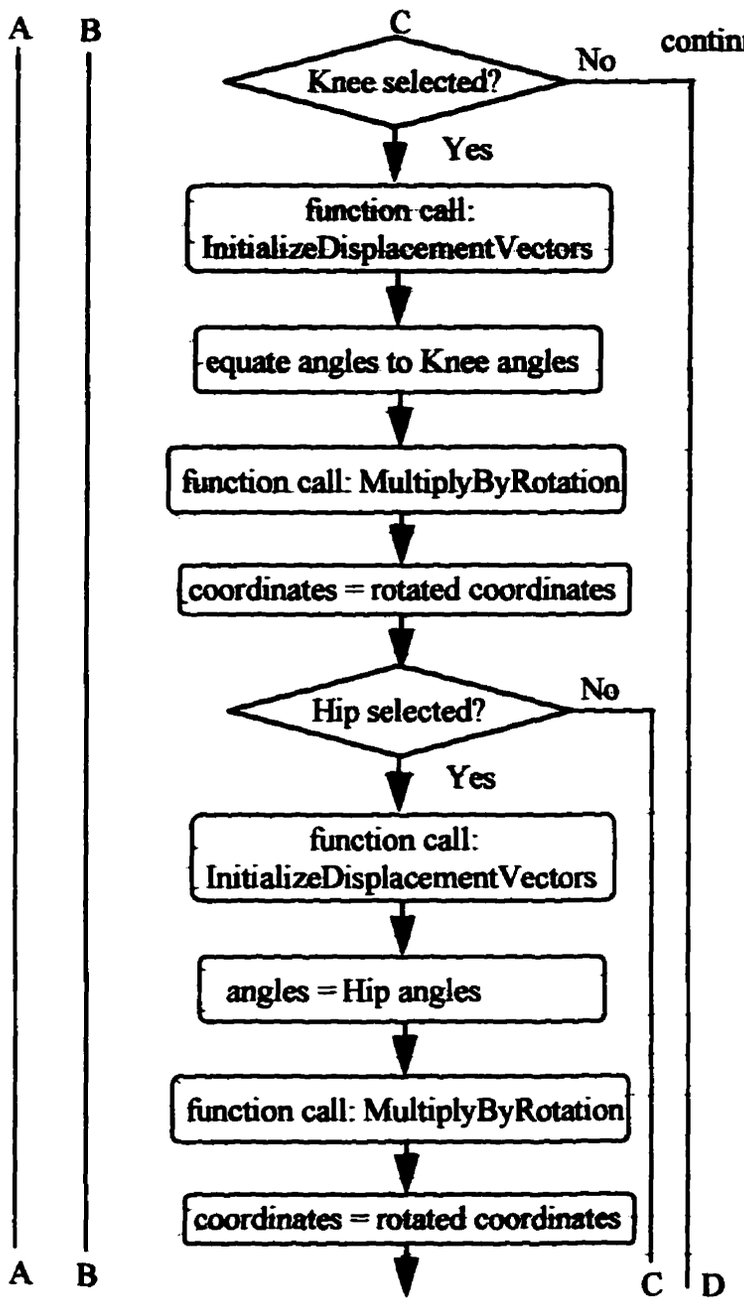
continued on next page

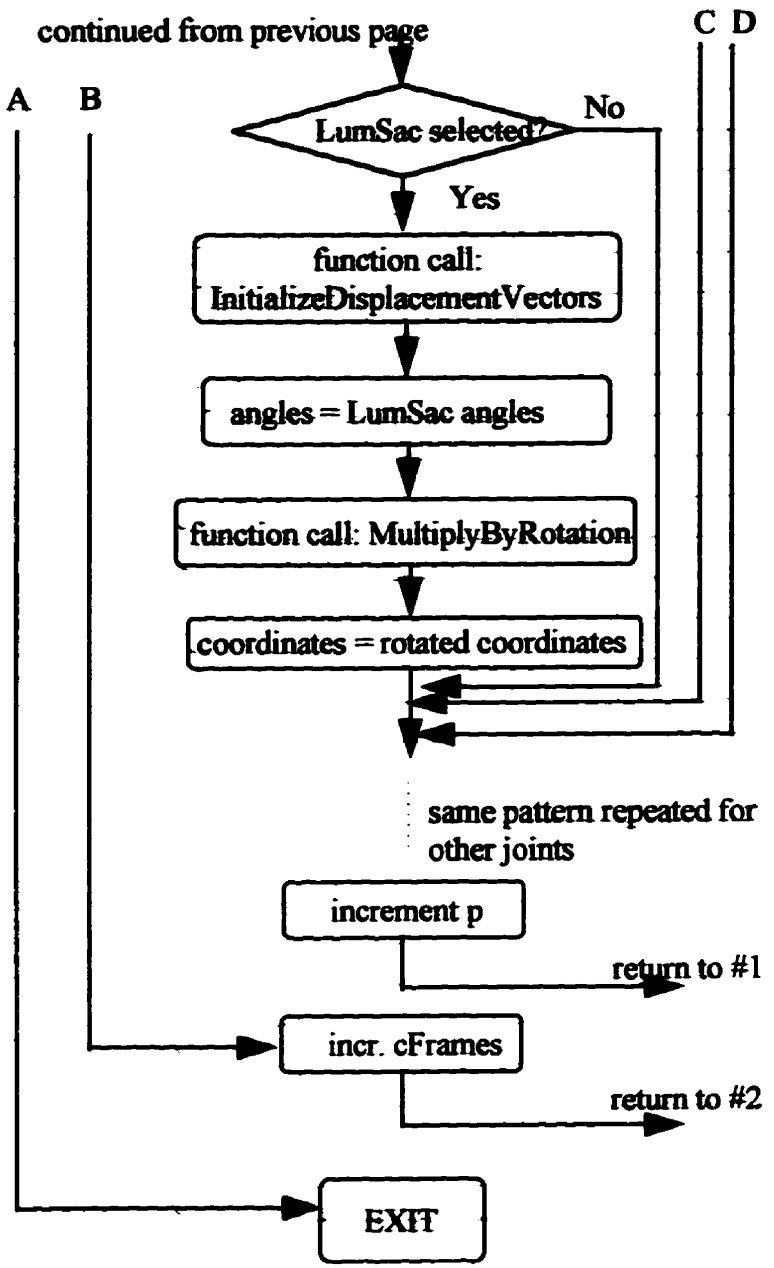
continued from previous page



continued on next page


A
B
C
continued from previous page
Hip selected?
No
Yes
function call:
InitializeDisplacementVectors
equate angles to Hip angles
function call: MultiplyByRotation
coordinates = rotated coordinates
LumSac selected?
No
Yes
function call:
InitializeDisplacementVectors
angles = LumSac angles
function call: MultiplyByRotation
coordinates = rotated coordinates
A
B


A
B
C
continued from previous page
Knee selected?
No
Yes
function call:
InitializeDisplacementVectors
equate angles to Knee angles
function call: MultiplyByRotation
coordinates = rotated coordinates
Hip selected?
No
Yes
function call:
InitializeDisplacementVectors
angles = Hip angles
function call: MultiplyByRotation
coordinates = rotated coordinates
A
B
C
D


continued from previous page
C D
A B
LumSac selected?
No
Yes
function call:
InitializeDisplacementVectors
angles = LumSac angles
function call: MultiplyByRotation
coordinates = rotated coordinates
same pattern repeated for
other joints
increment p
return to #1
incr. cFrames
return to #2
EXIT

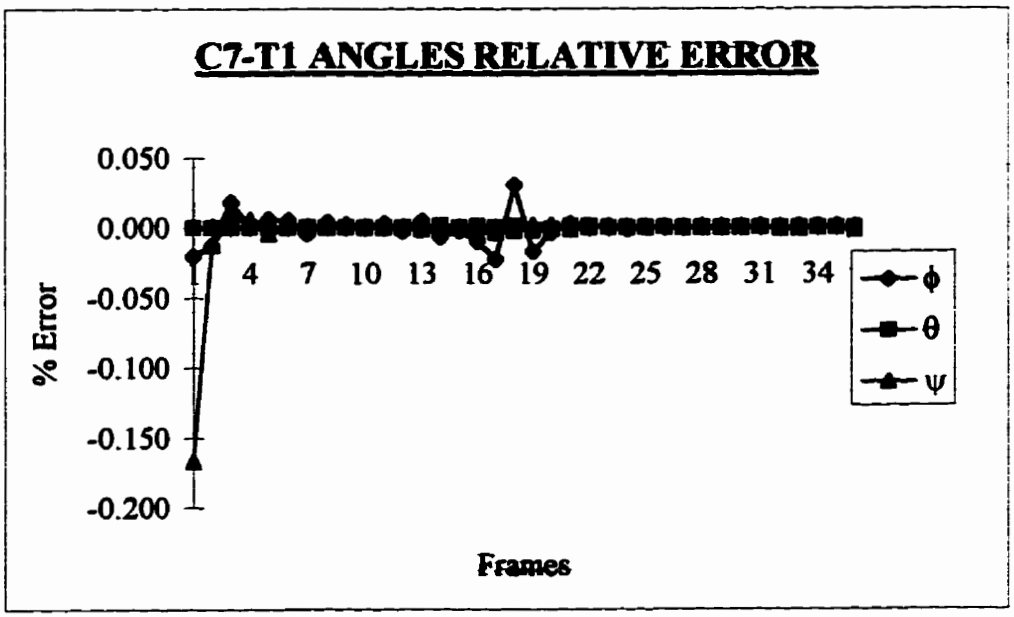
Appendix D

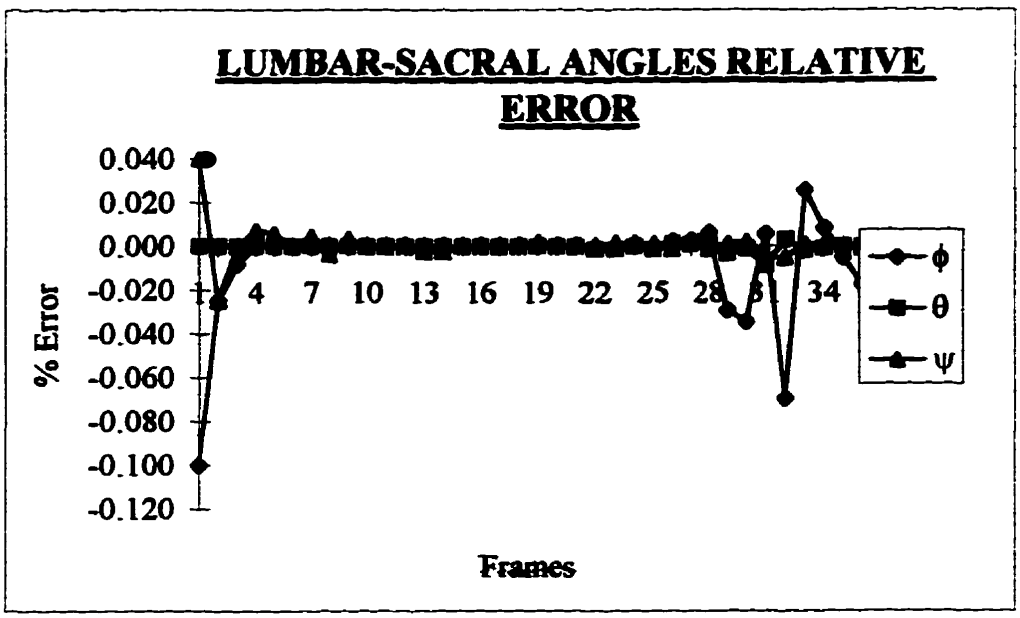
Relative Error Plots

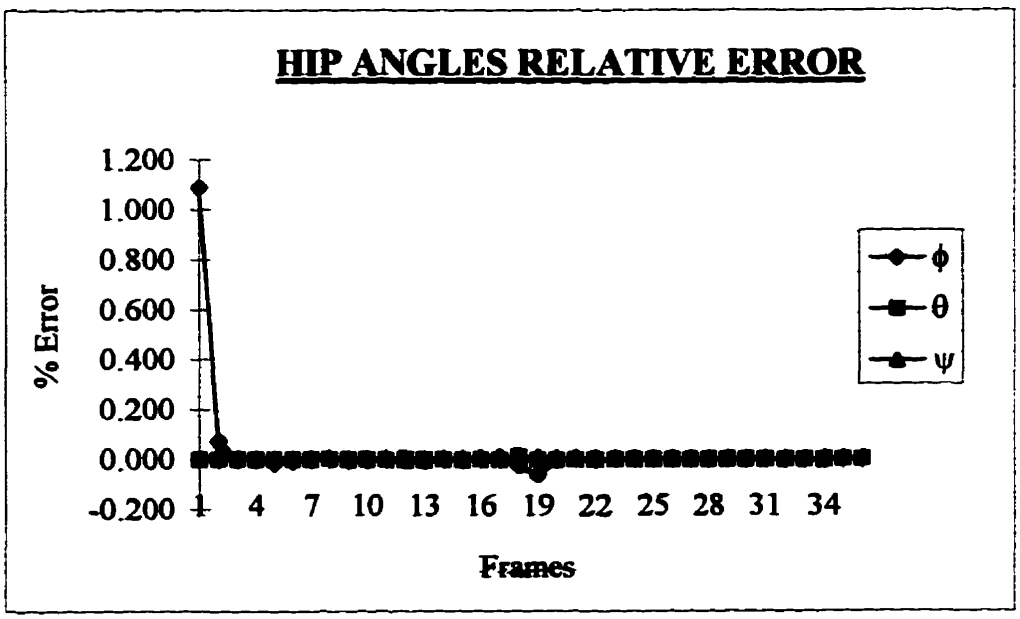
Table D.1: Range of Test Angles

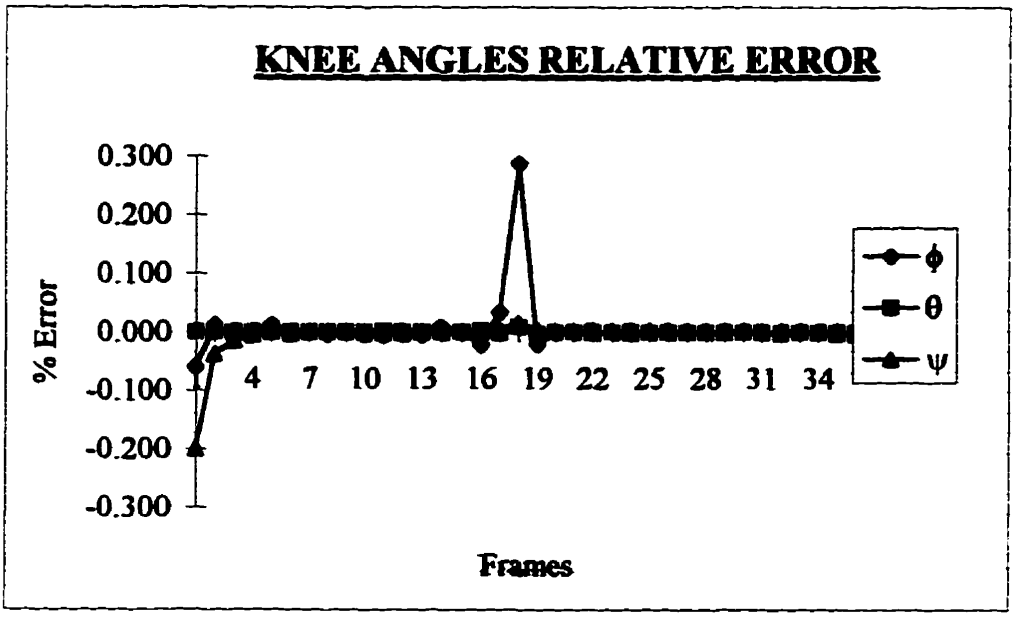
Joint			
	-5 to -110	-7 to -182	3 to 178
	3 to 178	1 to 106	-5 to -110
	1 to 106	3 to 178	-7 to -182
	-5 to -110	-7 to -182	3 to 178
	-7 to -110	-5 to 106	1 to 178
	3 to 106	1 to -110	-5 to 106
	1 to 178	3 to -178	-7 to -182
	3 to 178	1 to 106	1 to 106

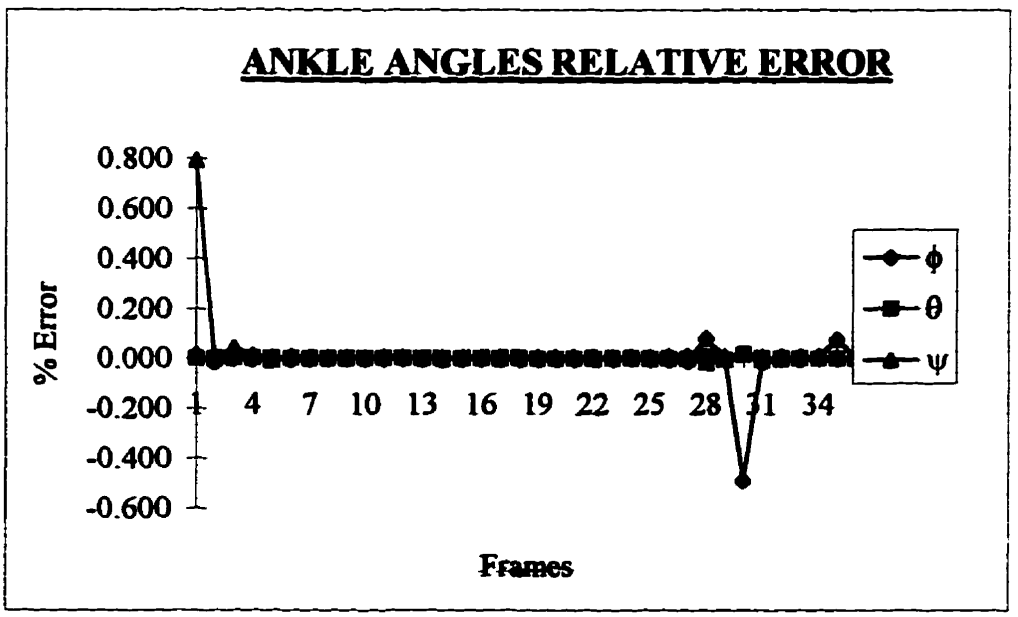
The following plots are a compilation of the %error plots resulting from the testing proceedure described in Chapter Four.

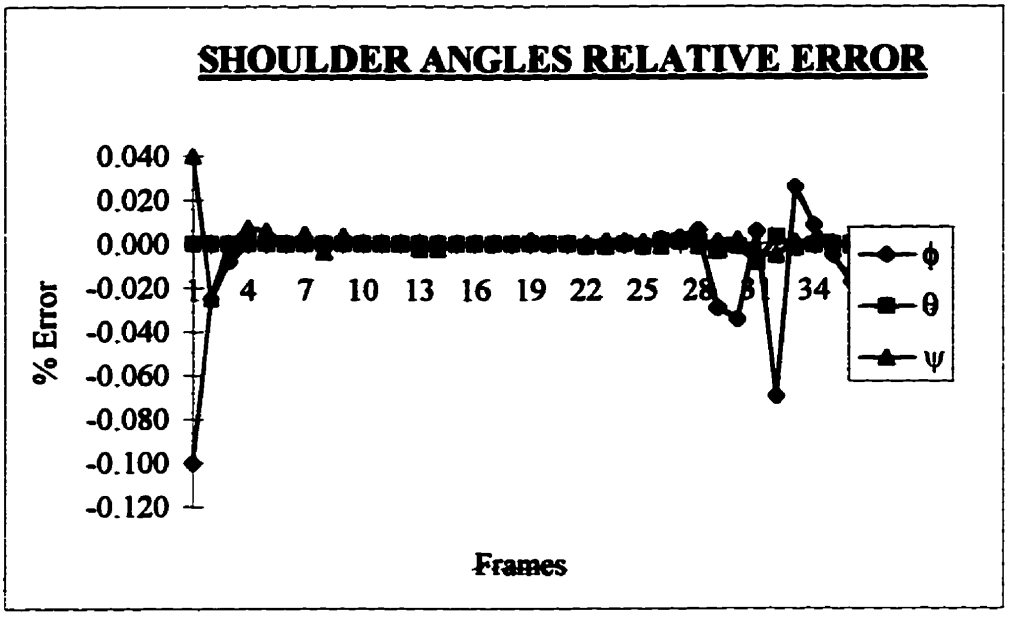


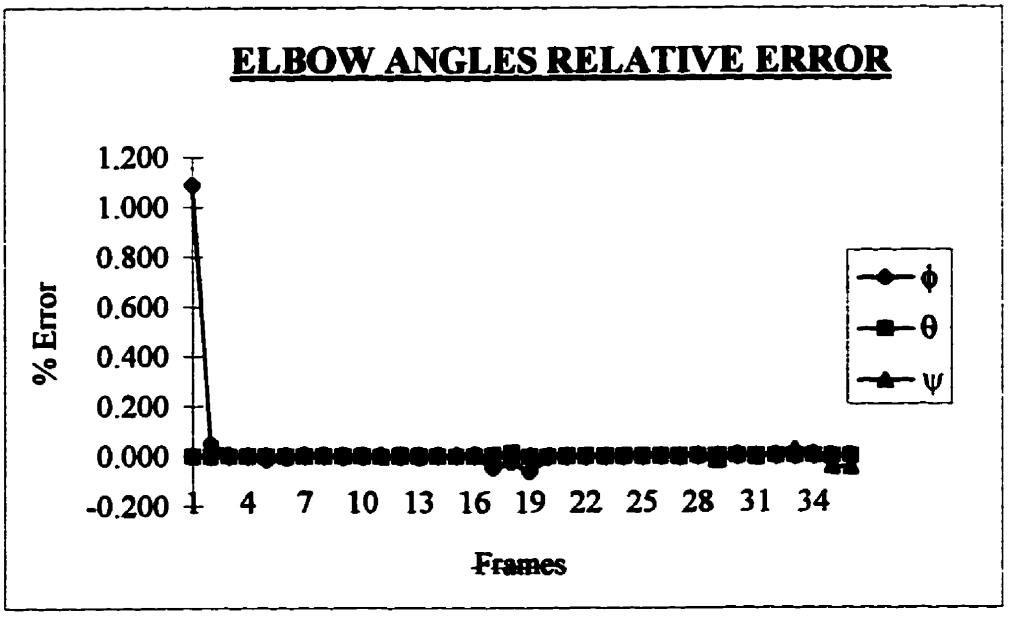

LUMBAR-SACRAL ANGLES RELATIVE ERROR
0.040
0.020
0.000
-0.020
-0.040
-0.060
-0.080
-0.100
-0.120
% Error
1
4
7
10
13
16
19
22
25
28
34
Frames
ϕ
θ
ψ

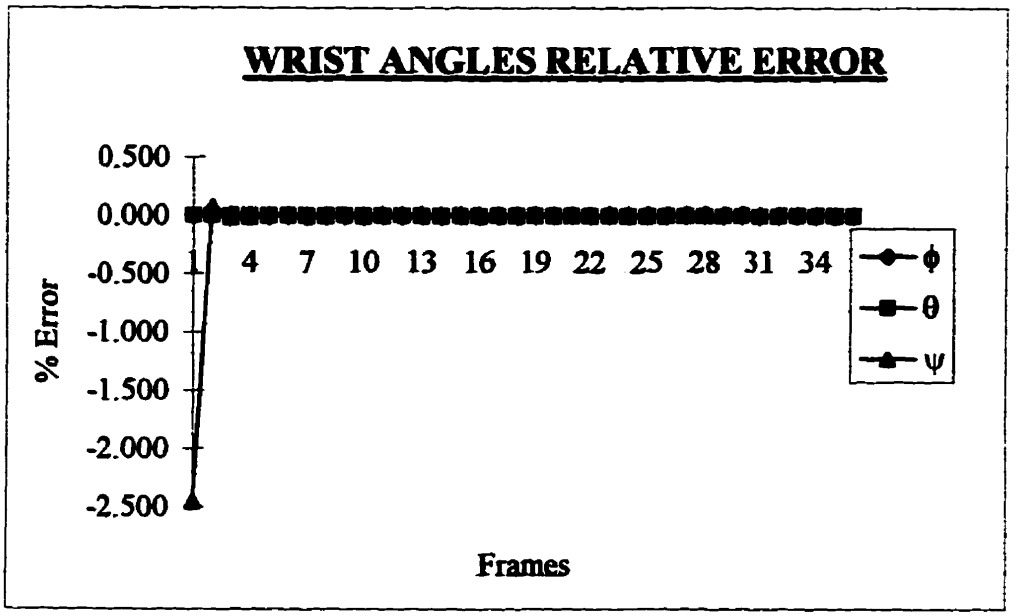

HIP ANGLES RELATIVE ERROR
1.200
1.000
0.800
0.600
0.400
0.200
0.000
-0.200
% Error
1
4
7
10
13
16
19
22
25
28
31
34
Frames
ϕ
θ
ψ


KNEE ANGLES RELATIVE ERROR
0.300
0.200
0.100
0.000
-0.100
-0.200
-0.300
% Error
4 7 10 13 16 19 22 25 28 31 34
Frames
ϕ
θ
ψ


ANKLE ANGLES RELATIVE ERROR
0.800
0.600
0.400
0.200
0.000
-0.200
-0.400
-0.600
% Error
4 7 10 13 16 19 22 25 28 31 34
Frames
ϕ
θ
ψ


SHOULDER ANGLES RELATIVE ERROR
0.040
0.020
0.000
-0.020
-0.040
-0.060
-0.080
-0.100
-0.120
% Error
1
4
7
10
13
16
19
22
25
28
31
34
Frames
ϕ
θ
ψ


ELBOW ANGLES RELATIVE ERROR
1.200
1.000
0.800
0.600
0.400
0.200
0.000
-0.200
% Error
1
4
7
10
13
16
19
22
25
28
31
34
Frames
ϕ
θ
ψ


WRIST ANGLES RELATIVE ERROR
0.500
0.000
-0.500
-1.000
-1.500
-2.000
-2.500
% Error
1
4
7
10
13
16
19
22
25
28
31
34
Frames
ϕ
θ
ψ