

DEVELOPMENT OF A 3-CAMERA VISION SYSTEM AND THE SADDLE  
MOTION ANALYSIS OF HORSES VIA THIS SYSTEM

A THESIS SUBMITTED TO  
THE GRADUATE SCHOOL OF NATUREL AND APPLIED SCIENCES  
OF  
MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÖZDE DOĞAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR  
THE DEGREE OF MASTER OF SCIENCE  
IN  
MECHANICAL ENGINEERING

SEPTEMBER 2009

Approval of the thesis:

**DEVELOPMENT OF A 3-CAMERA VISION SYSTEM AND THE SADDLE  
MOTION ANALYSIS OF HORSES VIA THIS SYSTEM**

submitted by **GÖZDE DOĞAN** in partial fulfillment of the requirements for the degree of **Master of Science in Mechanical Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen  
Dean, Graduate School of **Natural and Applied Sciences** \_\_\_\_\_

Prof. Dr. Suha Oral  
Head of Department, **Mechanical Engineering** \_\_\_\_\_

Prof. Dr. Reşit Soylu  
Supervisor, **Mechanical Engineering Dept., METU** \_\_\_\_\_

**Examining Committee Members**

Prof. Dr. Tuna Balkan  
Mechanical Engineering Dept., METU \_\_\_\_\_

Prof. Dr. Reşit Soylu  
Mechanical Engineering Dept., METU \_\_\_\_\_

Prof. Dr. Sıtkı Kemal İder  
Mechanical Engineering Dept., METU \_\_\_\_\_

Asst.Prof. Dr. İlhan Konukseven  
Mechanical Engineering Dept., METU \_\_\_\_\_

Asst. Prof. Dr. Kutluk Bilge Arıkan  
Mechatronics Engineering Dept., ATILIM UNIVERSITY \_\_\_\_\_

**Date** : 02.09.2009

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name, Last name : Güzde DOĐAN

Signature :

## **ABSTRACT**

### **DEVELOPMENT OF A 3-CAMERA VISION SYSTEM AND THE SADDLE MOTION ANALYSIS OF HORSES VIA THIS SYSTEM**

Dođan, Gzde

M.Sc., Department of Mechanical Engineering

Supervisor : Prof. Dr. Reřit Soylu

September 2009, 175 pages

One of the purposes of this study is to develop a vision system consisting of 3 inexpensive, commercial cameras. The system is intended to be used for tracking the motion of objects in a large calibration volume, typically 6.5 m. wide and 0.7 m. high. Hence, a mechanism is designed and constructed for the calibration of the cameras.

The second purpose of the study is to develop an algorithm, which can be used to obtain the kinematic data associated with a rigid body, using a vision system. Special filters are implemented in the algorithm to identify the 3 markers attached on the body. Optimal curves are fitted to the position data of the markers after smoothing the data appropriately. The outputs of the algorithm are the position, velocity and acceleration of any point (visible or invisible) on the body and the angular velocity and acceleration of the body. The singularities associated with the algorithm are also determined.

Using the vision setup and the developed algorithm for tracking the kinematics of a rigid body, the motions of the saddles of different horses are investigated for different gaits. Similarities and differences between horses and/or gaits are analyzed to lead to quantitative results. Using the limits induced by the whole body vibration of humans, for the first time in the world, daily, allowable riding time and riding distances are determined for different horses and gaits. Furthermore, novel, quantitative horse comfort indicators are proposed. Via the experiments performed, these indicators are shown to be consistent with the comfort assessment of experienced riders.

Finally, in order to implement the algorithms proposed in this study, a computer code is developed using MATLAB<sup>®</sup>.

Keywords: Vision System, Saddle Motion Analysis of Horses, Motion Tracking, Camera Calibration, Whole Body Vibration, Equine Motion Analysis, Allowable Riding Time, Allowable Riding Distance, Horse Comfort Indicators.

## ÖZ

### ÜÇ KAMERALI BİR GÖRÜNTÜ SİSTEMİNİN GELİŞTİRİLMESİ VE GELİŞTİRİLEN BU SİSTEM ARACILIĞIYLA ATLARDA EYER HAREKET ANALİZİ

Doğan, Gözde

Yüksek Lisans, Makina Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Reşit Soylu

Eylül 2009, 175 sayfa

Bu çalışmanın amaçlarından bir tanesi, pahalı olmayan üç ticari kameradan oluşan bir görüntü sisteminin geliştirilmesidir. Sistemin, büyük bir kalibrasyon hacmi (tipik olarak 6.5 m. en ve 0.7 m. yükseklik) içerisindeki nesnelerin hareket takibinde kullanılması amaçlanmıştır. Bu nedenle, kamera kalibrasyonu için bir mekanizma tasarlanmış ve üretilmiştir.

Çalışmanın ikinci amacı, bir görüntü sistemi kullanarak katı bir cisme ait kinematik verilere ulaşmak üzere kullanılacak bir algoritmanın geliştirilmesidir. Bu algorithmanda cismin üzerine yapıştırılmış 3 adet işaretleyicinin belirlenmesi amacıyla özel filtreler oluşturulmuştur. İşaretleyicilerin düzeltilen pozisyon verilerine optimum eğriler oturtulmuştur. Algoritmanın çıktıları cisim üzerindeki herhangi (görünür veya görünmez) bir noktanın pozisyon, hız ve ivmesi ile cismin açısal hızı ve ivmesidir. Algoritmanın tekil konumları da belirlenmiştir.

Görüntü sistemini ve katı bir cismin kinematik verilerini elde etmek üzere geliştirilmiş algoritmayı kullanarak, eyerlerin farklı atlara ait farklı yürüyüşlerdeki

hareketleri incelenmiştir. Atlar ve/veya yürüyüşleri arasındaki benzerlik ve farklılıklar incelenerek nicel sonuçlara ulaşılmıştır. İnsanların titreşimine maruz kalma sınırları kullanılarak, dünyada ilk defa, farklı atlar ve yürüyüşler için günlük izin verilen sürüş süre ve uzaklıkları hesaplanmıştır. Ayrıca, yeni, nicel at konfor göstergeleri önerilmiştir. Yapılan deneyler aracılığıyla, bu göstergelerin deneyimli binicilerin konfor algılamaları ile tutarlı olduğu gösterilmiştir.

Son olarak, MATLAB® kullanılarak, bu çalışmada önerilen algoritmaları gerçekleştirmek üzere, bir bilgisayar kodu geliştirilmiştir.

Anahtar Kelimeler: Görüntü Sistemi, Atlarda Eyer Hareket Analizi, Hareket Takibi, Kamera Kalibrasyonu, Tüm Vücut Titreşimi, Atlarda Hareket Analizi, Azami Sürüş Süresi, Azami Sürüş Uzaklığı, At Konfor Göstergeleri.

To My Family

## ACKNOWLEDGEMENTS

Firstly, I would like to thank my supervisor *Prof. Dr. Reşit Soylu* for giving me the opportunity of doing this research and his endless support with encouraging approach throughout the steps in formation of this thesis.

Surely, I am grateful to my mother *Feriha*, my father *Günay Doğan* and also my brother *Çağlar* for their invaluable support and endless love. Also my special thanks go to *Çağatay Tanıl* not only for encouraging me but also for his precious support and helpful suggestions.

Additionally, I would like to thank my friends; *Özge Arslan, A. Emre Öngüt, Gökhan Kiper and M. Emre Akçay* for their friendship and support. And I would like to thank my friend *Hatice Sinem Şaş* for her invaluable support by taking part in my initial experiments as a volunteer. Also all my colleagues, in the Department of Mechanical Engineering, warrant thanks for their friendship.

The technical assistance of the members of the *Machine Shop* in the Department of Mechanical Engineering during the production of the calibration rig mechanism is gratefully acknowledged. I would also thank the department technician *Turan Kalender* for his efforts for supplying some of the necessary equipment.

Finally, *Ankara Atlı Spor Pony Club* members for providing me the chance of working with the horses and the horseback riders, *Tolga Aydın, Özgen Ersoy, Elif Tansu and Sercan Yılmaz*, for taking part in the experimental studies are gratefully acknowledged.

## TABLE OF CONTENTS

ABSTRACT .....	iv
ÖZ .....	vi
ACKNOWLEDGEMENTS .....	ix
TABLE OF CONTENTS .....	x
LIST OF TABLES .....	xiv
LIST OF FIGURES .....	xv
LIST OF SYMBOLS.....	xix
CHAPTERS	
1. INTRODUCTION .....	1
1.1 Objective of the Thesis .....	1
1.2 Scope of the Thesis.....	1
1.3 Literature Survey .....	3
1.4 Outline of the Thesis.....	9
2. BASICS OF HORSES.....	10
2.1 Riding Equipment.....	10
2.2 Natural Horse Gaits .....	14
2.2.1 Walk .....	14
2.2.2 Trot .....	15
2.2.3 Canter .....	16
2.3 The 3D Motion of the Horse Saddle.....	17
3. THE VISION SYSTEM .....	18
3.1 Human Vision and Computer Vision .....	18
3.2 Pinhole Perspective in Camera Projection.....	19
3.3 Introduction to Camera Calibration.....	22
3.3.1 Intrinsic Parameters .....	23
3.3.2 Extrinsic Parameters .....	23
3.4 Camera Calibration.....	24

3.5	System of 3 Cameras .....	28
3.6	Production of the Calibration Rig Mechanism .....	33
4.	OPERATIONS ON THE DIGITAL IMAGE PLANE.....	36
4.1	Definition of Digital Image .....	36
4.2	Image Types .....	37
4.2.1	Binary Image .....	38
4.2.2	Grayscale Image .....	38
4.2.3	Full-Color Image .....	39
4.3	Conversion between Image Types.....	41
4.3.1	Conversion to Grayscale from RGB or HSI Image.....	41
4.3.2	Conversion to Binary Image from RGB or Grayscale .....	42
4.4	Selection, Characteristics and Location of Markers .....	42
4.5	Marker Detection .....	44
4.5.1	Filtering via Thresholding .....	45
4.5.2	Labeling Regions.....	45
4.5.3	Filtering via Area.....	46
4.5.4	Windowing .....	47
4.5.5	Filtering via Distance .....	47
5.	OPERATIONS IN THE 3D SPACE.....	53
5.1	Conversion from Image to World Coordinates .....	53
5.2	Preprocessing in the Position Data .....	56
5.2.1	Excluding the Data .....	56
5.2.2	Smoothing the Data .....	57
5.3	Conversion from the Camera to the Motion Reference Frame .....	58
5.4	Registration of 3D Shapes .....	61
5.5	Curve Fitting to Position to Reach Velocity and Acceleration of Markers....	64
5.5.1	Goodness of Fit.....	66
5.5.2	Curve Fitting Application to the Horse Motion .....	68
5.5.2.1	Curve Fitting to Position Data in the X-Motion Direction .....	68
5.5.2.2	Curve Fitting to Position Data in the Y (Gravity) and Z Motion Directions .....	69

5.5.3 Reaching the Velocity and Acceleration Data.....	71
6. KINEMATIC PROPERTIES OF ANY POINT ON A RIGID BODY.....	74
6.1 Determination of the Angular Velocity of a Rigid Body .....	74
6.2 Determination of the Angular Acceleration of a Rigid Body.....	80
6.3 Position of the Invisible Marker .....	81
6.4 Velocity and Acceleration of the Invisible Marker .....	85
7. HUMAN RESPONSE TO VIBRATION.....	87
7.1 Whole-Body Vibration .....	87
7.2 Vibration Measurement .....	87
7.2.1 Frequency Weightings.....	88
7.2.2 Root Mean Square .....	90
7.2.3 Peak Acceleration.....	90
7.2.4 Crest Factor (CF).....	91
7.2.5 Vibration Dose Value (VDV).....	91
7.2.6 Calculation of the Daily Exposures.....	91
8. CASE STUDIES.....	95
8.1 The Camera Set Up.....	95
8.2 Calibration of the 3-Camera System .....	96
8.3 Determination of the Gravitational Acceleration via the Vision Setup.....	100
8.3.1 Operations on Images .....	100
8.3.2 Operations in 3D.....	101
8.4 Determination of the Kinematics of Horse Saddle .....	104
8.4.1 Detailed Application to Reach Kinematics of Horse Saddle.....	104
8.4.2 Saddle Marker Data of Four Horses in Each Gait .....	123
8.5 Comparison of Horse Comfort Levels.....	128
8.5.1 Different Gaits of the Same Horse .....	129
8.5.2 Different Horses Performing Each Gait .....	134
9. CONCLUSION AND FUTURE WORK .....	140
9.1 Conclusion.....	140
9.2 Future Work.....	142

REFERENCES .....	144
APPENDICES	
A. USER'S GUIDE .....	148
B. FLOW CHARTS FOR THE CODES .....	158
C. TECHNICAL DRAWINGS OF THE CALIBRATION MECHANISM.....	173

## LIST OF TABLES

### TABLES

Table 7.1 Frequency weightings with the application areas and standards.....	88
Table 7.2 EAV and ELV .....	92
Table 8.1 Intrinsic parameters of the left camera .....	98
Table 8.2 Intrinsic parameters of the middle camera .....	98
Table 8.3 Intrinsic parameters of the right camera.....	99
Table 8.4 Extrinsic parameters of each camera pair .....	99
Table 8.5 The comparison of the actual and measured distances .....	103
Table 8.6 The horses features taking part in the experiments .....	104
Table 8.7 Various horse comfort indicators .....	122
Table 8.8 The riders' grades for horse gaits .....	134

## LIST OF FIGURES

### FIGURES

Figure 1.1 The analogy between human and computer vision systems .....	3
Figure 1.2 Different calibration objects with different patterns .....	4
Figure 1.3 (a) Markers on real horse (b) Karakuri horse simulator .....	5
Figure 1.4 Structure of the Karakuri horse with audio-visual effects .....	6
Figure 1.5 Markers on saddle (b) Stewart platform .....	7
Figure 2.1 Riding equipment.....	11
Figure 2.2 A general purpose saddle .....	13
Figure 2.3 The surface $z = x_1^2 - x_2^2$ with a saddle point at (0,0).....	13
Figure 2.4 Four-beat motion sequence of legs in walk .....	14
Figure 2.5 Four-beat motion of hooves in walk .....	14
Figure 2.6 Two-beat motion sequence of legs in trot.....	15
Figure 2.7 Diagonal two-beat motion of hooves in trot .....	15
Figure 2.8 Three-beat motion of hooves in canter .....	16
Figure 2.9 Three-beat motion sequence of legs in canter.....	16
Figure 2.10 Axes for the 3D motion of the horse.....	17
Figure 3.1 Different retinal projections of two dots .....	18
Figure 3.2 Pinhole imaging model .....	20
Figure 3.3 Perspective camera projection .....	21
Figure 3.4 Thin lens.....	22
Figure 3.5 Reference frames in vision.....	23
Figure 3.6 Coordinate representation of a point in the reference frames .....	24
Figure 3.7 Reference frames of a stereo camera system .....	28
Figure 3.8 Three camera system reference frames.....	29
Figure 3.9 The modes of a flashlight between simultaneous images.....	30
Figure 3.10 Local labeling of three camera pairs .....	32
Figure 3.11 Square grid dimensions.....	34

Figure 3.12 An image of the calibration object with the mechanism showing the degrees of freedom .....	35
Figure 4.1 Image reference frame showing pixel representation .....	36
Figure 4.2 Values of pixels in a binary image.....	38
Figure 4.3 Values of pixels in a grayscale image.....	39
Figure 4.4 RGB color cube.....	40
Figure 4.5 Separated color planes of an image .....	40
Figure 4.6 HSI (or HSV) color space .....	41
Figure 4.7 A marker (a) without light (b) with light .....	43
Figure 4.8 Sensitivity regions of horses (Adapted from Micklem, 2003).....	44
Figure 4.9 4-connected and 8-connected pixels .....	46
Figure 4.10 Final marker labels to be reached at the end of all filterings .....	46
Figure 4.11 Windowing operation.....	47
Figure 4.12 A sample that needs the distance filtering operation .....	48
Figure 4.13 Illustrative example for cross product filtering.....	50
Figure 4.14 The 3D matrix array for the image coordinates found in each camera... 50	
Figure 4.15 Sample stages of the image operations .....	52
Figure 5.1 The 3D matrix array for the camera coordinates .....	54
Figure 5.2 Smoothing a set of points.....	58
Figure 5.3 Model and measured point sets.....	62
Figure 5.4 A second order polynomial curve fitting .....	65
Figure 5.5 (a) Original and fitted data (b) Residuals of fits .....	67
Figure 5.6 Divergence of a fit .....	67
Figure 5.7 Kinematic properties of a marker in the x-motion direction.....	72
Figure 5.8 Kinematic properties of a marker in the y-motion direction.....	73
Figure 5.9 Kinematic properties of a marker in the z-motion direction.....	73
Figure 6.1 Motion and body reference frames in static shot.....	82
Figure 6.2 Motion and body reference frames during the motion.....	84
Figure 7.1 Weighing coefficients vs frequency.....	89
Figure 8.1 The system for the experiment (dimensions are in mm).....	95

Figure 8.2 Images of the calibration object in different positions as viewed from the (a) left camera (b) middle camera (c) right camera.....	97
Figure 8.3 The original and area filtered left camera images of the ball.....	101
Figure 8.4 The original and filtered left camera image of the plumb .....	101
Figure 8.5 Motion of the ball in gravity direction.....	102
Figure 8.6 The points between which the distances are calculated.....	103
Figure 8.7 RGB and black & white versions of the first image .....	105
Figure 8.8 Filtering on the first image.....	106
Figure 8.9 (a) Position of the markers (b) Smoothed results of the 1 <sup>st</sup> marker .....	107
Figure 8.10 Position of 1 <sup>st</sup> marker in motion reference frame .....	108
Figure 8.11 Static shot of the markers (in left camera) .....	109
Figure 8.12 Coordinates before and after rigid body correction .....	109
Figure 8.13 Curve fitted kinematic data of 1 <sup>st</sup> marker .....	111
Figure 8.14 All versions of omega around the x axis.....	113
Figure 8.15 All versions of omega around the y axis.....	114
Figure 8.16 All versions of omega around the z axis .....	115
Figure 8.17 All smoothed versions and average of omega .....	116
Figure 8.18 The resultant omega.....	116
Figure 8.19 All smoothed versions and average of alfa .....	117
Figure 8.20 The resultant alfa.....	117
Figure 8.21 Saddle marker viewed from the left camera .....	118
Figure 8.22 Position of (a) saddle marker (b) visible and saddle markers .....	119
Figure 8.23 Velocity and acceleration of the invisible saddle marker .....	120
Figure 8.24 Kinematic data of the invisible saddle marker.....	121
Figure 8.25 Walk data of the saddle marker for the 4 horses.....	124
Figure 8.26 Sitting trot data of the saddle marker for the 4 horses .....	125
Figure 8.27 Rising trot data of the saddle marker for the 4 horses .....	126
Figure 8.28 Canter data of the saddle marker for the 4 horses.....	127
Figure 8.29 acc_y_max, acc_y_Wb_rms versus gait style for 4 different horses ..	130

Figure 8.30 crest_factor, Distance_TA_8_limit_Wb versus gait style for 4 different horses.....	131
Figure 8.31 TA_8_limit_Wb, VDV_y_Wb versus gait style for 4 different horses	132
Figure 8.32 vel_x_avg, amplitude_y versus gait style for 4 different horses .....	133
Figure 8.33 Relations between comfort indicators and grades (walk) .....	136
Figure 8.34 Relations between comfort indicators and grades (sitting trot) .....	137
Figure 8.35 Relations between comfort indicators and grades (rising trot) .....	138
Figure 8.36 Relations between comfort indicators and grades (canter) .....	139
Figure B.1 Flow diagram of the whole procedure.....	158
Figure B.2 Create results folder.....	159
Figure B.3 Calibration of each camera.....	161
Figure B.4 Stereo calibration.....	162
Figure B.5 Image centroids.....	163
Figure B.6 Combine image centroids and calibration.....	165
Figure B.7 Smoothing.....	166
Figure B.8 Camera to motion reference frame.....	167
Figure B.9 Registration.....	168
Figure B.10 Curve fitting.....	169
Figure B.11 Comfort measure.....	170
Figure B.12 Angular velocity and acceleration.....	170
Figure B.13 Required (unseen) markers data.....	171
Figure B.14 Comfort measure of the motion of required markers.....	172
Figure C.1 Technical drawing of the whole calibration mechanism.....	173
Figure C.2 Wooden part of the mechanism (Part 1).....	174
Figure C.3 Translating part of the mechanism (Part 2) .....	174
Figure C.4 Rotating part of the mechanism (Part 3) .....	175

## LIST OF SYMBOLS

$h(x_1, x_2)$	: a two dimensional function
<b>H</b>	: Hessian matrix
$(\lambda_1, \lambda_2)$	: eigenvalues of Hessian matrix
(P.O.S.)	: period of suspension
$\mathbf{A}(X_c, Y_c, Z_c)$	: a scene point in perspective camera projection
$\mathbf{a}(x_c, y_c, z_c)$	: a projected scene point in perspective camera projection
$fl$	: focal length
PP	: principal point
O, X, Y, Z	: camera reference frame in perspective camera projection
o, x, y	: image reference frame in perspective camera projection
$i, j$	: general indices
$(X_i, Y_i, Z_i)$	: world coordinates of i'th scene point
$(x_i, y_i, z_i)$	: camera coordinates of i'th scene point
$(u_i, v_i)$	: image coordinates of i'th scene point
$\mathbf{R}_{CW}, \mathbf{t}_{CW}$	: rotation matrix and translation vector between camera and world coordinate systems
$(u_0, v_0)$	: image coordinates of principal point
$(\tilde{u}_i, \tilde{v}_i)$	: image coordinates of i'th scene point in pinhole projection
$(D_u, D_v)$	: metric conversion coefficients
$s_u$	: scale factor
$\delta u_i^r, \delta v_i^r$	: infinitesimal radial displacement of i'th point
$\delta u_i^t, \delta v_i^t$	: infinitesimal tangential displacement of i'th point
$k_1, k_2$	: radial distortion coefficients
$p_1, p_2$	: tangential distortion coefficients

$O, X_L, Y_L, Z_L$	: left camera reference frame
$O, X_M, Y_M, Z_M$	: middle camera reference frame
$O, X_R, Y_R, Z_R$	: right camera reference frame
$o_L, u_L, v_L$	: left image reference frame
$o_M, u_M, v_M$	: middle image reference frame
$o_R, u_R, v_R$	: right image reference frame
fps	: frame per second
k	: frame index
$F_L, F_M, F_R$	: frames corresponding to left, middle and right cameras
$\mathbf{R}_{ML}, \mathbf{t}_{ML}$	: rotation matrix and translation vector between middle and left cameras
$\mathbf{R}_{RL}, \mathbf{t}_{RL}$	: rotation matrix and translation vector between right and left cameras
$\mathbf{R}_{RM}, \mathbf{t}_{RM}$	: rotation matrix and translation vector between right and middle cameras
$O_{\text{mech}}, X_{\text{mech}}, Z_{\text{mech}}, Y_{\text{mech}}$	: calibration mechanism reference frame
$I(u, v)$	: digital image represented as function
$\mathbf{I}$	: digital image represented as matrix
$o, u, v$	: image reference frame
$u_{\text{max}}, v_{\text{max}}$	: dimensions of a digital image
$a(a_u, a_v)$	: pixel data of point $a$
$L$	: number of gray values
$\kappa$	: number of bits to store one pixel
$T$	: image transformation function
$H(u, v)$	: transformed image
R, G, B	: red, green and blue components of a pixel in RGB image
$l$	: level of thresholding (conversion to BW from grayscale )
$[area_{\text{min}}, area_{\text{max}}]$	: acceptable area interval
$W_c$	: window center

$W_{du}, W_{dv}$	: dimensions of image window
$m$	: marker index
$M$	: number of markers
$K$	: number of images
$c$	: region index
$C$	: number of regions
$D_M$	: number of image distances between markers
$D_C$	: number of image distances between regions
$DM_{ij}$	: distances between markers $i$ and $j$
$DC_{ij}$	: distances between regions $i$ and $j$
$CD$	: number of all distance combinations
$DI_{ij}$	: distance interval
$\mathbf{r}_{ij}$	: vector from marker $i$ to $j$
$\mathbf{PM}$	: pixel matrix of markers
$\mathbf{PM}_L, \mathbf{PM}_M, \mathbf{PM}_R$	: pixel matrix of markers in left, middle and right camera image ref. frames
$\mathbf{CM}$	: position matrix of markers in camera reference frame
$\mathbf{x}_L, \mathbf{x}_M, \mathbf{x}_R$	: position vector of a point in left, middle, right camera reference frame
$\mathbf{CM}_{LM,L}$	: 3D coordinates of markers in left camera, left and middle calibrated
$\mathbf{CM}_{LR,L}$	: 3D coordinates of markers in left camera, left and right calibrated
$\mathbf{CM}_{MR,M}$	: 3D coordinates of markers in middle camera, middle and right calibrated
$\mathbf{CM}_{MR,L}$	: 3D coordinates of markers in left camera, middle and right calibrated
$\mathbf{CM}_L$	: average of 3D coordinates of markers in left camera
$\mathbf{u}_1$	: unit vector in motion direction of the selected marker

$\mathbf{u}_2^e$	: unit vector in estimated gravity direction of selected marker
$\mathbf{u}_3$	: unit vector orthogonal to $\mathbf{u}_1^s$ and $\mathbf{u}_2^{s,e}$
$\mathbf{u}_2$	: unit vector orthogonal to $\mathbf{u}_1^s$ and
$t$	: time index
$T$	: period
$\mathbf{R}_{CM}$	: rotation matrix between camera and motion reference frame
$\mathbf{x}_C$	: position vector of each marker in camera reference frame
$\mathbf{x}_M$	: position vector of each marker in motion reference frame
$\mathbf{X}_M, \mathbf{Y}_M, \mathbf{Z}_M$	: position matrices of each marker in x, y, z-motion reference frame
$\mathbf{MM}$	: 3D coordinates of all markers in motion reference frame
$e_m$	: distance error between model and measured data set
$f$	: registration objective function
$\overset{o}{\mathbf{R}}$	: optimal rotation matrix
$\overset{o}{\mathbf{q}}_t$	: optimal unit quaternion vector of translation
$\mathbf{q}_R$	: optimal unit quaternion vector of rotation
$\mathbf{q}$	: complete registration state vector
$\mathbf{x}_m$	: position vector of m'th point
$\mathbf{x}_m^m$	: position vector of m'th model point
$\mathbf{x}_m^r$	: position vector of registered m'th point
$\overset{o}{\mathbf{Q}}_{-t}$	: overall optimal translation matrix
$\overset{o}{\mathbf{RR}}$	: overall optimal rotation matrix
$\mathbf{MM}^m$	: position matrix of model points in motion reference frame
$\mathbf{MM}^r$	: position matrix of registered points in motion reference frame
$t$	: dependent variable

$t_{\max}$	: maximum value of dependent variable
$g(t)$	: one dimensional function
$g_f(t)$	: curve fitted one dimensional function
$SSE$	: sum of squares due to error
$P_x^m(t), P_y^m(t), P_z^m(t)$	: position of marker m in motion reference frame as a function of time
$g_F(t)$	: Fourier series expression
$a_n, b_n$	: Fourier series coefficients
$n$	: Fourier series coefficient index
$N$	: number of terms in Fourier series
$\omega$	: fundamental frequency of Fourier series
$\xi, \theta$	: parameters
$E$	: number of extreme values of a position function
$V_x^m(t), V_y^m(t), V_z^m(t)$	: velocity of marker m in motion reference frame
$A_x^m(t), A_y^m(t), A_z^m(t)$	: acceleration of marker m in motion reference frame
$V_{x,avg}^m$	: average velocity of marker m
$\mathbf{v}_1, \mathbf{v}_2, \text{ and } \mathbf{v}_3$	: velocity vectors of the markers 1, 2 and 3
$\mathbf{r}_{ij}$	: position vector from marker i to j
$\boldsymbol{\omega}$	: angular velocity of the rigid body
$\mathbf{K}_1, \mathbf{K}_2, \mathbf{K}_3 \text{ and } \mathbf{K}_4$	: known vectors
$\mathbf{u}$	: unknown vector
$\boldsymbol{\alpha}$	: angular acceleration of the rigid body
$\mathbf{n}$	: unit vector of the cross products of the vectors $\mathbf{r}_{21}$ and $\mathbf{r}_{31}$
$c$	: a constant
$\mathbf{u}_1$	: unit vector in the direction of the first axis of the body reference frame
$\mathbf{u}'_2$	: a temporary unit vector

$\mathbf{u}_3$	: unit vector orthogonal to $\mathbf{u}_1$ and $\mathbf{u}'_2$
$\mathbf{u}_2$	: unit vector orthogonal to $\mathbf{u}_1$ and $\mathbf{u}_3$
$\mathbf{R}_{MB}$	: rotation matrix between motion and body reference frame
$P_x^s(t), P_y^s(t), P_z^s(t)$	: position of saddle marker in motion reference frame
$V_x^s(t), V_y^s(t), V_z^s(t)$	: velocity components of saddle marker in motion reference frame
$A_x^s(t), A_y^s(t), A_z^s(t)$	: acceleration components of saddle marker in motion reference frame
$\mathbf{v}_s(t)$	: velocity vector of saddle marker
$\mathbf{a}_s(t)$	: acceleration vector of saddle marker
$f$	: frequency
$W_b(f), \dots, W_m(f)$	: frequency weighting functions
$A_{wb}(t), A_{wk}(t)$	: weighted accelerations in y- motion direction
$a_{wb, rms}, a_{wk, rms}$	: root mean square value of weighted accelerations
$\tau$	: measurement duration
$a_{wb, peak}, a_{wk, peak}$	: peak value of weighted accelerations
CF	: crest factor
A(8)	: total daily exposure in r.m.s
$T_{A(8)}$	: time to reach the A(8) threshold
VDV	: vibration dose value
$VDV_{threshold}$	: threshold value for VDV
$T_{VDV}$	: time to reach $VDV_{threshold}$
EAV	: Exposure Action Value
ELV	: Exposure Limit Value
$D^{action}$	: distance to be travelled for action
$D^{limit}$	: distance to be travelled for limit

## **CHAPTER 1**

### **INTRODUCTION**

#### **1.1 Objective of the Thesis**

There are two main objectives of this study. One of the objectives is to develop a modular vision system that consists of three standard, commercial cameras. This system, as a whole with its hardware and software, may be utilized for passive marker tracking operations. By tracking passive markers, one may obtain kinematical properties (position, linear and angular velocity and acceleration, etc) of required points and/or objects in 3D space.

The second objective of this study is to use the aforementioned vision system to obtain the kinematical properties of saddles belonging to various horses in their three natural gaits, namely walk, trot (sitting and rising) and canter. The differences between the gaits of the same horse and the differences between the gaits of different horses (executing the same gait) are to be examined. Furthermore, the relationship between the saddle motion and the comfort level of the rider on the horse will be investigated.

#### **1.2 Scope of the Thesis**

One of the main requirements for setting up a multi-camera vision system is the process of camera calibration. Briefly, camera calibration is the process of finding the intrinsic and extrinsic parameters of a camera which describes the relationship between the 2D image coordinates and the 3D world coordinates of required points.

In order to apply the calibration algorithm, a planar checkerboard patterned object is necessary. Since the calibration volume is much larger than usual applications, the rig has to be large, portable and easily rotatable. A mechanism mounted on a four wheeled chassis (for translation) and having two degrees of freedom (for rotation around the horizontal and vertical axes) is designed and manufactured.

Since the recorded movies have to be extracted into image frames, readily available software, AVD Video Processor<sup>®</sup>, is used for this purpose. After the extraction of the frames, the images have to be processed by means of various algorithms. A general algorithm, for obtaining the coordinates of a point (stationary, or moveable) in 3D space is developed in accordance with the vision set up in MATLAB<sup>®</sup>. Using the developed program, motion analysis of the saddles of different horses in different gaits has been performed. The aforementioned experimentation is executed in an enclosed arena (manege) in Ankara Pony Club using 4 horses each of which is marked with passive markers made from special 3M<sup>®</sup> tapes.

During riding a horse, most parts of the saddle are invisible to the cameras because of the rider's position. Hence in this application, markers are attached on the body of the horse. Smoothing the position of each marker in the camera reference frame is followed by representing the coordinates in a desired reference frame (named as motion reference frame). The rigid body constraint is applied to the markers. The position analysis of the saddle is then followed by analysis of other kinematic properties, such as linear velocity and acceleration, after suitable curve fitting operations. Angular kinematic properties can also be obtained, as long as at least three markers have been used. The relation between the markers on the horse and on the saddle is determined in the next step. The vibration on the saddle is of interest; therefore acceleration is the main property to be analyzed in detail. A number of vibration measure indicators, such as root mean square and peak acceleration, acceptable riding durations, etc... are considered. Finally, the grades that the horseback riders have given to the comfort of the riding are to be used as a rating

scale among the horses and compared with the data obtained from the vibration measure indicators.

### 1.3 Literature Survey

Early researches about capturing humans and animals in motion using single or multiple cameras were firstly done by Muybridge (1887). Among the animals, horses have been of primary importance and the initial analysis of the motion of different horses has been obtained from the sequence of image frames.

According to Forsyth and Ponce (2003), in order to obtain the 3D metric information of objects from the 2D image coordinates, a vision system consisting of more than one camera should be constructed. The authors state that when at least two cameras are separated by a distance and take the picture of the same scene, similar to human eyes, the depth information can be reached. This is known as stereopsis (Figure 1.1).

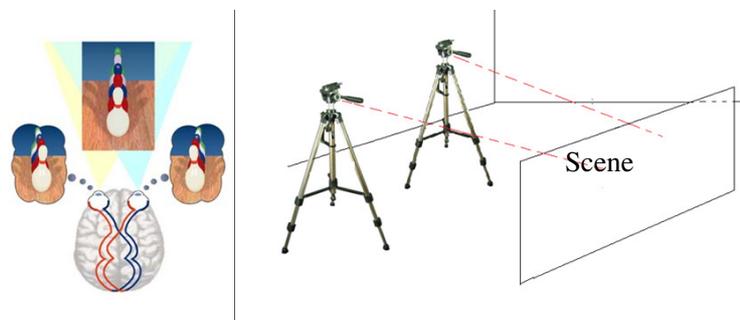


Figure 1.1 The analogy between human and computer vision systems

The authors also verify that in order to find the relation between the image and world coordinates of points, in other words to solve the calibration problem, a calibration

rig patterned by grids drawn in orthogonal planes should be formed. Figure 1.2 shows different calibration objects.

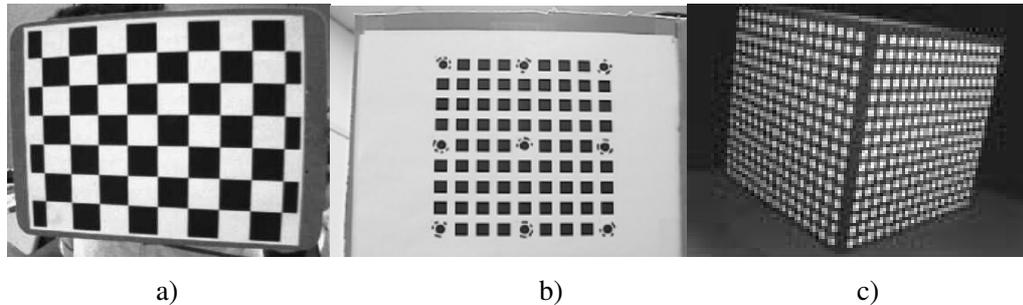


Figure 1.2 Different calibration objects with different patterns  
(a) Bouguet (2008) b) Zhang (2000) c) Heikkilä & Silvén (1997))

In this respect, Zhang (2000) suggests that the cameras observe a planar pattern as a model at a few (at least two) different orientations to extract the coordinates of several points. However, Zhang's study includes only modeling of the radial distortions. However, Heikkilä and Silvén (1997) suggest a four-step camera calibration technique compensating both for the radial and tangential distortions. As inspired from the studies of Zhang, Heikkilä and Silvén, Bouguet (2008) has released a practical and functional MATLAB<sup>®</sup> toolbox for camera calibration.

There are many researches on tracking active or passive markers, in real time or offline, that are placed on the bodies of humans or animals. Based on the researches, commercial motion capture systems, such as VICON<sup>®</sup>, SIMI<sup>®</sup>, CODAMOTION<sup>®</sup> and QUALISYS<sup>®</sup> have been constructed. Basically, identification of points can be achieved by using fundamental image processing techniques as indicated by Gonzalez and Woods (2002).

Having mentioned the general approach to vision in the literature, reasons and results of the studies based on horses are indicated next. The main reason for investigating the motion of horses is to construct riding simulators that give the feeling of a horse. Simulators designed by Armen (1990) and Cannovino (2001) are driven by the user by means of springs and/or switches. The simulator invented by Yamagushi, Kito, Iguchi, Yoshida and Ishigure (1991) is driven by one or more motors and includes a control unit. Common purposes for designing such simulators are entertainment, training the handicapped children and giving the basic leg movements of horses. The vibrations generated by the swinging (up and down, back and forth and left and right) during riding a horse assist the therapy of handicapped or autistic people. Kijima et al. (2003) shows that after riding such a simulator, the muscles are strengthened due to the relaxation and tension and the sense of balance is improved due to the high eye position. The simulator used by Kijima et al. is inspired from a traditional technique in Japan called Karakuri, or puppeteering (Sato et al., 1999). In the study of Sato et al. (1999), the periodical data for the simulator is taken from the video recording of the markers on the joints of the legs of a real horse as shown in the Figure 1.3 (a). The joints of the simulator are driven by the motors connected to each foot and controlled by changing the angular velocity of the motors as shown in the Figure 1.3 (b).



Figure 1.3 (a) Markers on real horse (b) Karakuri horse simulator

Furthermore, in order to enhance the feeling and increase the realism of the simulator, audio-visual effects, synchronized with the motion, are added (Figure 1.4). The comparison between a real and an artificial horse is made and the results are found to be satisfactory.

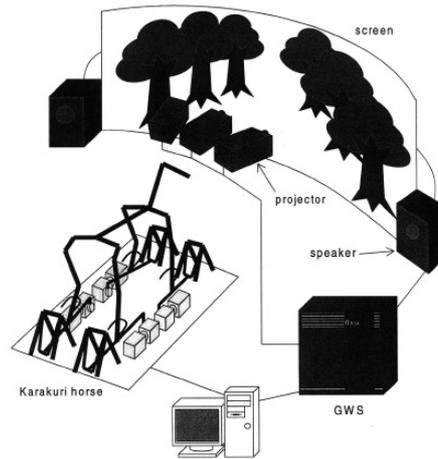


Figure 1.4 Structure of the Karakuri horse with audio-visual effects

Shinomiya et al. (2003) demonstrate the favorable effects of passive exercises, like horseback riding, which increase the muscular strength among the elderly and which induce the muscular contraction among obese patients with type 2 diabetes. Owing to the fact that securing trainers, training places and breeding horses have high costs for the tests, Shinomiya, Nomura, Yoshida and Kimura (1997) have constructed a system that can reproduce the saddle movement. To this purpose, the saddle is marked with markers as seen in Figure 1.5 (a), and afterwards data is collected by using six high speed cameras surrounding the motion scene. A Stewart platform-shaped parallel mechanism is chosen in the design reproducing the motion in two, three or six axes (Figure 1.5 (b)). The system is composed of drive, control, virtual reality (VR) and operating units. Speed, direction and walking style changes are claimed to be performed. Riders, having different backgrounds of horseback riding

have been exposed to evaluation of the simulator performing walking gait. The results show that as the number of axes introduced by the mechanism increases, scores awarded by the riders increase but in practice it requires more labor and cost.



(a)



(b)

Figure 1.5 Markers on saddle (b) Stewart platform

Another reason for investigating horses in motion is to model them in computer environment. Koenig and Bekey (1993) construct a kinematical model of a horse in computer environment and present the 3D model in different gaits including variations. Although the hardware instrumentation is under construction, the authors claim that the resulting animations look realistic.

Yet another motivation for the studies on the motion of horses is the detection of lameness. Particularly, veterinarian faculties have researches in this respect. Studies are based on the diagnosis of problems during movement in laboratories with special equipment (such as the readily available systems) which may help to rehabilitate horses. To demonstrate, Keegen et al. (2003) tracks the retroreflective spherical markers attached to the head and/or body of horses trained to trot on a treadmill.

High speed cameras (120 frames/ second) are preferred to record the horse's movement. 2-dimensional position of each marker is collected and the obtained data is converted into a 3-dimensional sequence of the horse's movement using the Vicon Motion Analysis System. From the outputs of the analysis, horses showing symptoms of lameness can be detected.

Having mentioned the benefits of horseback riding, the side effects of the vibrations exposed on the rider during the motion should also be considered. However, to the author's knowledge, there doesn't exist any studies on this topic in the literature. Yet, researches about human exposure to vibration can be adapted to the vibrations induced on the saddle of a horse. As Mansfield (2005) indicates, whole-body vibration should be investigated when people are exposed to vibrations in industrial environment or in transportation systems. Vibrations can be measured by means of acceleration and frequency, where accelerometers are the suggested devices for measuring accelerations. At the end of the analysis of the vibration measure indicators, maximum allowable duration of exposures to vibrations on horse saddle is to be reached. Rimell and Mansfield (2007) give the implementation of frequency weighting filters as digital filters with the formulae of filtering coefficients at any frequency. The presented formulae can be implemented in software after doing necessary modifications in the domain basis.

For practical purposes, saddles may be assumed to be rigid. However, acceptable variations in the constant dimensions (or between markers attached on the body of horses) may occur during the motion, and/or the vision system may yield results indicating falsely that the dimensions are varying in each frame. In order to remedy this situation, the 3D object registration algorithm proposed by Besl and McKay (1992) may be used. Their approach gives the best alignment of a measured, erroneous data set with a model data set.

## **1.4 Outline of the Thesis**

An introduction to the thesis is given in Chapter 1, including the scope and objectives of the study with a survey of literature.

Chapter 2 presents a brief summary of the horse riding equipment and the features of natural horse gaits, namely walk, trot and canter. In Chapter 3, the basics of a vision system such as camera projections, internal and external camera calibration parameters and the calibration procedure are explained. Chapter 4 gives an introduction to digital images. The algorithms developed are explained, such as marker detection by means of filters. In Chapter 5, the details of operations performed in the world coordinates are introduced. Conversions from image to camera frame and camera to any other reference frame are discussed. Furthermore, a registration method for 3D objects is presented. A general approach to the curve fitting process and the goodness of fits are presented. Chapter 6 presents an approach to reach the angular properties of a rigid body. It is followed by reaching the kinematical data of any invisible marker tracking algorithm. In Chapter 7, measures for the human response to whole-body vibration are presented. The action and limit values of vibration duration are introduced within the standards. In Chapter 8, the results of various experiments, performed by using 4 horses, are given as case studies.

Finally, Chapter 9 presents a brief summary of the work in accordance with the conclusions. Several suggestions for future work are also presented.

## **CHAPTER 2**

### **BASICS OF HORSES**

Humans began to domesticate horses thousands of years ago not only for their assistance in various areas such as agriculture, warfare, but also in entertainment, sports, races and therapy. Horses are gentle creatures and have strong instincts which bring an advantage in training (Micklem, 2003). One of the main benefits of riding a horse as a sport is that both men and women, and young and mature can improve the quality of their lives. However, horses can also be frightening and risk generating animals because of an exposure to an unexpected effect coming outside, or because of rider's behaviors going beyond his/her current level of ability.

#### **2.1 Riding Equipment**

In Figure 2.1, the main pieces of a horse's riding equipment (or tack) can be seen. For the health and comfort of both the horse and rider, a supportive structure, namely saddle, is fastened tightly to the back of the horse by means of girth. Saddle, regardless of type, should properly fit both the horse and the rider and be positioned so that the rider settles over the horse's center of balance (Figure 2.2). A loose girth should be avoided such that the tightened girth should be tightened a little more after 10-15 minutes when the relaxations of the horse and the equipment are realized. Indeed, a badly fitting saddle may result in slippage of the saddle and rider from the horse.

Bit is a piece of metal held in the mouth of the horse by the bridle. Reins attached to the bit and the stirrups attached to the saddle are the controllers of the horse. In other words, the rider directs the motion of the horse by means of the reins and stirrups. Saddle cloth (pad or blanket) may also be used to protect the saddle from dirt and sweat. Some saddle clothes are designed such that the shocks are absorbed and the damage on the back muscles of the horse is lessened.



Figure 2.1 Riding equipment  
(Adapted from Micklem, 2003)

Saddles are designed in a special shape such that they fully cover the back of the horse, while making the rider have the correct sitting position. The surface of the

saddle is similar to the surface obtained in Figure 2.3 by means of computer software. The surface plotted in Figure 2.3 is given by

$$z = h(x_1, x_2) = x_1^2 - x_2^2 \quad (2.1)$$

Clearly, the critical points are obtained from the solution of the 2 equations given by;

$$\frac{\partial h}{\partial x_1} = 2x_1 = 0, \quad \frac{\partial h}{\partial x_2} = -2x_2 = 0 \quad (2.2)$$

which yield  $x_1 = x_2 = 0$ . The second partial derivatives, on the other hand, are given by;

$$\frac{\partial^2 h}{\partial x_1^2} = 2, \quad \frac{\partial^2 h}{\partial x_2^2} = -2, \quad \frac{\partial^2 h}{\partial x_1 x_2} = 0 \quad (2.3)$$

which lead to the Hessian matrix,  $\mathbf{H}$ ;

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 h}{\partial x_1^2} & \frac{\partial^2 h}{\partial x_1 x_2} \\ \frac{\partial^2 h}{\partial x_1 x_2} & \frac{\partial^2 h}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 2 & 0 \\ 0 & -2 \end{bmatrix} \quad (2.4)$$

Since the Hessian matrix evaluated at the critical point  $(0, 0)$  yields one positive and one negative eigenvalue, i.e. 2 and -2, the critical point  $(0, 0)$  corresponds to a saddle point. Clearly, this point is neither a minimum nor a maximum.

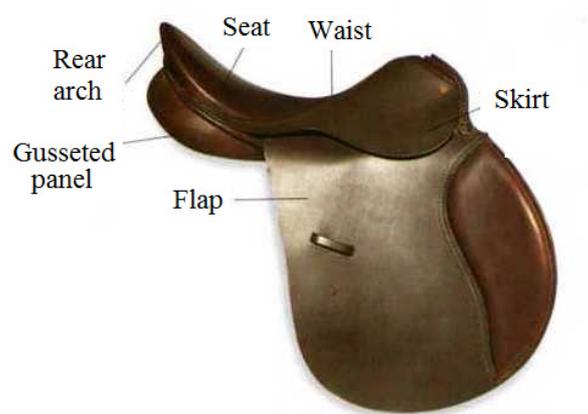


Figure 2.2 A general purpose saddle  
(Adapted from Micklem, 2003)

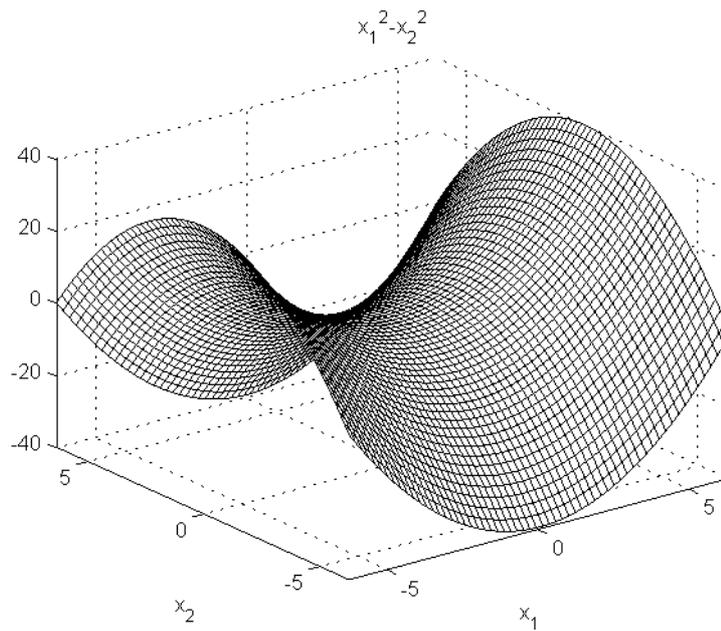


Figure 2.3 The surface  $z = x_1^2 - x_2^2$  with a saddle point at (0,0)

## 2.2 Natural Horse Gaits

Basically, there are two types of gaits, natural gaits known by horses from birth and trained gaits taught by order. The natural gaits are walk, trot and canter in the order of increasing speed which have different motion characteristics.

### 2.2.1 Walk

Walk is a four-beat natural movement (Figure 2.4 and Figure 2.5). The horse touches his legs (or hooves) on the ground as; 1st beat: right hind leg, 2nd beat: right foreleg, 3rd beat: left hind leg, 4th beat: left foreleg. Among all paces, it is the most stable one, since at least two hooves are always on the ground (Micklem, 2003). The slowest gait of horse is walk with a velocity of 5-7 km/h.

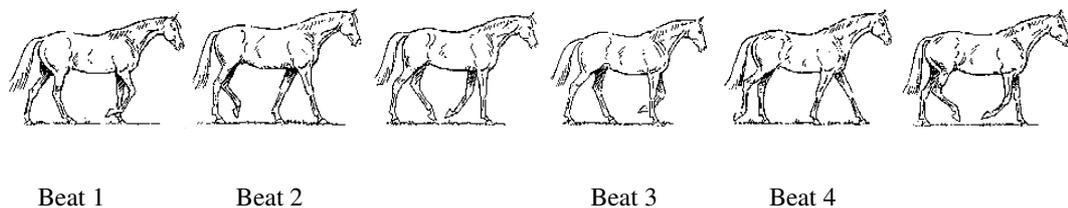


Figure 2.4 Four-beat motion sequence of legs in walk

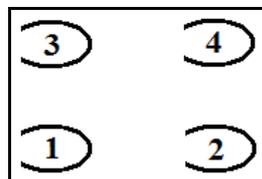


Figure 2.5 Four-beat motion of hooves in walk

### 2.2.2 Trot

There are two beats in the trot stride. A period of suspension (P.O.S.) in each beat occurs as all four legs leave the ground. The legs move in diagonal pairs. One of the most distinguishing characteristics of trot is the dominant rising up and down motion in each beat, called posting. The sequence of the legs and hooves are shown in Figure 2.6 and Figure 2.7. Trot is a gait faster than walk and slower than canter with an average velocity of 8-10 km/h.

The type of trot where the rider stays in contact with the saddle, feeling as if he is in a car exposed to bumps at regular periods, is called sitting trot. In the rising trot, on the other hand, the rider avoids the bumps by rising out of the saddle in synchronization with the horse's motion.

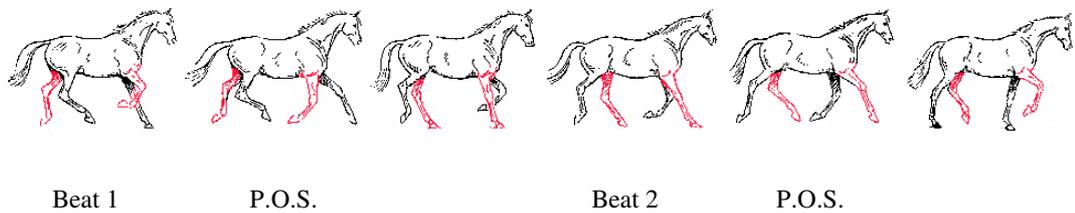


Figure 2.6 Two-beat motion sequence of legs in trot

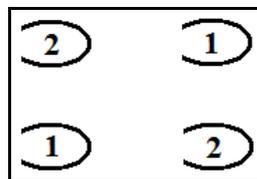


Figure 2.7 Diagonal two-beat motion of hooves in trot

### 2.2.3 Canter

Canter is a three-beat pace with two types: left-lead canter and right-lead canter (Figure 2.8). Figure 2.9 shows the motion frames of the left-lead cantering horse since the horse's left front leg is hitting the ground in front of the right. The motion starts with hind leg touching the ground first, leading to the diagonal pair of left hind leg and right foreleg. Beat 3 occurs when the (leading) left foreleg touches the ground while the right hind leaves. It ends with a period of suspension.

Average velocity of a horse in canter is 16-20 km/h. Also, the canter becomes a gallop if the diagonal legs split up causing each leg to touch the ground separately. Therefore, the gait has a four-beat sequence and higher velocity than canter.

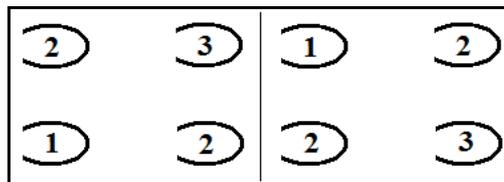


Figure 2.8 Three-beat motion of hooves in canter

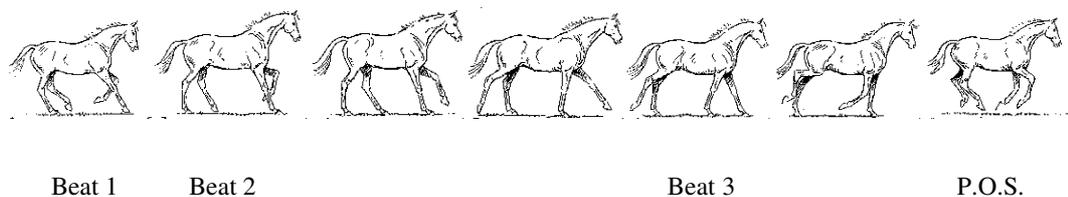


Figure 2.9 Three-beat motion sequence of legs in canter

### 2.3 The 3D Motion of the Horse Saddle

Horse saddle motion has six degrees of freedom. The motion is decomposed in three orthogonal axes. The cranial and caudal axes (Figure 2.10) correspond to the forward and backward motions, respectively. Up and down (vertical) movement is shown by the proximal and distal axes. Finally, the dorsal and ventral axes make up the side to side (lateral) movement.

For the sake of simplicity, x-motion, y-motion and z-motion axes are used in this thesis. The positive x-motion, y-motion axis and z-motion axes correspond to the *cranial*, *distal* and *dorsal* axes, respectively.

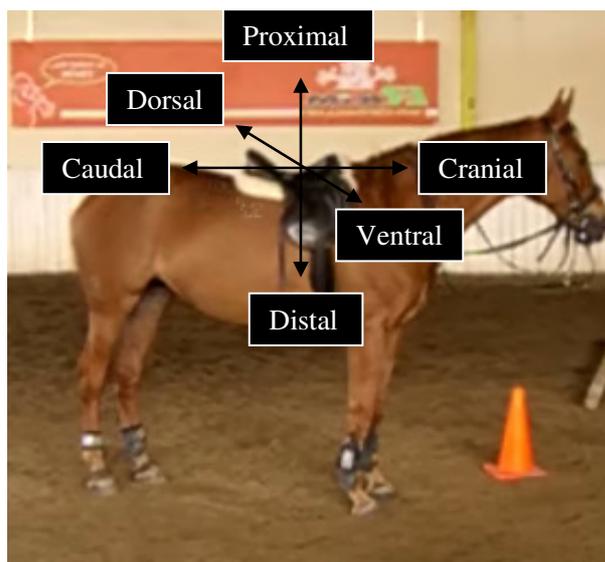


Figure 2.10 Axes for the 3D motion of the horse

## CHAPTER 3

### THE VISION SYSTEM

The resemblance between the human vision system and the systems that are utilized to reconstruct multiple views obtained from cameras is introduced in this chapter. In both cases, at least two views of the same scene are required for the reconstruction of the scene in 3D.

#### 3.1 Human Vision and Computer Vision

For humans and many animals, one of the most important tasks of the brain is to achieve the reconstruction of the world surrounding and the most important sense is the visual sense to accomplish this task (Mansson, 1998).

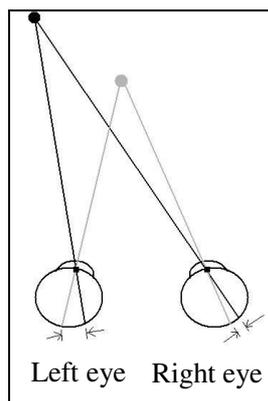


Figure 3.1 Different retinal projections of two dots

In order to acquire the full representation of the space, the relation between the 3D world and two separate 2D images perceived by the two eyes must be obtained. Since the eyes are located with some distance in between, the same 3D scene is viewed slightly different by each eye. Brain is capable of measuring this disparity and using it to produce the sensation of depth, which is referred as stereopsis (see Poggio & Poggio, 1984). Figure 3.1 illustrates the different retinal projections of two dots in two eyes which are caused by the difference in perspective. There exists a number of studies based upon the reconstruction and modeling of the human vision system by means of dynamic programming, correlation-based stereo matching and multiscale binocular fusion algorithms (Marr & Poggio, 1979).

As inspired from human and animal vision, computer vision is an enterprise which deals with artificial vision systems implemented in software and hardware. Analogous to humans having two eyes, depth can be measured with at least two images taken by separated cameras at the same time. 2D image information of objects is to be reconstructed in the 3D scene by a camera calibration procedure. To accomplish various tasks, such as object recognition, modeling, motion estimation, etc... geometry, physics and statistics are the main fields to consider.

### **3.2 Pinhole Perspective in Camera Projection**

Pinhole perspective projection model is a very simple approximation of the imaging process (Forsyth & Ponce, 2003). Perspective projection generates the inverted image, namely the virtual image in front of the pinhole at the same distance obtained in the actual image plane (see Figure 3.2). As stated before, pinhole perspective is an approximation of the real camera projection where the aperture area is infinitesimal, a point. In reality, cameras have a pinhole of a finite size and they are equipped with lenses. Lenses allow gathering light rays and keep pictures in sharp focus coming from a finite area.

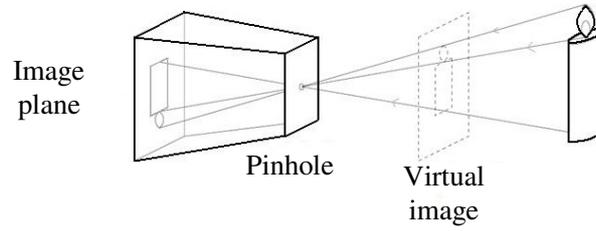


Figure 3.2 Pinhole imaging model

In Figure 3.3, the perspective camera projection model is illustrated.  $Z$  coordinate of the camera reference frame constitutes the principal axis which is perpendicular to the image plane. Intersection of the image plane and the principal axis is called the principal point. As illustrated, a scene point  $\mathbf{A}$ , has the coordinates  $(X_c, Y_c, Z_c)$  and on the image plane the same point's projection, indicated by  $\mathbf{a}$  has the coordinates  $(x_c, y_c, z_c)$ , both represented in the camera reference frame. Evidently, using similar triangles one can obtain the relations below:

$$\frac{x_c}{X_c} = \frac{z_c}{Z_c}, \quad \frac{y_c}{Y_c} = \frac{z_c}{Z_c}, \quad z_c = fl \quad (3.1)$$

From these relations, the coordinates of the projected point can be derived as:

$$x_c = fl \frac{X_c}{Z_c}, \quad y_c = fl \frac{Y_c}{Z_c} \quad (3.2)$$

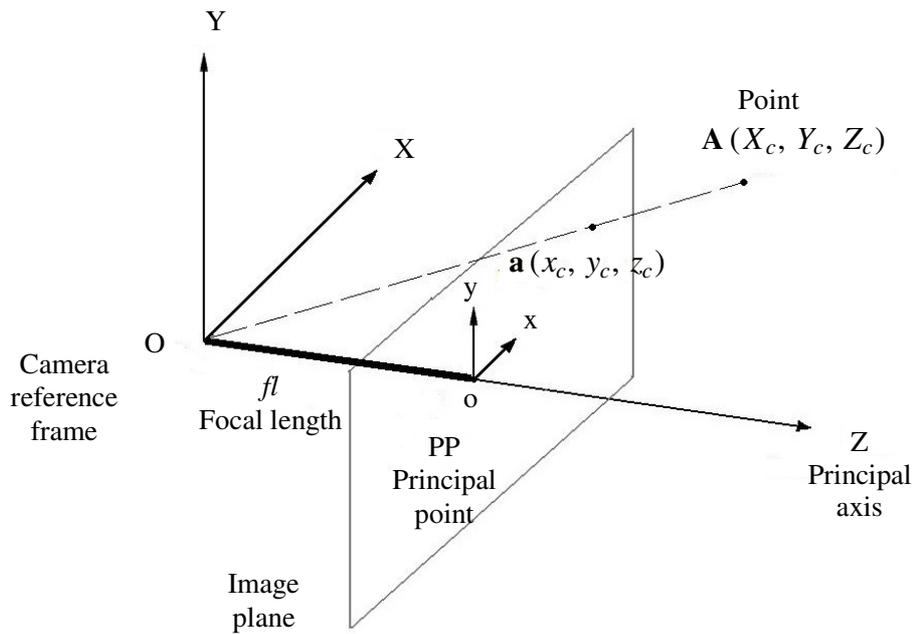


Figure 3.3 Perspective camera projection

Most cameras are equipped with lenses. Figure 3.4 shows a thin lens model which is the basics of more complicated optical systems. In reality, lenses are exposed to aberrations or distortions during manufacturing. Those distortions, named as radial and tangential distortions, should be taken into account. Distortion corrections should be applied in order to reach further results in success.

The equation for the thin lens model is given by:

$$\frac{1}{Z + fl} + \frac{1}{z + fl} = \frac{1}{fl} \quad (3.3)$$

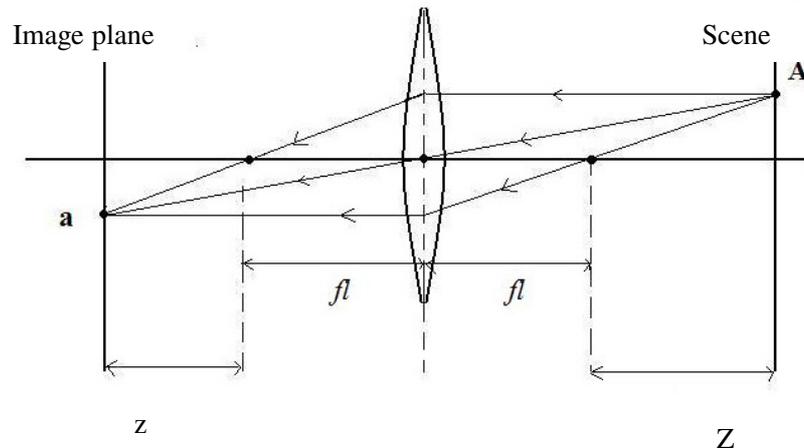


Figure 3.4 Thin lens

### 3.3 Introduction to Camera Calibration

In order to find the mapping between 2D images and the 3D world coordinates, in other words, to extract metric information from 2D images, camera calibration is a vital step. The success of a camera vision system depends on the accuracy of the calibration. For camera calibration, there exist several methods in the literature originating from photogrammetry, including minimization of a nonlinear error function, closed-form solutions or the application of the two methods. In this thesis, single and stereo camera calibrations are performed by using the toolbox of Bouguet (2008), written in MATLAB<sup>®</sup>.

It should be noted that Equation (3.2) is valid only for the case that distances are measured in the camera reference frame and the image center is coincident with the principal point. Therefore, the solution of the calibration problem requires that the relations between the three coordinate systems, namely image, camera and world (or the second camera of a camera pair) coordinate systems (or reference frames) are determined. The geometrical and optical parameters that describe the transformations

between these coordinate systems are named to be the camera calibration parameters (see Figure 3.5) which may be classified as intrinsic and extrinsic parameters.

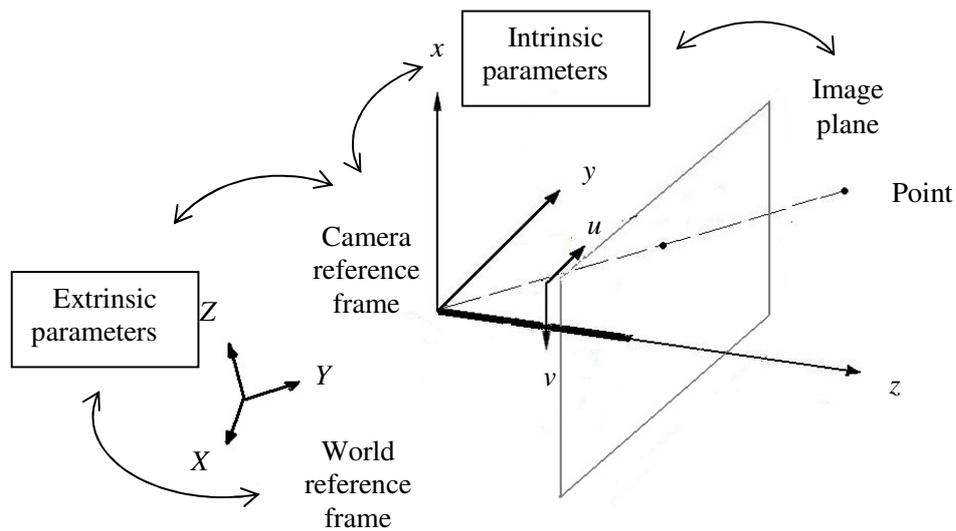


Figure 3.5 Reference frames in vision

### 3.3.1 Intrinsic Parameters

The internal physical characteristics of a camera are named to be intrinsic parameters. These parameters do not depend on the camera location. Focal length, scale factor, lens distortions and the coordinates of the image center (or the principal point) are the intrinsic parameters, which relate the camera reference frame to the image reference frame. (See Section 3.4 for the definitions.)

### 3.3.2 Extrinsic Parameters

The extrinsic camera parameters are required to obtain the relationship between the camera and world reference frames (or, between two cameras in a stereo camera

system). In particular, when multiple cameras are in use, these parameters describe the transformation between the different camera reference frames. Therefore, extrinsic (or, external) parameters depend on the locations of the cameras.

### 3.4 Camera Calibration

The MATLAB<sup>®</sup> calibration toolbox introduced by Bouguet (2008) and utilized in this study is based upon the approach of Heikkilä and Silvén (1997). Both the tangential and radial lens distortions are considered in their approach, whereas in the approach of Zhang (2000), only the radial distortions are taken into account. In this section, the details of Heikkilä and Silvén's (1997) calibration method are explained.

Let us consider a point at  $(X_i, Y_i, Z_i)$  in the world coordinates. In order to express this point in the image coordinates, firstly, transformation to the camera coordinates,  $(x_i, y_i, z_i)$ , is necessary (see Figure 3.6). In the transformation between the world and camera coordinates, the extrinsic parameters are utilized.

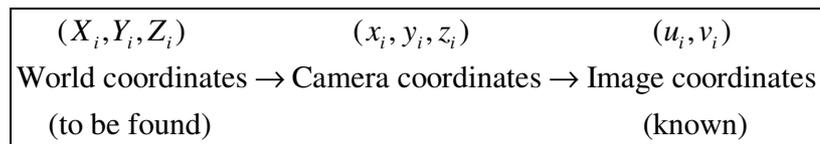


Figure 3.6 Coordinate representation of a point in the reference frames

The transformation between the world and camera coordinates may be represented by the relation:

$$\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \mathbf{R}_{\text{CW}} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + \mathbf{t}_{\text{CW}} \quad (3.4)$$

where  $\mathbf{R}_{\text{CW}}$  is the rotation matrix relating the world and camera coordinates and  $\mathbf{t}_{\text{CW}}$  is the translation vector (between the origins of the world and camera reference frames.) It should be noted that  $\mathbf{R}_{\text{CW}}$  and  $\mathbf{t}_{\text{CW}}$  are the extrinsic parameters that are obtained as a result of camera calibration.

In the second step, the intrinsic parameters are utilized. Based on the pinhole model, projection of the point at  $(x_i, y_i, z_i)$  on the image plane can be obtained as in Equation (3.5) where  $(\tilde{u}_i, \tilde{v}_i)$  represents the pinhole projection coordinates (in metric units) where the origin is at the principal point with coordinates  $(u_0, v_0)$ :

$$\begin{bmatrix} \tilde{u}_i \\ \tilde{v}_i \end{bmatrix} = \frac{fl}{z_i} \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (3.5)$$

Image coordinates are to be represented in the unit of pixels; therefore metric units should be converted to pixels. The coefficients,  $D_u$  and  $D_v$  are introduced to perform this conversion. Furthermore the origin of the image coordinate system is in the upper left corner of the image array. Therefore, the coordinates (in pixels) of any point  $(x_i, y_i, z_i)$  with the origin in the upper left corner of the image array is given by;

$$\begin{bmatrix} u'_i \\ v'_i \end{bmatrix} = \begin{bmatrix} D_u s_u \tilde{u}_i \\ D_v \tilde{v}_i \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (3.6)$$

where  $s_u$  is a scale factor.

Since the pinhole model is an approximation of the real camera model, a more comprehensive model should be used. At that point, distortion coefficients are introduced to provide corrections for the distorted image coordinates. Distortions have radial and tangential components which can be modeled with the following expressions, respectively:

$$\begin{bmatrix} \delta u_i^r \\ \delta v_i^r \end{bmatrix} = \begin{bmatrix} \tilde{u}_i(k_1 r_i^2 + k_2 r_i^4 + \dots) \\ \tilde{v}_i(k_1 r_i^2 + k_2 r_i^4 + \dots) \end{bmatrix} \quad (3.7)$$

$$\begin{bmatrix} \delta u_i^t \\ \delta v_i^t \end{bmatrix} = \begin{bmatrix} 2p_1 \tilde{u}_i \tilde{v}_i + p_2 (r_i^2 + 2\tilde{u}_i) \\ p_1 (r_i^2 + 2\tilde{v}_i) + 2p_2 \tilde{u}_i \tilde{v}_i \end{bmatrix} \quad (3.8)$$

Here,

$k_1, k_2, \dots$ : coefficients for radial distortion.

$p_1, p_2$ : coefficients for tangential distortion.

$$r_i = (\tilde{u}_i^2 + \tilde{v}_i^2)^{1/2} \quad (3.9)$$

It should be noted that the radial distortions cause the actual image point to be radially displaced in the image plane, whereas the tangential distortions cause the actual image point to be tangentially displaced in the image plane. The tangential distortions are due to decentering, i.e. imperfections in the centering of the lens components and manufacturing deficiencies. Additionally the deficiencies in the collinearity in the centers of curvature of lens surfaces introduce the tangential component. In addition to tangential and radial distortions, linear and thin prism distortions are also proposed in the literature. However, according to Heikkilä and Silvén (1997), in most cases, the error due to the linear and thin prism distortions is small and the distortion components are insignificant. For this reason, only tangential

and radial distortions are compensated. The authors state that one or two radial distortion coefficients are enough to compensate for the distortions. Hence, for radial distortions,  $k_1$  and  $k_2$ , for tangential distortions,  $p_1$  and  $p_2$  are employed in the equations.

After mentioning the reasons and effects of the distortions, a suitable camera model, including the pinhole projection with the components of the distortion correction coefficients, is given by:

$$\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} D_u s_u (\tilde{u}_i + \delta u_i^r + \delta u_i^t) \\ D_v (\tilde{v}_i + \delta v_i^r + \delta v_i^t) \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix} \quad (3.10)$$

where  $(u_i, v_i)$  is the corrected (with respect to the radial and tangential distortions) version of  $(u'_i, v'_i)$  in Equation (3.6).

Therefore, the number of intrinsic parameters to be estimated during camera calibration is 10. These parameters are the effective focal length,  $fl$ ; the scale factor,  $s_u$ ; the coordinates of the principal point,  $(u_0, v_0)$ ; pixel to metric conversion coefficients,  $(D_u, D_v)$ ; the coefficients of radial distortion,  $k_1, k_2$  and the coefficients of tangential distortion,  $p_1, p_2$ . The number of intrinsic parameters to be estimated is constant, whereas the number of extrinsic parameters depends on the number of camera views. For each camera, the number of extrinsic parameters to be estimated is six. Three of these parameters are associated with  $\mathbf{R}_{CW}$  and the remaining 3 are the components of  $\mathbf{t}_{CW}$  (see Equation 3.4).

Having obtained the relations between the reference frames which contain intrinsic and extrinsic parameters, the calibration problem has to be solved. Among a number of methods proposed for this purpose, the authors prefer to get advantage of the

method suggested by Wei and Ma (1993). For the details of the formulation, refer to Heikkilä and Silvén (1997).

During the application of the toolbox, firstly, each camera has to be calibrated in order to reach the intrinsic parameters. Then a stereo calibration should be applied to find the location of the cameras with respect to each other. As a result, any point's world coordinates (within the calibration volume) with respect to any chosen camera can be attained. Figure 3.7 illustrates the reference frames of a stereo camera system.

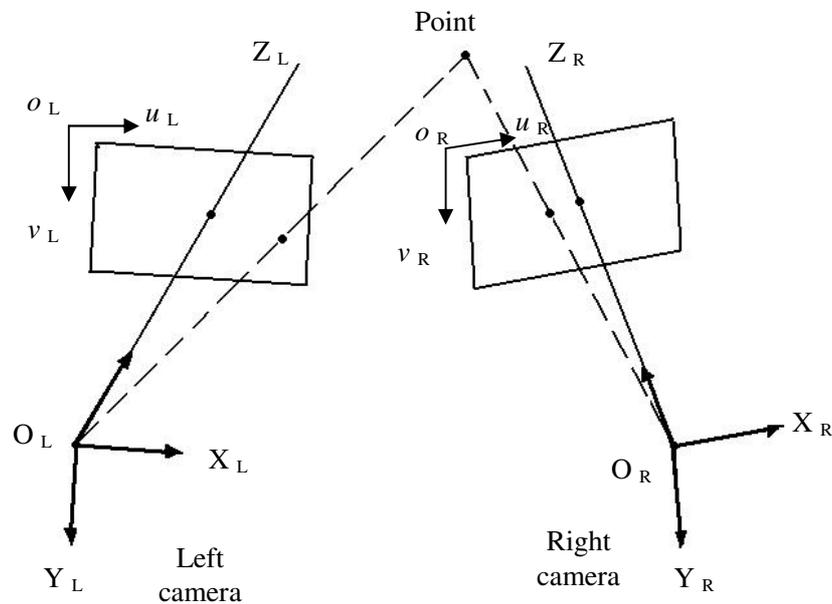


Figure 3.7 Reference frames of a stereo camera system

### 3.5 System of 3 Cameras

In this study, a trio vision system consisting of 3 cameras is employed in order to obtain more accurate results (compared to a stereo vision system). Refer to Figure 3.8 to visualize the trio camera system.

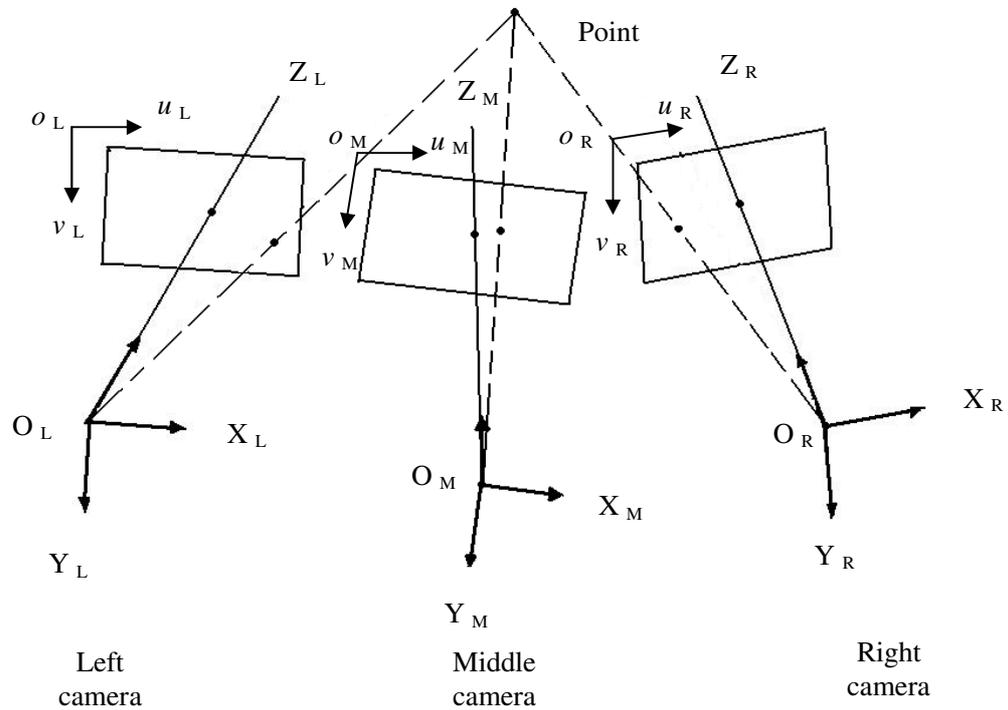


Figure 3.8 Three camera system reference frames

In the calibration process of the three camera system, firstly, each camera is calibrated with the calibration toolbox of Bouguet (2008) in order to attain the intrinsic parameters belonging to each camera. The properties of the cameras used in this study are given below:

- Model of the camera: Casio<sup>®</sup> Ex-Z750.
- Resolution at video mode: 640 x 480 pixels.
- Memory card type: Kinston 2 GB.
- Maximum duration of video recording: 1 hour for 2 GB card.
- Frame rate: 30 fps.

Since the resolutions of the images should be the same for both the calibration and the motion recordings, the video mode is used for the calibration recordings as well. AVD Video Processor<sup>®</sup> is used to extract images from the calibration and motion video files.

Once the calibration recordings are completed, the positions of the cameras should remain the same for the point tracking recording, otherwise the intrinsic and extrinsic parameters obtained lead to erroneous results. Most of the cameras have a limited capacity of recording a file (e.g. 10 minutes), whereas the Casio<sup>®</sup> Ex-Z750 has the advantage of recording a movie as long as the memory card permits. For this reason, among the commercial cameras, Casio<sup>®</sup> Ex-Z750 is chosen. This yields a non-stop recording duration of nearly 1 hour, which is enough for both the calibration and motion recordings. Another important fact is that, the focus of the camera changes when the auto focus mode is on. This causes changes in the internal distances of the camera. Therefore, the focus mode of the cameras should be chosen as infinity during recording, as stated by Forsyth and Ponce (2003).

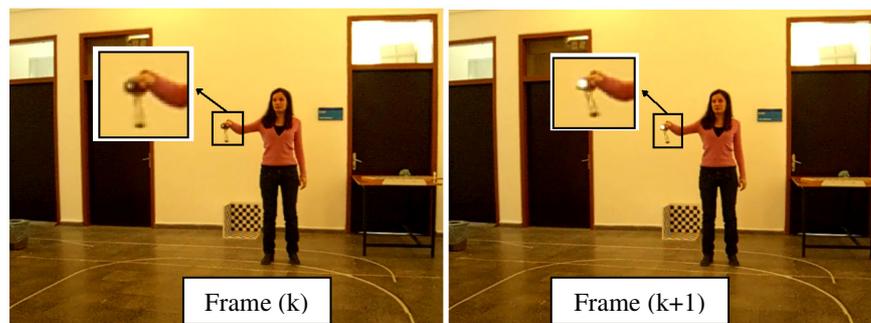


Figure 3.9 The modes of a flashlight between simultaneous images

Unfortunately, remote controls, which would easily accomplish synchronized recording, are not provided with the cameras. For this reason, a flashlight is turned

on and off during recording. Having extracted the movie files to its frames, off and on modes between the two simultaneous extracted frames can easily be recognized (see Figure 3.9). This, in fact, leads to a synchronization procedure which is described next.

If one records the scene starting from the left camera, continuing with the middle and right cameras; the synchronization between the frame numbers of the different cameras are given by the following relations:

$$F_L(k) \Leftrightarrow F_M(k + \Delta F_{LM}) \quad (3.11)$$

$$F_M(k) \Leftrightarrow F_R(k + \Delta F_{MR}) \quad (3.12)$$

$$F_L(k) \Leftrightarrow F_R(k + \Delta F_{LR}) \quad (3.13)$$

where

$$\Delta F_{LM} > 0, \Delta F_{MR} > 0, \Delta F_{LR} > 0 \quad (3.14)$$

Here,  $F_L(k)$ ,  $F_M(k)$ ,  $F_R(k)$  refers to the  $k$ 'th image in the left, middle and right cameras, respectively.  $\Delta F_{LM}$  is the frame difference between left and middle cameras,  $\Delta F_{MR}$  stands for the frame difference between middle and right cameras, and  $\Delta F_{LR}$  represents the frame difference between left and right cameras.

In order to apply the camera calibration procedure presented by Bouguet (2008), a planar object covered by a checkerboard pattern is necessary. The dimensions of the grids should be decided such that each square can be identified by the program. Moreover, the dimensions of the calibration object should be chosen to satisfy the condition that the calibration volume covers the points to be tracked. Images of the

calibration object in different orientations should then be obtained by each camera of the setup. Zhang (2000) emphasizes two significant points at this stage. One of the points is that, images of the calibration object should be taken in positions that are not parallel. In other words, the orientation of the object should be different for each image since the information obtained from parallel images does not bring any additional constraints. The second point is that as the number of the images is increased, the accuracy of the results is also increased. In the calibration process of the toolbox of Bouguet (2008), image coordinates of the grid corners are labeled manually (which constitutes a model for mapping the image area to the volume of interest).

After performing the calibration of each camera, the intrinsic parameters of each camera are obtained. In the next step, the stereo calibration toolbox is used for the stereo calibration of each pair of cameras in order to reach the extrinsic parameters (associated with the position of each camera with respect to each other). The three cameras can be considered to be composed of three camera pairs (Figure 3.10).

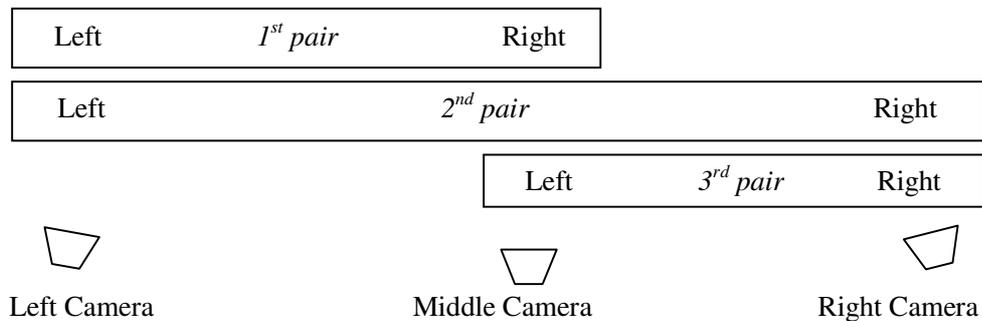


Figure 3.10 Local labeling of three camera pairs

Firstly, the left and middle cameras are calibrated to reach the rotation matrix and translation vector between this pair. Secondly, the left and right cameras are

calibrated and the rotation matrix and translation vector between this pair are found. Finally, the middle and right cameras are calibrated to acquire the rotation matrix and translation vector between this pair. In all three stereo calibrations, cameras are labeled locally as left and right. The user's manual and the flow charts of the codes appended to the thesis give detailed explanations about running the codes, for the whole procedure including calibration (see Appendix A & Appendix B).

At the end of the stereo calibrations, the three rotation matrices and the three translation vectors are generated and they are abbreviated as;  $\mathbf{R}_{ML}$ ,  $\mathbf{R}_{RL}$ ,  $\mathbf{R}_{RM}$ ,  $\mathbf{t}_{ML}$ ,  $\mathbf{t}_{RL}$  and  $\mathbf{t}_{RM}$ .

### **3.6 Production of the Calibration Rig Mechanism**

In usual applications, the calibration volume is quite small such that the calibration object fully covers the volume of interest. However, in the case of horse saddle motion analysis, the motion is performed in a much larger volume than in usual cases. Furthermore, in order to cover the whole necessary motion volume, the set up should be located relatively further from the scene. Particularly, at least two periods of the paces of horses is required to be recorded. Therefore the cameras should be located at 6-7 meters far from the scene which is to be 6-7 meters wide horizontally. The vertical distance to be calibrated is calculated as 0.8 meters. Consequently, the calibration object is required to be prepared to satisfy aforementioned conditions.

In the decision stage regarding the pattern dimensions, checkerboard patterns with various dimensions are drawn by means of AUTOCAD<sup>®</sup>. Patterns are printed and recorded in different locations. Dimensions of the square grids used are 35x35 mm, 40x40 mm, 50x50 mm, 55x55 mm and 60x60 mm (Figure 3.11). The obtained images are then used in the calibration toolbox. Squares of 35, 40 and 45 mm sizes are not identified clearly due to the resolution of cameras. If cameras having higher

resolution were used, those grids could be identified easily. The smallest square size that can be identified is 50 mm. Since smaller grids result in a more detailed mapping, dimensions greater than 50 mm are not preferred. Therefore the square dimension of the calibration pattern is decided to be 50 mm.

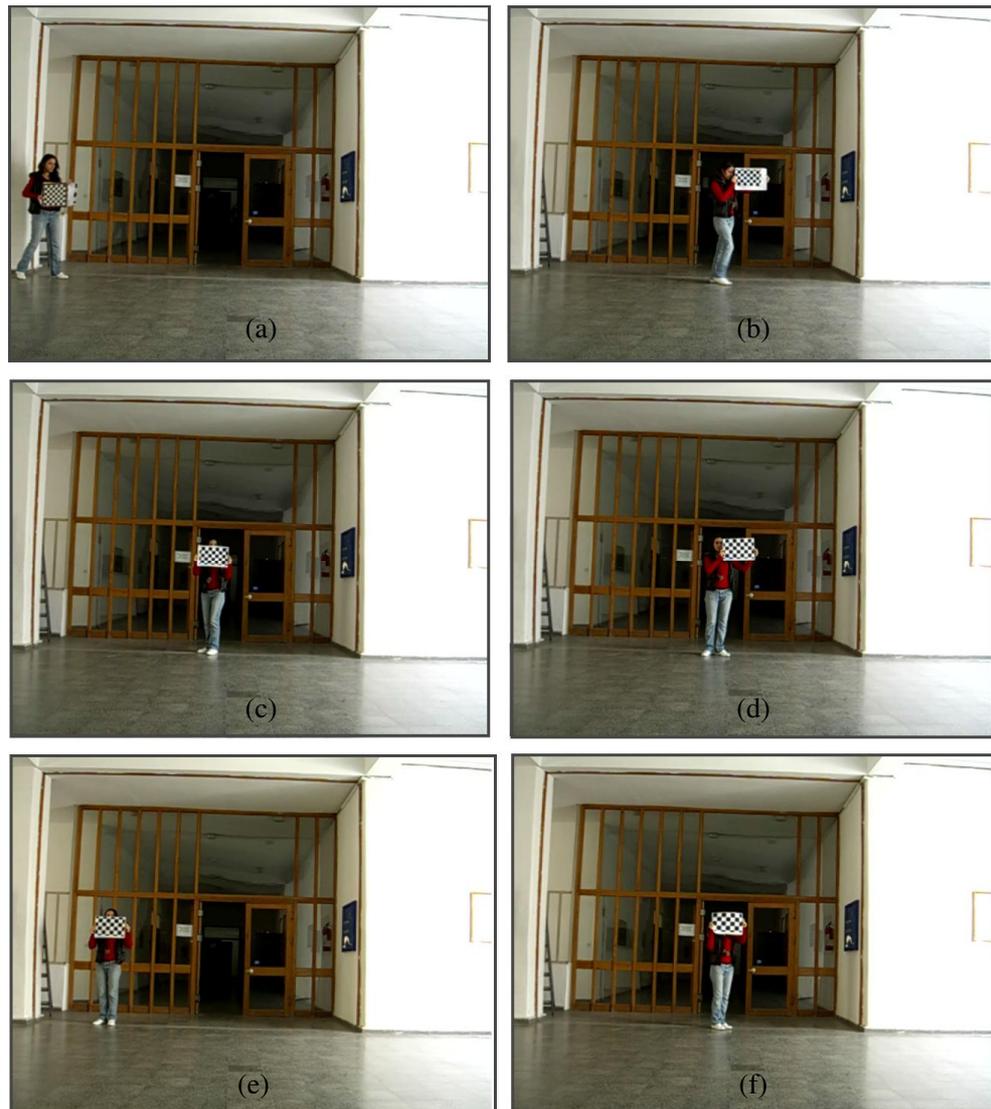


Figure 3.11 Square grid dimensions

(a) 35x35 mm (b) 40x40 mm (c) 45x45 mm (d) 50x50 mm (e) 55x55 mm (f) 60x60 mm

The mechanism that aims to transport the calibration object is designed and manufactured such that both translations and rotations can be accomplished. As seen in Figure 3.12, the user can translate the mechanism in two directions ( $x_{\text{mech}}$  and  $z_{\text{mech}}$ ) due to the special wheels chosen, which can rotate around vertical and horizontal axes. This constitutes the lower part of the mechanism.

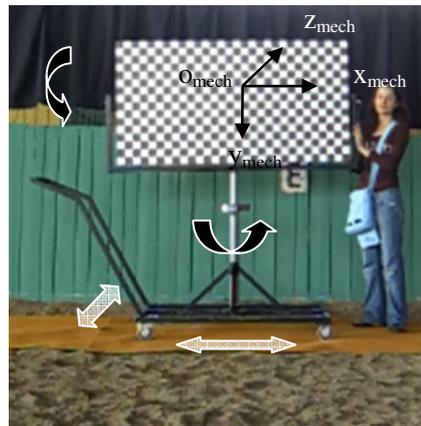


Figure 3.12 An image of the calibration object with the mechanism showing the degrees of freedom

The upper part of the mechanism is designed such that, the user may rotate the calibration object along two axes, namely,  $x_{\text{mech}}$  and  $y_{\text{mech}}$ . The technical drawings of the mechanism are presented in Appendix C.

## CHAPTER 4

### OPERATIONS ON THE DIGITAL IMAGE PLANE

#### 4.1 Definition of Digital Image

Digital image is a function having the domain of spatial (plane) coordinates and the range of intensity (color value/ gray level). In other words, an image is a finite and two dimensional matrix array whose elements, namely pixels, correspond to the color values.

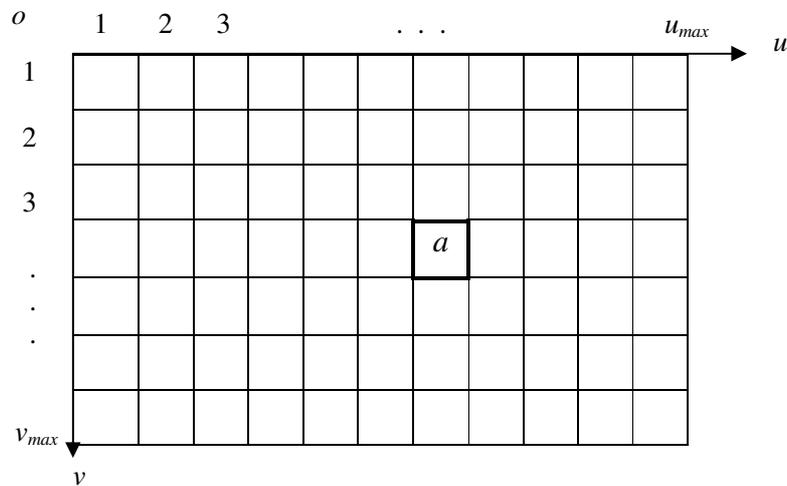


Figure 4.1 Image reference frame showing pixel representation

A digital image of  $u_{max} \times v_{max}$  can be represented by a reference frame, composed of the image pixel coordinates, e.g. a pixel located at  $a$  has the pixel coordinates of

$(a_u, a_v)$ . (refer to Figure 4.1). The color value at this point is expressed as  $I(a_u, a_v)$ . Alternatively, a digital image can be represented as a matrix array, such that, when read in a software program, it is expected be converted to a matrix array composed of rows and columns, with size of  $v_{\max} \times u_{\max}$  (as seen in Equation 4.1).

$$\mathbf{I} = \begin{bmatrix} I_{1,1} & I_{1,2} & \cdots & \cdots & I_{1,u_{\max}} \\ I_{2,1} & I_{2,2} & \cdots & \cdots & I_{2,u_{\max}} \\ \vdots & \vdots & \ddots & \boxed{I_{a_v, a_u}} & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ I_{v_{\max},1} & I_{v_{\max},2} & \cdots & \cdots & I_{v_{\max},u_{\max}} \end{bmatrix}_{v_{\max} \times u_{\max}} \quad (4.1)$$

For the same point on the image,  $I_{a_v, a_u}$  corresponds to the color value with the indices of  $(a_v, a_u)$ .

The purpose of using digital cameras in vision systems is that the requirement of sampling (digitizing the coordinate values) and quantization (digitizing the amplitude values) of the continuous sensed data is executed by the cameras internally. Therefore, the need for these operations is eliminated with the use of digital cameras, which brings an advantage to the user.

## 4.2 Image Types

The colors of an object, perceived by humans and animals, are determined by the nature of light reflected from the object (Gonzalez & Woods, 2002). Light is called achromatic, if it contains color, or monochromatic, if it is void of color. The only characteristics of this type of light are the intensity levels. On the other hand, chromatic light spans the electromagnetic spectrum from approximately  $0.43 \mu\text{m}$  to  $0.79 \mu\text{m}$  in wavelength, which corresponds to the visible band in the EM spectrum.

Images can be described in terms of the color or intensity levels of the pixels that compose the image.

### 4.2.1 Binary Image

One of the achromatic image representations is the binary image type. In the binary image, each pixel takes a value of one of only the two discrete values, 0 or 1. Pixels with the value 0 are displayed as black; whereas white pixels are represented by 1 (see Figure 4.2). Binary images, when read in a software program, are usually considered as logical arrays.

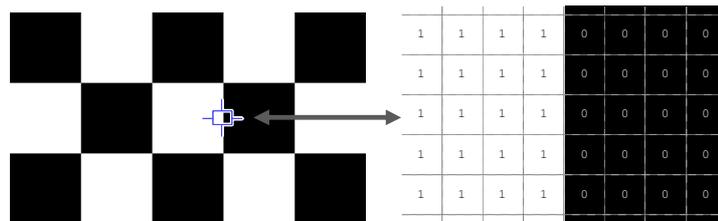


Figure 4.2 Values of pixels in a binary image

### 4.2.2 Grayscale Image

The other achromatic image representation is the grayscale (or gray level) image type. In the grayscale image, the intensity ranges from black to gray, and gray to white. Similar to the binary image, black pixels correspond the integer value of 0, and white pixels correspond the integer value of 1 in the grayscale image. Gray levels stand for the change in the intensity (see Figure 4.3). The number of the gray levels,  $L$ , is usually a power of 2, where  $\kappa$  is the number of bits to store one pixel (see Equation 4.2).

$$L = 2^{\kappa} \quad (4.2)$$

Therefore, the range of the gray levels is  $[0, L-1]$ . Mostly,  $\kappa$  is 8 bits, constituting a gray level range of  $[0, 255]$ .

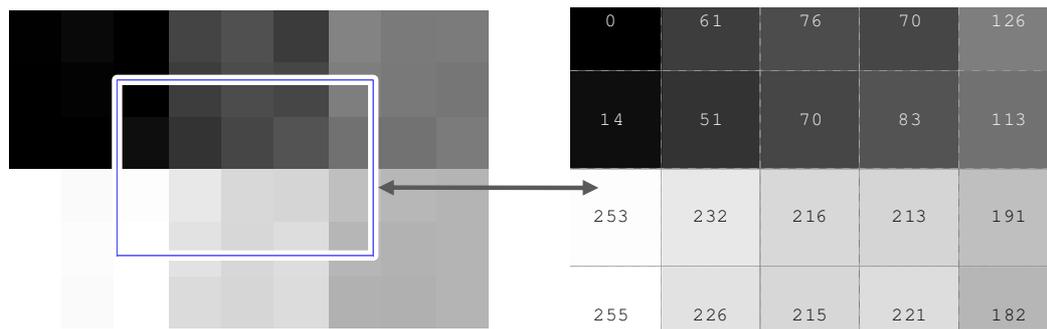


Figure 4.3 Values of pixels in a grayscale image

### 4.2.3 Full-Color Image

A chromatic type of image is the full-color image. In the full-color image, each pixel is specified by three values. That is, the image is stored as a matrix of size  $v_{\max} \times u_{\max} \times 3$  when represented as an array. The color models are standardized such that; if each of the three components defines the primary colors, the image is to be represented as RGB color model. Red, green and blue constitutes the primary colors due to the absorption characteristics in the eye. Magenta (red plus blue), cyan (green plus blue), yellow (red plus green) are named as the secondary colors of the RGB model. For convenience, the values of colors may be normalized in the range  $[0,1]$  (Figure 4.4). The color of each pixel is determined by the combination of the red, green, and blue intensities stored in each color plane at the pixel's location in a software program (Mathworks, Inc., 2006).

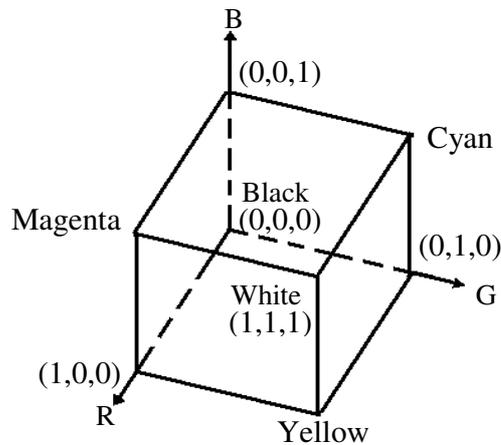


Figure 4.4 RGB color cube

The separated monochrome planes of a color image are shown in the Figure 4.5. The white region in each plane corresponds to the highest value of the separated color.

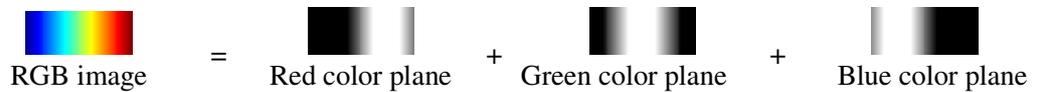


Figure 4.5 Separated color planes of an image

Another color model is CMY where unlike the RBG model, the primary colors are cyan, magenta and yellow and red, green and blue are the secondary colors. RGB or CMY color models are suitable to describe colors practically. Rather, HSI (or HSV) color model is convenient if color values are to be represented by its hue, saturation, and intensity (value) components. Hue is the component describing the pure color characteristics varying from 0 to 360 degrees; starting from red, through yellow, green, cyan, blue, magenta, and back to red. Saturation, varying between 0 and 1, can be thought of the purity of a color. Finally, value component shows the brightness. Varying from 0 to 1, the corresponding colors become increasingly brighter (see

Figure 4.6). The conversions between the color models (RGB, CMY, HSI) can be found in (Gonzalez & Woods, 2002).

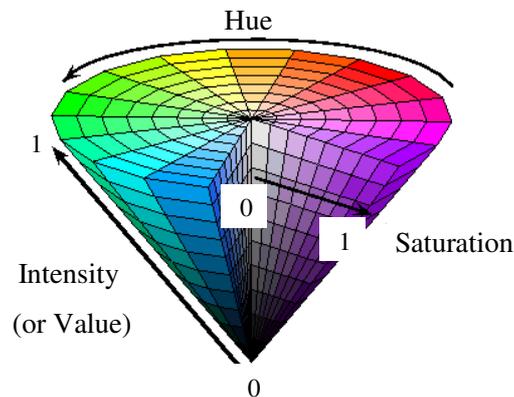


Figure 4.6 HSI (or HSV) color space

### 4.3 Conversion between Image Types

Images can be transformed from one type to another for various reasons. For instance, extracting an object from an image can be realized by means of image type conversion. Image Processing Toolbox of MATLAB<sup>®</sup> introduces various functions for this purpose. A general transformation can be expressed as in Equation (4.3) where  $I(u, v)$  stands for the original image to be transformed,  $T$  is a transformation function and  $H(u, v)$  is the transformed image.

$$H(u, v) = T[I(u, v)] \quad (4.3)$$

#### 4.3.1 Conversion to Grayscale from RGB or HSI Image

As mentioned previously, if an image is represented as RGB or HSI, each pixel is composed of 3 values. If the image is required to be represented as a grayscale

image, each pixel is to have one component, the gray value. Particularly, if the image is in HSI form, the hue and saturation values are to be eliminated whereas, the intensity is retained. On the other hand, if the image is in RGB form, then the following transformation equation is valid:

$$\text{Gray} = 0.2989 \cdot R + 0.5870 \cdot G + 0.1140 \cdot B \quad (4.4)$$

#### **4.3.2 Conversion to Binary Image from RGB or Grayscale**

A binary version of an image can be created by a thresholding function. If the image to be converted is of color type, it should firstly be converted to a grayscale image. As mentioned under the previous heading, image thresholding is of great importance for object identification purposes. A global thresholding function can be defined as below where the threshold value,  $l$ , defines the level of thresholding:

$$H(u,v) = \begin{cases} 1 & \text{if } I(u,v) > l \\ 0 & \text{if } I(u,v) \leq l \end{cases} \quad (4.5)$$

In this study, RGB to binary image conversion is applied at the first step of the passive marker identification, where the markers are attached to the horses and saddles.

#### **4.4 Selection, Characteristics and Location of Markers**

The essential point in the selection of a passive marker is the visibility of the marker during motion. The marker should also be easily isolated from other objects and the background. At the outset, color information from markers of various colors was decided to be used. That is, circular markers with diameters of 30-40 mm having various colors, such as red, green or blue, were tracked. This was realized by means

of a color value filtering algorithm. If the distance between the camera and the scene varies between 1-4 meters, the color of the markers is clearly identified by the camera. On the other hand, in the applications where the distance between the camera and the scene is larger than 4-5 meters, as in horse motion analysis, the color of the markers cannot be identified throughout the whole motion. Therefore, the use of color markers is not convenient, if the camera is located far from the scene of interest.

Another alternative marker is the spherical type markers available in the METU Gait Analysis Laboratory. These are wooden markers which are of 25 mm diameter and which are covered by 3M<sup>®</sup> retroreflective material of gray-silver color. However, the size and shape of these markers are impractical for this study. The retroreflective material covering these spheres however is convenient, since it directly sticks to the object and there is no distance between the marker center and the object. The retroreflective material is available as a tape of cloth, which is then cut in circular shapes with a diameter of 60 mm. In order to get full advantage of the reflective characteristic of the markers, a light source of 1000W is directed to the markers during the motion. When a light source of such a high power is directed to the markers, the silver-gray color is obtained as white in the images (see Figure 4.7). The markers are taped on body of the horse and the saddle at appropriate locations.

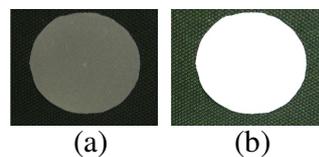


Figure 4.7 A marker (a) without light (b) with light

As mentioned before, the saddle motion of horses in different gaits will be investigated. However, when the rider gets on the horse, the saddle is occupied by

the lower part of his body. Hence, the markers are located on the skin of horses, as close as possible to saddle. Horses are sensitive to touch, so the markers should be located on an insensitive part of the horse. As seen in Figure 4.8, the least sensitive portion corresponds to the dotted region which is a convenient choice for the markers to be taped.

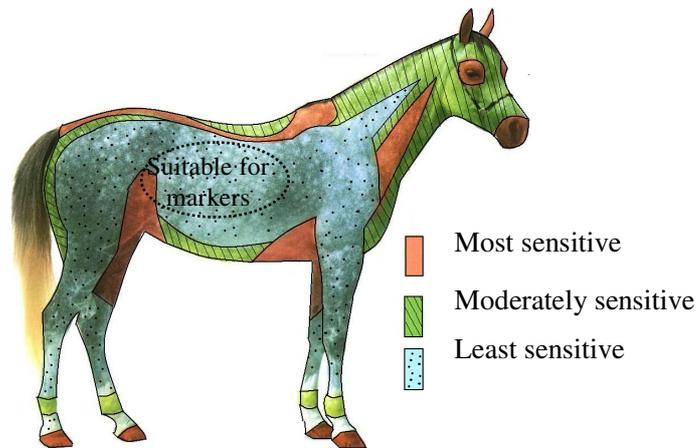


Figure 4.8 Sensitivity regions of horses (Adapted from Micklem, 2003)

#### 4.5 Marker Detection

Following the calibration process, the motion recorded by the three cameras should be transferred to the computer. The images are extracted from the movie files by the AVD Video Processor<sup>®</sup>, similar to the calibration image selection. Since there are 3 cameras, the images are labeled as left, middle and right followed by the frame number. Using the flashlight information, the frames are synchronized and prepared for the purpose of reaching the passive marker image coordinates. The image coordinates of the markers are obtained by means of filtering algorithms. At the end of the filtering operations, only the white pixels belonging to the markers (with the

center information) remain. The procedure is executed for each of the cameras, independently.

#### **4.5.1 Filtering via Thresholding**

In this study, the image files are obtained in the RGB format. The retroreflective characteristics of the markers are realized only if the images are converted to black and white. The sequence of images is firstly converted to the grayscale format, then to the binary. As expected, the thresholding function is utilized with the threshold value, in the range of [0,255] being introduced as input by the user. Since the markers are expected to be white, the threshold value(s) should be close to the upper value of the range. If the threshold value holds for all images of the motion (e.g. from the beginning to the end of a trotting of horse video), the threshold level is kept constant. However, due to the light conditions of the scene, one may enter varying value for the threshold level. Visibility of all markers and isolation from all other objects and the background should be considered, in all images. Indeed, there may appear other regions satisfying the threshold condition. Therefore, additional filtering methods are necessary.

#### **4.5.2 Labeling Regions**

After converting the original images to binary ones, the white regions are labeled so that they are distinguished from each other. In this process, the black pixels remain unchanged as 0, while the white pixels are labeled by means of either 4 or 8-connectivity. In this study, 8-connectivity is preferred to avoid bypassing the markers. The difference of 4 and 8 connected pixels are illustrated in Figure 4.9. For details of labeling, refer to Mathworks, Inc., 2006. For convenience, the markers are located horizontally. The markers remain in the same sequence during the motion of the horse, unless large rotations occur, which may disorder the sequence of markers. The centers of regions are found by computing the geometrical center for the images.

At the end of filtering methods, the labeling of markers on the body of the horses has to be obtained in a similar manner as shown in Figure 4.10. (The number of tracked markers equals to 3 in the horse application.)

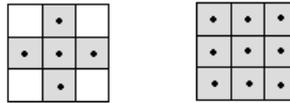


Figure 4.9 4-connected and 8-connected pixels



Figure 4.10 Final marker labels to be reached at the end of all filterings (a close look)

In the following steps, properties of the first image of the motion are used as the input. In the first image, labels of markers to be extracted should be entered, from which the overall geometric center of markers can be reached. This input is a necessity for the windowing operation.

### 4.5.3 Filtering via Area

In this step, a function that filters regions having areas out of a defined range is implemented. After reaching to approximate information on the area of all regions in all images, a range in the form of  $[area_{min}, area_{max}]$  is provided (while entering the inputs of the first image) for the areas of the allowable regions. Therefore, in all

images of the motion, the pixels will be converted to black (eliminated), if the area of the region it belongs to is out of the desired interval.

#### 4.5.4 Windowing

In order to perform filtering operations for all images in a certain region, rather than the whole image, a windowing operation is introduced. Again using the first image, a window size for the region of interest is defined.  $W_c^e(k)$  denotes the overall geometric center of markers 1, 2 and 3, in frame k. The height and width of the rectangular window ( $W_{du}$ ,  $W_{dv}$ ) are decided such that the window includes all markers to be tracked (Figure 4.11). The window center of the (k+1)'th image,  $W_c(k+1)$ , is taken at the overall marker image center of the k'th image,  $W_c^e(k)$  (see Figure 4.11).  $W_c^e(k+1)$  corresponds to the overall geometric center of markers 1, 2 and 3 in frame k+1. For the consequent images, the window slides as the overall marker center moves, i.e. the window is a dynamic one, rather than static.

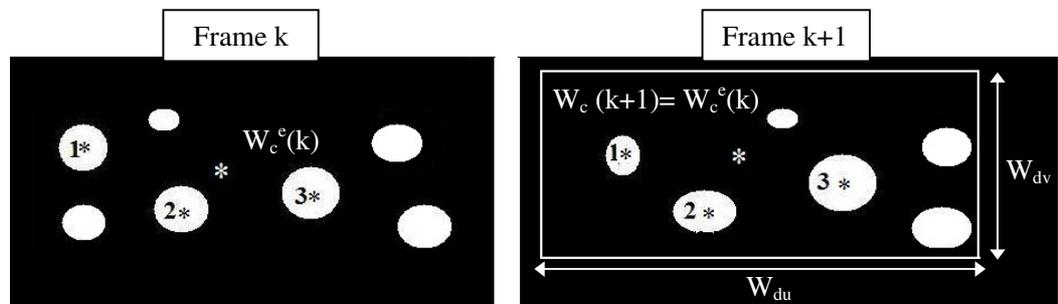


Figure 4.11 Windowing operation

#### 4.5.5 Filtering via Distance

There may still remain regions that make noise, i.e. act like a pseudo marker, which cannot be filtered by the previous operations namely, thresholding and area filtering,

in the windowed area. Using the distance filter, one can get rid of such regions (Figure 4.12). In the first image, the image centers of the markers, i.e.  $(u_i, v_i)$  for  $i=1,2,\dots,M$  may be found. Here  $(u_i, v_i)$  denotes the coordinates of the center of the  $i$ 'th marker and  $M$  denotes the number of markers. Hence all possible distances between the markers can be obtained via the equation;

$$DM_{ij} = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} \Big|_M \quad (4.6)$$

where  $DM_{ij}$  denotes Euclidean distance between marker  $i$  and  $j$ . Note that the number of such distances is given by:

$$D_M = [M(M-1)]/2 \quad (4.7)$$

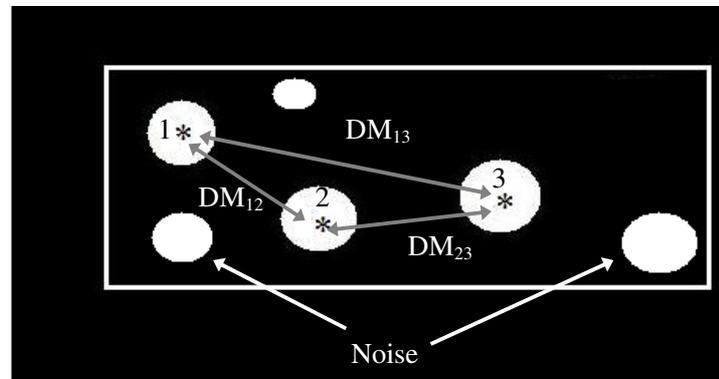


Figure 4.12 A sample that needs the distance filtering operation

Let now the total number of regions, including the markers be denoted as  $C$ . Then, the total number of Euclidean distances that can be measured between the centers of all regions will be given by:

$$D_c = [C(C-1)]/2 \quad (4.8)$$

where the distance between the i'th and j'th regions is given by:

$$DC_{ij} = \sqrt{(u_i - u_j)^2 + (v_i - v_j)^2} \Big|_c \quad (4.9)$$

In the distance filtering operation, among the possible combinations of distances between regions, required distance combination (CD) satisfying distance threshold ( $DT_{ij}$ ) (introduced as a percentage by user) will be accepted. The number of distance combinations and the required distance interval ( $DI_{ij}$ ) can be denoted as:

$$CD = \frac{(D_c)!}{(D_M)!(D_c - D_M)!} \quad (4.10)$$

$$DI_{ij} = [-DM_{ij} \cdot DT_{ij}, +DM_{ij} \cdot DT_{ij}] \quad (4.11)$$

At the end of the distance filtering operations, one may still have noisy results which include pseudo markers. Hence, a final filter, using the cross product operation, is implemented after the distance filter for reaching the image coordinates of the true markers. An example illustrating the cross product filter is given in the next paragraph.

As shown in Figure 4.13, there exist two choices for the second marker, labeled as marker 2 and marker 2'. Both markers satisfy the conditions used in the area and distance filters. The correct position of the second marker is marker 2 if and only if  $\text{sgn}[(\mathbf{r}_{12} \times \mathbf{r}_{23})_3]$  in the considered frame is equal to  $\text{sgn}[(\mathbf{r}_{12} \times \mathbf{r}_{23})_3]$  in the first frame (which is known). Otherwise, the correct position of the marker is marker 2'. Here,  $\text{sgn}$  denotes the sign operator,  $\mathbf{r}_{ij}$  denotes the vector from marker i to marker j and  $(..)_3$  denotes the third component of the vector inside the parenthesis.

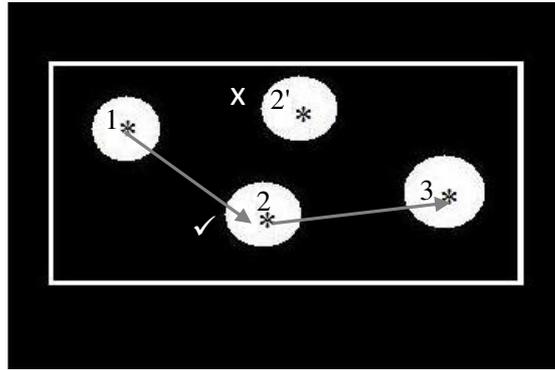


Figure 4.13 Illustrative example for cross product filtering

At the end of the operations introduced, a 3D matrix array,  $\mathbf{PM}$ , including coordinates of the image centers of all markers in each frame is obtained. The dimension of  $\mathbf{PM}$  is  $M \times 2 \times K$  where  $M$  and  $K$  represent the number of markers to be tracked and number of images, respectively (Figure 4.14).

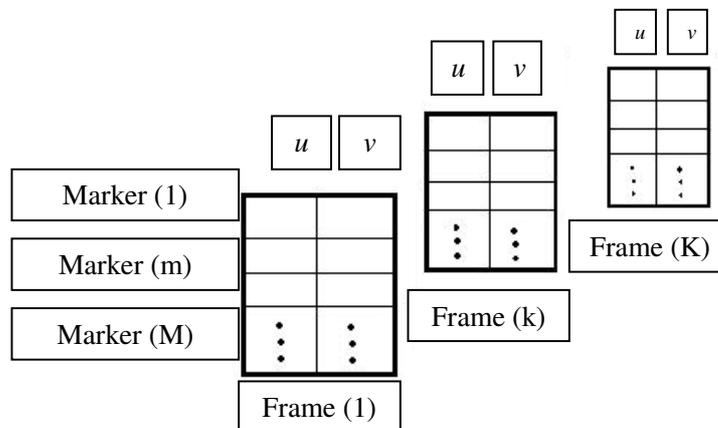


Figure 4.14 The 3D matrix array for the image coordinates found in each camera

Since the operations are performed for the 3 cameras of the set up, the resultant matrices are denoted as:  $\mathbf{PM}_L$ ,  $\mathbf{PM}_M$  and  $\mathbf{PM}_R$  for the left, middle and right cameras,

respectively. A sample series of operations on one image are shown in Figure 4.15. Note that one may track the operations by using the titles of the graphs.

- i) The original RGB image.
- ii) The image after RGB to BW thresholding.
- iii) Labeled image (The center of each white region is labeled).
- iv) Area filtered image. (Only the areas in the specified area interval have been kept.)
- v) The regions in the pre-defined window.
- vi) The (desired) markers and their centers, after applying the distance and cross product filters.

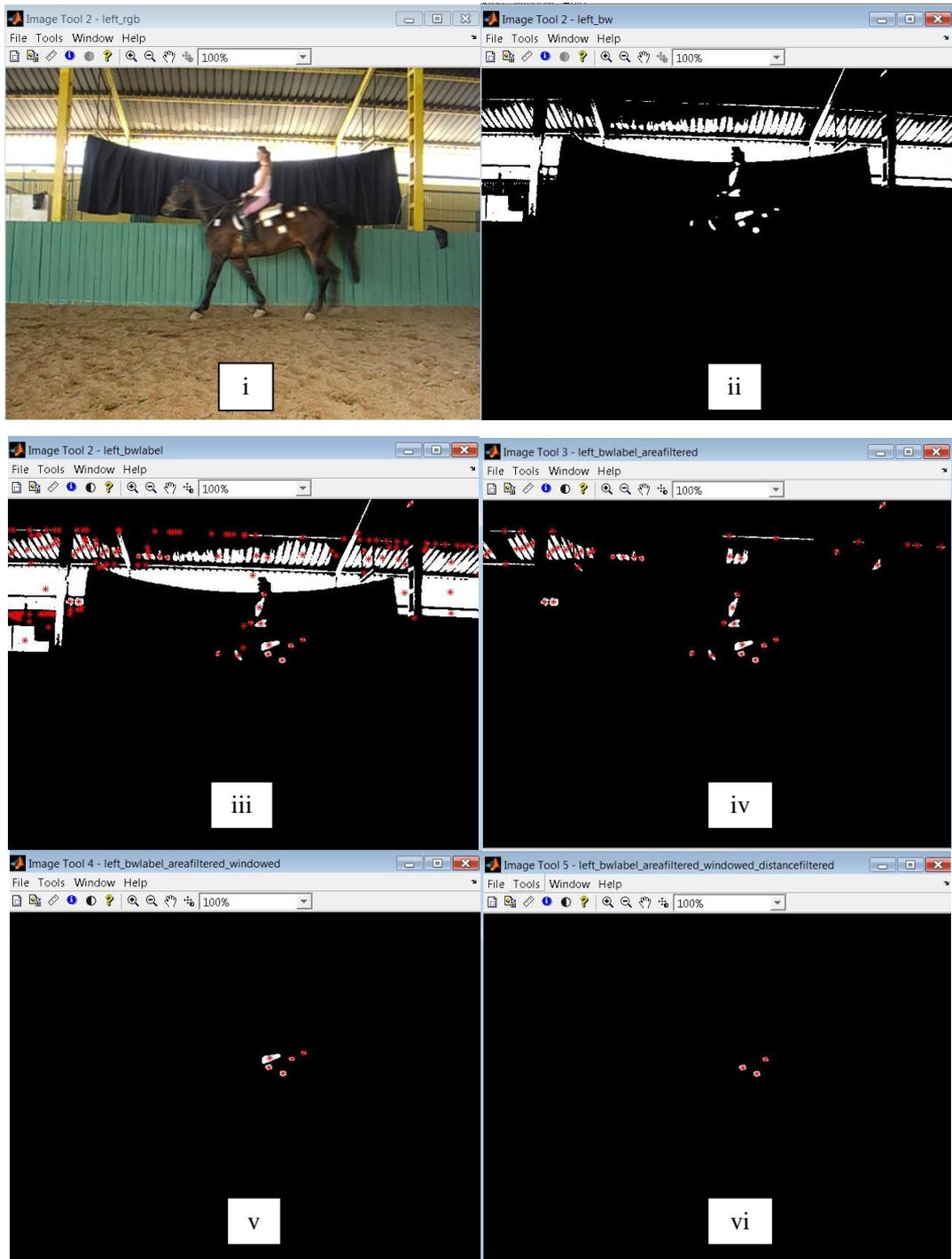


Figure 4.15 Sample stages of the image operations

## CHAPTER 5

### OPERATIONS IN THE 3D SPACE

Having reached the centers of the visible markers in the left, right and middle camera image reference frames, the next step is acquiring the coordinates of the markers in the fixed world reference frame. The topics introduced in this chapter aim to give the details of the techniques in sequence with the algorithms developed.

#### 5.1 Conversion from Image to World Coordinates

In the MATLAB<sup>®</sup> toolbox of Bouguet (2008), a function that computes the 3D position information of points is provided. As long as the image coordinates (obtained locally from the left and right camera views) and the camera parameters are provided, the coordinates of the position of points may be obtained in the reference frames of both cameras. This procedure is known as stereo triangulation (Bouguet, 2008).

As mentioned in Chapter 3, the camera setup has 3 stereo pairs. Therefore, using the stereo calibration results of the global left and middle cameras (locally left and right), the world coordinates of the marker centers can be obtained in the global left (or leftmost) camera. Additionally, with stereo calibration results of global left and right cameras (locally left and right), the world coordinates of the same marker centers can be obtained in the leftmost camera. On the other hand, using the stereo calibration results of the globally middle and right (locally left and right) cameras, the world coordinates of the marker centers can be obtained in the middle camera. Since the

transformation matrices associated with each pair of cameras have been obtained before, the coordinates represented in middle camera can be transformed to the leftmost camera. A 3D matrix array,  $\mathbf{CM}$ , containing the coordinates of the centers of the markers in each camera frame is defined (Figure 5.1).

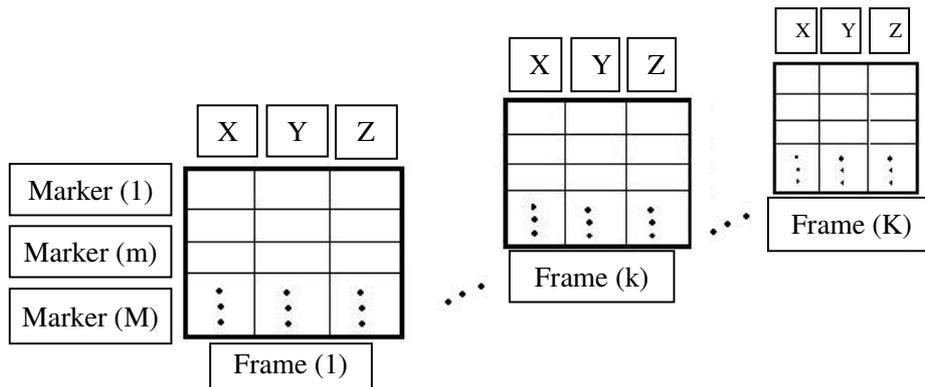


Figure 5.1 The 3D matrix array for the camera coordinates

$\mathbf{CM}(m,1,k)$  : X component of marker (m), in frame (k).

$\mathbf{CM}(m,2,k)$  : Y component of marker (m), in frame (k).

$\mathbf{CM}(m,3,k)$  : Z component of marker (m), in frame (k).

where the dimensions of this matrix is  $M \times 3 \times K$ .

For the 3 pairs of cameras, resultant 3D matrices are denoted as:  $\mathbf{CM}_{LR,L}$ ,  $\mathbf{CM}_{LM,L}$  and  $\mathbf{CM}_{MR,M}$  where  $\mathbf{CM}_{LR,L}$  is the matrix with coordinates in left camera frame (left and right cameras calibrated),  $\mathbf{CM}_{LM,L}$  is the matrix with coordinates in left camera frame (left and middle cameras calibrated) and  $\mathbf{CM}_{MR,M}$  is the matrix with coordinates in middle camera frame (middle and right cameras calibrated).

The important point here is that the 3 coordinates of the markers should be obtained in the same reference frame.  $\mathbf{CM}_{LR,L}$ ,  $\mathbf{CM}_{LM,L}$  are both represented in the left camera

reference frame. However,  $\mathbf{CM}_{MR,M}$  is obtained in the middle camera reference frame. Hence it is necessary to convert the coordinates from the middle to the left camera frame. The transformation of coordinates from the (locally) left camera frame to the right camera frame may be presented [see Bouguet (2008)] by the equation:

$$\mathbf{x}_R = \mathbf{R}_{RL}\mathbf{x}_L + \mathbf{t}_{RL} \quad (5.1)$$

where  $\mathbf{x}_R$  and  $\mathbf{x}_L$  denote the coordinates of a point in the right and left cameras, respectively.  $\mathbf{R}_{RL}, \mathbf{t}_{RL}$  are the rotation matrix and the translation vector to represent the left camera coordinates in the right. In the set up, the local right camera corresponds to the global middle camera, while the local left camera corresponds to the global left. Therefore, with an appropriate change of notation, Equation (5.1) becomes

$$\mathbf{x}_M = \mathbf{R}_{ML}\mathbf{x}_L + \mathbf{t}_{ML} \quad (5.2)$$

which leads to

$$\mathbf{x}_L = \mathbf{R}_{ML}^{-1}(\mathbf{x}_M - \mathbf{t}_{ML}) \quad (5.3)$$

Continuing with the previous notation, for all image frames and all markers, the coordinates in the middle camera reference frame can be transformed into coordinates in the left camera frame with the syntax below:

$$\begin{aligned} &\text{for } k = 1 : K \\ &\quad \text{for } m = 1 : M \\ &\quad \quad \mathbf{CM}_{MR,L}(m,:,k) = \mathbf{R}_{ML}^{-1}(\mathbf{CM}_{MR,M}^T(m,:,k) - \mathbf{t}_{ML}) \\ &\quad \text{end, end} \end{aligned} \quad (5.4)$$

Hence, for a given point, one obtains 3 different position vectors (from the 3 stereo pairs) in the leftmost camera reference frame. If there were no errors (due to calibration, resolution, etc), these 3 position vectors should be identical. In practice, however, the position vectors are pretty close to each other, but not the same. Therefore, the average of these 3 position vectors found via the syntax:

```

for k = 1 : K
    for m = 1 : M
        
$$\mathbf{CM}_L(i,:,k) = \frac{\mathbf{CM}_{LM,L}(i,:,k) + \mathbf{CM}_{LR,L}(i,:,k) + \mathbf{CM}_{MR,L}(i,:,k)}{3} \quad (5.5)$$

    end
end
end

```

is taken to be the correct position vector of the point.

## 5.2 Preprocessing in the Position Data

In general, if there exist individual data points (called outliers) which are inconsistent with the general nature of the rest of the data set, they should be identified. This is because outliers adversely affect the curve fitting (which is to be applied in the following steps) results since the sum of squares of the residuals is minimized in the fitting processes (Mathworks, Inc., 2002). Therefore, it is a good idea to preprocess the position vectors of the markers, since the cameras in the setup may introduce outliers which may occur due to the measurement errors.

### 5.2.1 Excluding the Data

One of the preprocessing techniques is to exclude the outliers from the data. After observing the general behavior of the dataset, data points that are inconsistent are excluded with this technique. The points, or sections, have to be marked one by one,

or an exclusion rule should be constructed to perform this operation. Due to practical reasons and eventual loss of data, in this thesis work, data exclusion from the position data is not preferred.

### **5.2.2 Smoothing the Data**

The second technique of preprocessing of data is referred as smoothing. Noise in the data can be reduced by various smoothing techniques. Different smoothing types may be required for different purpose. Smoothing includes neighboring operations. There is no loss of data unlike excluding. Some popular types of smoothing are moving average filtering, “lowess” (locally weighted scattering using first-degree polynomial) and loess (locally weighted scatter plot smooth using a second-degree polynomial). Robust versions of “lowess” and “loess” (“rloess” and “rloess”) may also be used. In the tests of smoothing, moving average filtering was seen to be unsuitable for the purposes of this study, since the end points of the data are not included in moving average filtering and this kind of filtering is exposed to noise (or outliers). However, robust types of lowess and loess are advantageous such that they are resistant to outliers. That is, robust smoothing automatically excludes the outliers. In the first plot of Figure 5.2, a set of data having two outliers is shown. Smoothing using lowess is clearly observed in the second plot which is affected by the outliers. In the third plot, the robustness of the smoothing can be shown with the resistance to the outliers. Hence, robust smoothing is preferred in this study to preprocess the position vectors of the markers. Each point’s camera coordinate components are smoothed by the ‘rloess’ type with a span of 10% as introduced in MATLAB<sup>®</sup>.

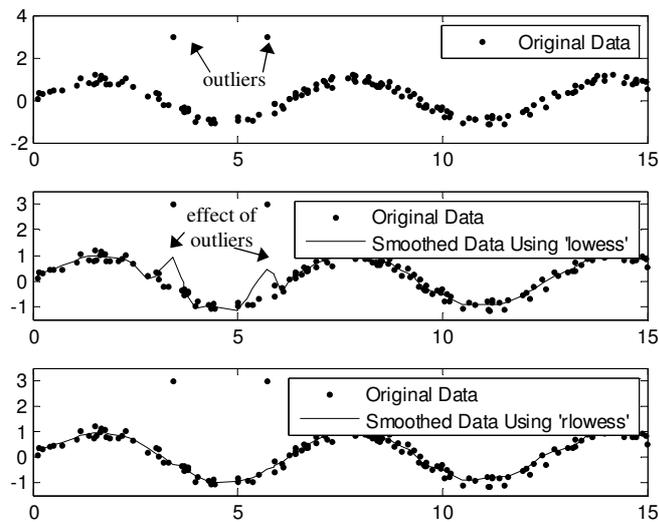


Figure 5.2 Smoothing a set of points

### 5.3 Conversion from the Camera to the Motion Reference Frame

Since examining the motion of the horse saddle in the camera reference frame is not convenient, the motion characteristics of the markers on the horse should be represented in a more suitable coordinate frame. Therefore, position vector of each marker will be expressed in the so called motion reference frame of one of the markers selected by the user. The three orthogonal axes of this so called motion coordinate frame are determined by using the following procedure.

Pre-experiments of this study have shown that saddle movements have different properties in each gait of the motion. However, as expected, a common property is that the motion is periodic in all directions other than the direction of motion of the horse. Therefore, one axis of the motion reference frame is selected to be coincident with a vector  $\mathbf{u}_1$  in the direction of the net displacement of one of the markers (to be chosen by the user) in one (or two) period(s). The unit vector  $\mathbf{u}_1$  is given by either one of the expressions:

$$\mathbf{u}_1 = \frac{\mathbf{x}(t = T) - \mathbf{x}(t = 0)}{|\mathbf{x}(t = T) - \mathbf{x}(t = 0)|} \quad (5.6)$$

$$\mathbf{u}_1 = \frac{\mathbf{x}(t = 2T) - \mathbf{x}(t = 0)}{|\mathbf{x}(t = 2T) - \mathbf{x}(t = 0)|} \quad (5.7)$$

where  $\mathbf{x}$  denotes the position vector of a marker and  $T$  corresponds to the period of motion.  $\mathbf{u}_1$  is given by Equation (5.6) if one period of motion is recorded. If two periods are recorded, on the other hand, Equation (5.7) should be used. The  $\mathbf{u}_1$  direction is called to be the x-motion direction.

The second unit vector of the motion reference frame will correspond to the gravity direction. In order to obtain the gravity direction, images of a plumb are taken while recording the motion of interest. The upper and lower parts of the plumb are marked by 2 reflective markers. Then, the 3D coordinates of the two end points of the plumb are obtained in the left camera reference frame. The markers on the upper and lower parts of the plumb correspond to the tail and tip of the gravity vector, respectively. The unit vector in the gravity direction is expected to be perpendicular to the pre-defined x-motion direction vector. However, in practice, the measured angle between the two vectors is very close to  $90^\circ$ , but not exactly 90 degrees. Therefore, a correction is required at this step. Let, now, the gravity vector obtained via the plumb be referred as the estimated gravity vector and let it be denoted by  $\mathbf{u}_2^e$ . The third unit vector of the motion reference frame,  $\mathbf{u}_3$ , is obtained using the cross product of the two vectors as seen in Equation (5.8).

$$\mathbf{u}_3 = \frac{\mathbf{u}_1 \times \mathbf{u}_2^e}{|\mathbf{u}_1 \times \mathbf{u}_2^e|} \quad (5.8)$$

Finally, having obtained the two orthogonal directions of the motion reference frame, the corrected gravity vector,  $\mathbf{u}_2$ , can be obtained as:

$$\mathbf{u}_2 = \mathbf{u}_3 \times \mathbf{u}_1 \quad (5.9)$$

Note that if  $\mathbf{u}_1$ ,  $\mathbf{u}_2$  and  $\mathbf{u}_3$  are expressed in the camera frame, the rotation matrix  $\mathbf{R}_{CM}$  that relates the camera and motion frames will be given by:

$$\mathbf{R}_{CM} = [\mathbf{u}_1 \mid \mathbf{u}_2 \mid \mathbf{u}_3] \quad (5.10)$$

such that;

$$\mathbf{x}_C = \mathbf{R}_{CM} \mathbf{x}_M \quad (5.11)$$

where  $\mathbf{x}_C$  and  $\mathbf{x}_M$  denotes the coordinates of a point in the camera and motion frames, respectively. Equation (5.11) yields:

$$\mathbf{x}_M = \mathbf{R}_{CM}^{-1} \mathbf{x}_C \quad (5.12)$$

The directions of  $\mathbf{u}_2$  and  $\mathbf{u}_3$  are labeled as the y-motion direction (or gravity) and the z-motion directions, respectively. At the end of the operations, each marker is represented in one of the marker's (decided by the user) motion coordinate frame.

Up to this point, the independent variable used in the motion analysis has been the frame number. From this point on, the independent variable is converted to time given by:

$$t = (k - 1)/\text{fps} \quad (5.13)$$

Here,  $t$  denotes time;  $k$  denotes the frame number such that at  $t=0$ ,  $k=1$  and  $\text{fps}$  denotes the recorded frames per second (which is a property of the camera used).

The components of each marker's position vector in the motion reference frame will be denoted by  $\mathbf{X}_M, \mathbf{Y}_M, \mathbf{Z}_M$  each of which has the dimension of  $M \times K$ . Similar to  $\mathbf{CM}$ ,  $\mathbf{MM}$  is the matrix that represents the 3D coordinates of all markers in the motion reference frame.

#### **5.4 Registration of 3D Shapes**

In this study, the objective is to obtain the motion of the saddle. Particularly, the motion of a single point on the saddle which is in contact with the rider is of interest. Unfortunately, this marker is not visible when the horse is being ridden, since the marker is blocked by the rider's body. To overcome these difficulties, three markers are attached on the skin of the horse. These markers are as close to the invisible marker as the constraints allow. Then the 3 markers on the skin and the invisible marker on the saddle are assumed to belong to the same rigid body, namely the saddle. In the following paragraphs, a registration algorithm, i.e., a correction algorithm satisfying the rigid body constraints between the 3 markers, is introduced.

Besl and McKay (1992) propose an algorithm based on quaternions for 3D object registration. The best alignment of a "measured" data set with a "model" data set can be assured with their approach (Figure 5.3). In this figure,  $e_1, \dots, e_m, \dots, e_M$  are the distances between the corresponding measured and model data sets, representing the errors.

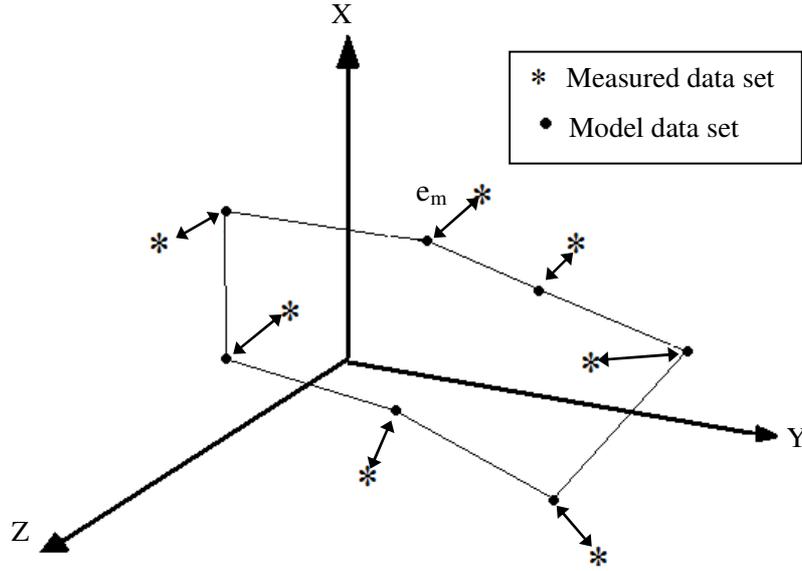


Figure 5.3 Model and measured point sets

In order to apply the procedure, a model data set is necessary. To this purpose, the 3D coordinates of the markers in a static position are used. The measured data set, on the other hand, corresponds to the markers positions during the motion. The algorithm is based on minimization of the error function:

$$f(\mathbf{q}) = \frac{1}{M} \sum_{m=1}^M \left\| \mathbf{x}_m^m - \mathbf{R}(\mathbf{q}_R) \mathbf{x}_m - \mathbf{q}_t \right\|^2 \quad (5.14)$$

Here,  $\mathbf{x}_m^m$  and  $\mathbf{x}_m$  are the model and the measured position data (of marker  $m$ ) sets, respectively.  $\mathbf{R}$  is a rotation matrix relating the model and the measured data sets. The optimal value of  $\mathbf{R}$ ,  $\mathbf{R}^o$ , is to be found at the end of the optimization.  $\mathbf{q}_t$  is a unit quaternion representing the amount of translation between the model and measured data sets. The optimal value of  $\mathbf{q}_t$ ,  $\mathbf{q}_t^o$ , is to be found at the end of the optimization. The unit quaternion  $\mathbf{q}_R$ , given by

$$\mathbf{q}_R = [q_0 \ q_1 \ q_2 \ q_3]^T \quad (5.15)$$

may be used to represent the rotation matrix  $\mathbf{R}$  via

$$\mathbf{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix} \quad (5.16)$$

Furthermore,

$$\mathbf{q}_t = [q_4 \ q_5 \ q_6]^T \quad (5.17)$$

$$\mathbf{q} = [\mathbf{q}_R \ | \ \mathbf{q}_t]^T \quad (5.18)$$

By minimizing  $f(\mathbf{q})$  in closed form (see Besl and McKay (1992) for details), the registered (i.e. corrected by means of rigidity) coordinates of the measured data set  $\mathbf{x}_m^r$ , can be obtained in the following manner.

$$\mathbf{x}_m^m = \overset{o}{\mathbf{R}}(\mathbf{q}_R)\mathbf{x}_m^r + \overset{o}{\mathbf{q}}_t \quad (5.19)$$

which leads to

$$\mathbf{x}_m^r = \overset{o}{\mathbf{R}}(\mathbf{q}_R)^{-1} \left( \mathbf{x}_m^m - \overset{o}{\mathbf{q}}_t \right) \quad (5.20)$$

The registration should be realized in all frames of the motion, therefore  $\overset{o}{\mathbf{q}}_t$  and  $\overset{o}{\mathbf{R}}$  is different in each frame of the motion. The matrix having components of  $\overset{o}{\mathbf{q}}_t$  for each

frame is represented with  $\overset{o}{\mathbf{Q}}_t$  and the matrix having components of  $\overset{o}{\mathbf{R}}$  for each frame is  $\overset{o}{\mathbf{RR}}$ .  $\mathbf{MM}^m$  is the model data points coordinates in the motion reference frame. To reach the registered coordinates of the markers (in the motion reference frame),  $\mathbf{MM}^r$  (of size  $M \times 3 \times K$ ), the following syntax is used :

```

for k=1:K
    for m=1:M
         $\mathbf{MM}^r(m, :, k) = \overset{o}{\mathbf{RR}}(:, :, k)^{-1}(\mathbf{MM}^m(m, :)^T - \overset{o}{\mathbf{Q}}_t(:, k))$       (5.21)
    end
end

```

### 5.5 Curve Fitting to Position to Reach Velocity and Acceleration of Markers

The data points obtained at the end of an experiment usually requires curve fitting operations. Primarily, two different techniques exist to fit curves to data points. One of the procedures is the derivation of a single curve to realize the general trend, without necessarily matching the individual data point, named as regression. On the other hand, interpolation is used if the fitted curve is required to pass through the data points. In general, experimental data requires a regression type of curve fitting operation (Chapra & Canale, 2003). In this study, MATLAB<sup>®</sup> curve fitting algorithms are utilized for the curve fitting processes.

Curve fitting is finding the parameters of a curve that fits a set of points. In this thesis, only parametric curve fitting is employed. In general, a set of points can be represented by dependent and independent variables. For each independent variable, a value is assigned as the corresponding dependent variable. In order to obtain the coefficient estimates of the fitted curve, the least squares method is used which

minimizes the summed square of the errors (or residuals). The function to be minimized is given by:

$$SSE = \sum_{t=t_0}^{t=t_{\max}} (g(t) - g_f(t))^2 \quad (5.22)$$

where  $g(t)$  is the original data (the dependent variable) to which a curve is fitted and  $t$  is the independent variable.  $g_f(t)$  represents the curve to be fitted.  $g - g_f$  is defined as the error (or residual). In other words, the residual is the difference between the real value and the predicted value of the dependent variable. In Figure 5.4, the vertical distances of the points to the curve, corresponding to the residuals, are shown. The residuals and  $SSE$  are quite useful for the evaluation of the goodness of fits.

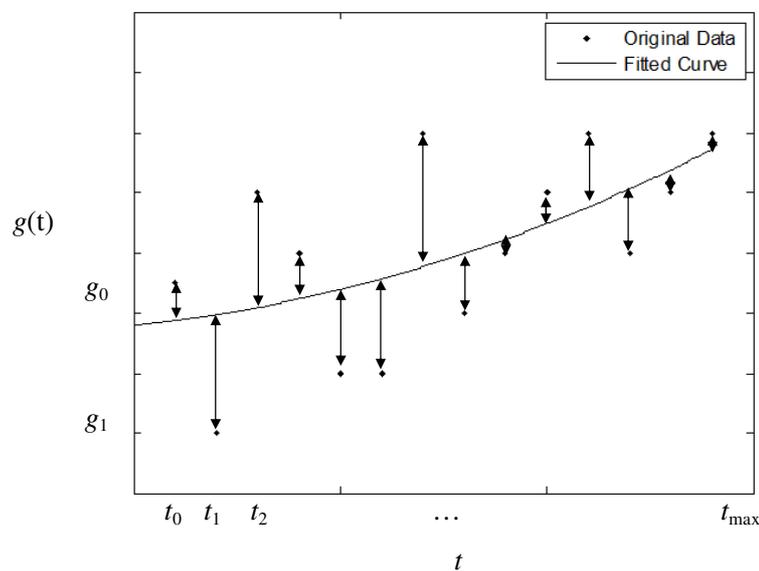


Figure 5.4 A second order polynomial curve fitting

When a curve is to be fit, firstly, the general behavior of the data set has to be visualized and various alternatives sharing similar behavior with the data have to be tried. Then a best fit can be obtained with the help of the approach in the evaluation of goodness.

In this study, the independent variable is time ( $t$ ), whereas the x-motion, y-motion and z-motion coordinates of each marker during the motion are the dependent variables.

### **5.5.1 Goodness of Fit**

Having fitted curves to a set of data, in order to determine the best curve along the alternatives, some goodness of fit measures should be examined. Comparing the graphical and numerical results yields the best alternative of a curve. Statistics assists in determining the quantitative goodness of fit. The initial approach in the determination of the best curve is a visual examination of the fitted function with the original data. The behavior of the residuals shows the goodness of the fit. For instance, if they are systematically positive or negative, the fit is a poor fit. If the residuals appear to behave randomly, it suggests that the model fits the data well. Figure 5.5 shows linear and quadratic fits on the same data with the residuals.

Here, the residuals of the quadratic fit behave in a positive way indicating that it is a poor fit, whereas, linear fit residuals are distributed randomly around zero indicating that linear fit is a better fit for the data set.

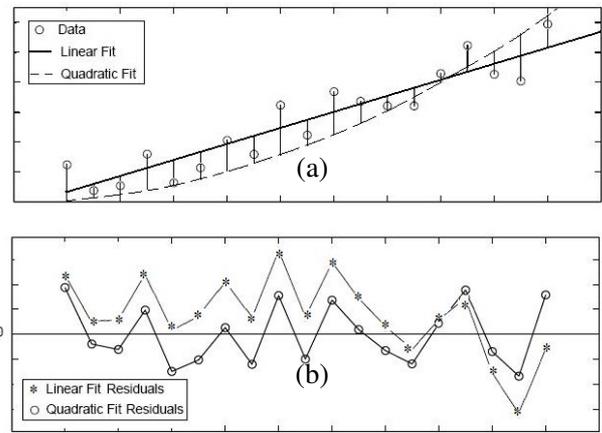


Figure 5.5 (a) Original and fitted data (b) Residuals of fits  
(Adapted from Mathworks, Inc., 2002)

Examination of the future behavior of the fit is another way of deciding the goodness of the curve. For some cases, the fit is compatible with the set of data presented; however, when the future behavior of the fit is observed, unexpected cases may occur. Widening the axis limits is an alternative to inspect the future behavior of the fitted curve. In Figure 5.6 population of a town is presented. If a sixth degree of polynomial is fitted, for the years up to 2000, the result seems satisfactory. However, when the future behavior of the fit is analyzed, the result shows fit has a divergent trend.

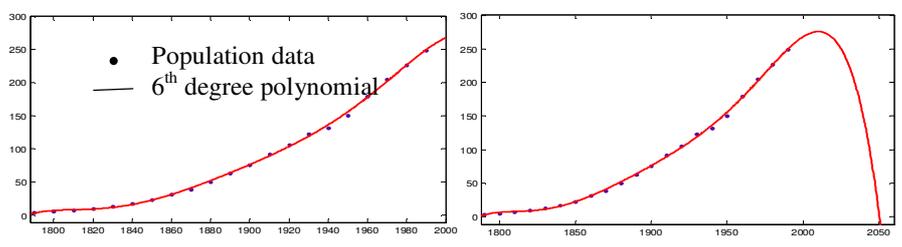


Figure 5.6 Divergence of a fit

Together with the graphical approach, numerical results of a fitting process can also reveal the goodness of a fit. If the sum of squares due to error (*SSE*) is close to zero, the curve is assumed to have a good fitting result. Also, confidence intervals of the coefficients belonging to the fit should be checked. A fit with a larger confidence interval is likely to be a less accurate fit and it should be eliminated. In practice, for the best function to be fit, all alternative models should be analyzed both with graphical and numerical types. Examining the fits only graphically or numerically may yield a poor type of choice.

### **5.5.2 Curve Fitting Application to the Horse Motion**

The components of the registered position vector of the *m*'th marker in the three motion axes are denoted as  $P_x^m(t)$ ,  $P_y^m(t)$ ,  $P_z^m(t)$ . Curve fitting operation is applied to these three components expressed in the motion reference frame. After fitting curves to the components of the position vector, the linear velocity and acceleration of the markers, in the motion reference frame will be obtained. However, when a curve fitting operation is applied to the position vector, the rigid body constraints will be violated. Therefore, the registered position data is used together with the velocity and acceleration data obtained by taking time derivative of the curve fitted position data.

#### **5.5.2.1 Curve Fitting to Position Data in the X-Motion Direction**

The horses have to be warmed up in the manege before being recorded by the cameras. This is essential to obtain steady motion with nearly a constant speed. Therefore, for the *x* coordinate of the markers motion in the motion reference frame, a linear polynomial is fitted.

### 5.5.2.2 Curve Fitting to Position Data in the Y (Gravity) and Z Motion Directions

The characteristics of each horse in each gait is different. The common characteristic is the periodicity of the motion. Therefore, periodic functions in the form of Fourier series are fitted to the position data both in the y and z motion directions. As is well known, the Fourier series is a sum of sine and cosine functions to describe a periodic signal. The trials have revealed that, fitting a Fourier function is more suitable than fitting polynomials of high degree or fitting sine functions. In order to decide upon the optimum degree,  $N^o$ , of the Fourier series to be fitted, the following fact is employed.

Let the fitted Fourier series be represented as:

$$g_F(t) = a_0 + \sum_{n=1}^N a_n \cos(n\omega t) + b_n \sin(n\omega t) \quad (5.23)$$

where  $\omega$  is the fundamental frequency of the signal,  $N$  is the number of harmonics in the series, and  $a_0$ ,  $a_n$  and  $b_n$  are the coefficients to be determined. The fact proposed here is that the Fourier equation given by Equation (5.23) has at most  $2N$  real, extreme (maximum, minimum or inflection) points. The proof of the proposed fact is given below.

In order to find the extremes of the function  $g_F(t)$ , the second derivative,  $\ddot{g}_F(t)$  should be equated to zero. Firstly, it is noted that  $g_F(t)$  can be expressed as;

$$g_F(\theta) = a_0 + a_1 \cos(\theta) + b_1 \sin(\theta) + a_2 \cos(2\theta) + b_2 \sin(2\theta) + \dots + a_N \cos(N\theta) + b_N \sin(N\theta) \quad (5.24)$$

where

$$\theta = \omega t \quad (5.25)$$

Hence, the derivative of  $g_F(\theta)$  with respect to  $\theta$  is given by:

$$\frac{dg_F(\theta)}{d\theta} = -a_1\sin(\theta)+b_1\cos(\theta) - 2a_2\sin(2\theta)+2b_2\cos(2\theta)+\dots - Na_N\sin(N\theta)+Nb_N\cos(N\theta) \quad (5.26)$$

Substituting the trigonometric identities;

$$\sin(2\theta)=2\cos\theta\sin\theta \quad (5.27)$$

$$\cos(2\theta)=(\cos\theta)^2+(\sin\theta)^2 \quad (5.28)$$

$$\sin(3\theta)=3(\cos\theta)^2\sin\theta - (\sin\theta)^3 \quad (5.29)$$

$$\cos(3\theta)=(\cos\theta)^3 - 3\cos\theta(\sin\theta)^2 \quad (5.30)$$

etc., into Equation (5.26), one obtains an expression in terms of  $\cos\theta$  and  $\sin\theta$ . Finally introducing the identities;

$$\sin\theta = \frac{2\xi}{1+\xi^2} \quad (5.31)$$

$$\cos\theta = \frac{1-\xi^2}{1+\xi^2} \quad (5.32)$$

$$\xi = \tan(\theta/2) \quad (5.33)$$

into the resulting expression and equating the result to zero, one obtains a polynomial equation of degree  $2N$  in  $\xi$ . Clearly, the maximum number of roots of  $\frac{dg_F(\theta)}{d\theta} = 0$  is  $2N$ . Therefore,  $g_F(t)$  has at most  $2N$  extreme points.

Let  $E$  be the number of extremes, i.e. the number of maxima and minima observed in a periodic curve. Using the fact proved above, a Fourier function with at least  $N=E/2$

order has to be fitted to that curve to avoid under-fitting. Another fit with order  $N=E/2+1$  can also be tested. Orders higher than  $E/2+1$  may result in over-fitting.

In the applications of curve fitting to position coordinates in the y and z motion directions, the user is firstly made to visualize the registered position data plots of each marker and asked to enter the number of extremes, E, observed. Then, two alternative Fourier fits, with orders  $E/2$  and  $(E/2 + 1)$ , are presented. The user decides upon the fit to be used. The suggested, default order is, however,  $E/2$ .

### **5.5.3 Reaching the Velocity and Acceleration Data**

Having fitted polynomial fits to the x-motion, and Fourier fits to the y and z motion position data, as the next step, velocities ( $V_x^m(t), V_y^m(t), V_z^m(t)$ ) and accelerations ( $A_x^m(t), A_y^m(t), A_z^m(t)$ ) of each marker m can be obtained. The derivatives of the position (fitted curve) data in all axes are calculated analytically, yielding the velocity. As expected, the velocity of each marker in the x-motion direction is constant, whereas, the velocity of each marker in the y and z motion directions is a periodic curve.

Similarly, the acceleration of each marker can be calculated analytically by differentiating the obtained velocity. Acceleration of the markers in the x-motion direction is zero as expected. On the contrary, the accelerations in the y and z motion directions have a varying nature (sinusoidal curve, again) due to the up and down (vertical) and side to side (lateral) movements of the horse body. It should be noted that, the acceleration in the y direction is the most important for the rider during motion (see chapter 7).

In Figure 5.7, a sample series of plots are presented, showing the resulting curve fitting process together with the velocity and acceleration data. The average velocity ( $V_{x,avg}^m$ ) is the slope of the fitted first degree polynomial and acceleration is zero.

A sample group of graphs are shown in Figure 5.8 illustrating the results of both proposed Fourier fits applied to the position of a marker in the y- motion direction for a trotting horse. Since results of both fit types are presented, one can compare the characteristics of the motion. Furthermore, other indicators for the goodness the fits are presented to the user in the software.

Finally, a similar procedure is applied to the z motion coordinates, the resultant offered graphs are presented in Figure 5.9, with other fitness indicators. The selection of the suitable fit is left to the user as in the previous case.

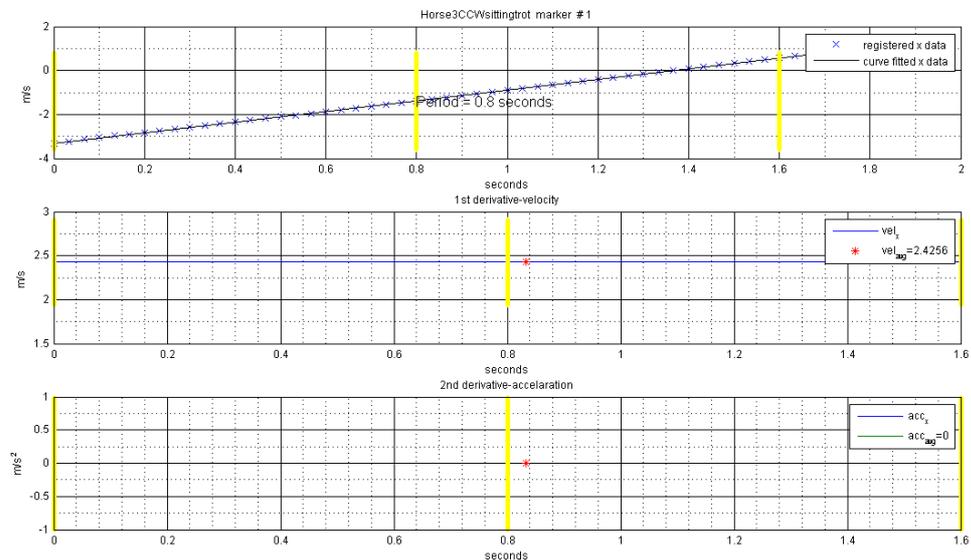


Figure 5.7 Kinematic properties of a marker in the x-motion direction for a trotting horse

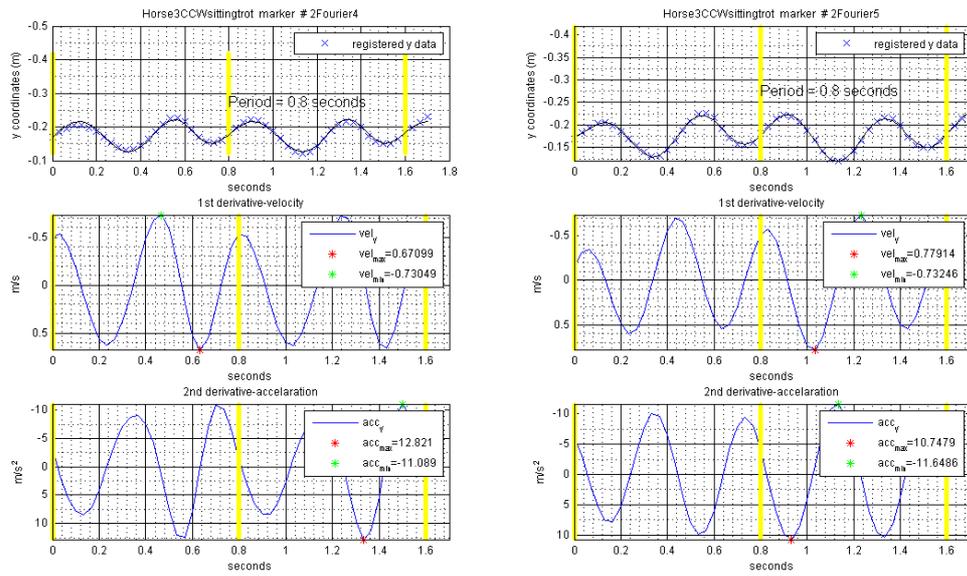


Figure 5.8 Kinematic properties of a marker in the y-motion direction for a trotting horse

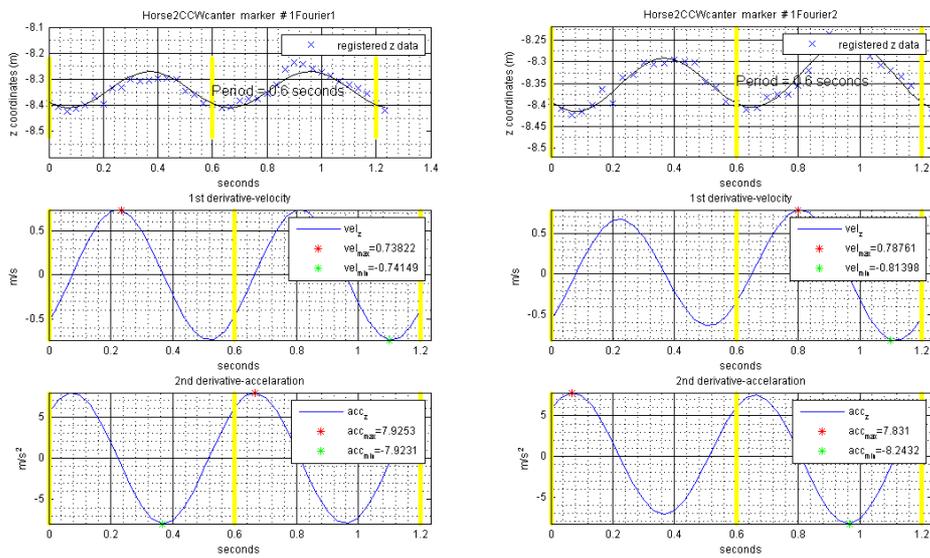


Figure 5.9 Kinematic properties of a marker in the z-motion direction for a trotting horse

## CHAPTER 6

### KINEMATIC PROPERTIES OF ANY POINT ON A RIGID BODY

Provided that the kinematic properties of at least three points on a rigid body are known, it can be proved that the kinematic data of any other point can also be reached. Firstly, the approach used to obtain the angular kinematic data of a rigid body is presented. Next, it is demonstrated that the position, velocity and acceleration of any point on the rigid body could be reached using the given information regarding the 3 markers.

#### 6.1 Determination of the Angular Velocity of a Rigid Body

Consider three points on the rigid body (corresponding to the markers on the horse) whose kinematic data have already been obtained and represented in a fixed reference frame. The following equations relate the velocities of the 3 points using the relative velocity concept;

$$\mathbf{v}_1 = \mathbf{v}_2 + \boldsymbol{\omega} \times \mathbf{r}_{21} \quad (6.1)$$

$$\mathbf{v}_1 = \mathbf{v}_3 + \boldsymbol{\omega} \times \mathbf{r}_{31} \quad (6.2)$$

where  $\mathbf{r}_{ij}$  is the position vector from marker  $i$  to marker  $j$ ;  $\mathbf{v}_1$ ,  $\mathbf{v}_2$ , and  $\mathbf{v}_3$  are the velocity vectors of the markers 1, 2 and 3; and  $\boldsymbol{\omega}$  is the angular velocity of the rigid body. Here,  $\mathbf{r}_{ij}$ ,  $\mathbf{v}_1$ ,  $\mathbf{v}_2$ , and  $\mathbf{v}_3$  are assumed to be known, whereas  $\boldsymbol{\omega}$  is to be obtained.

There are two solutions ( $\boldsymbol{\omega}^1$  and  $\boldsymbol{\omega}^2$ ), yielding the same result, for the angular velocity of the body. These solutions are given by;

$$\boldsymbol{\omega}^1 = \frac{(\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1)}{-\mathbf{r}_{31} \cdot (\mathbf{v}_2 - \mathbf{v}_1)} \quad (6.3)$$

and

$$\boldsymbol{\omega}^2 = \frac{(\mathbf{v}_2 - \mathbf{v}_1) \times (\mathbf{v}_3 - \mathbf{v}_1)}{\mathbf{r}_{21} \cdot (\mathbf{v}_3 - \mathbf{v}_1)} \quad (6.4)$$

which are valid provided that the denominators are not equal to zero, i.e.,

$$-\mathbf{r}_{31} \cdot (\mathbf{v}_2 - \mathbf{v}_1) \neq 0 \quad (6.5)$$

$$\mathbf{r}_{21} \cdot (\mathbf{v}_3 - \mathbf{v}_1) \neq 0 \quad (6.6)$$

The proof for Equations (6.3) and (6.4) is given below.

Consider the two vector equations given by;

$$\mathbf{K}_1 \times \mathbf{u} = \mathbf{K}_2 \quad (6.7)$$

$$\mathbf{K}_3 \times \mathbf{u} = \mathbf{K}_4 \quad (6.8)$$

where  $\mathbf{K}_1$ ,  $\mathbf{K}_2$ ,  $\mathbf{K}_3$  and  $\mathbf{K}_4$  are known vectors, while  $\mathbf{u}$  is the unknown vector. Taking the cross product of Equations (6.7) and (6.8) side by side, one obtains;

$$(\mathbf{K}_1 \times \mathbf{u}) \times (\mathbf{K}_3 \times \mathbf{u}) = \mathbf{K}_2 \times \mathbf{K}_4 \quad (6.9)$$

which may be simplified as

$$((\mathbf{K}_1 \times \mathbf{u}) \cdot \mathbf{u})\mathbf{K}_3 - ((\mathbf{K}_1 \times \mathbf{u}) \cdot \mathbf{K}_3)\mathbf{u} = \mathbf{K}_2 \times \mathbf{K}_4 \quad (6.10)$$

where

$$(\mathbf{K}_1 \times \mathbf{u}) \cdot \mathbf{u} = 0 \quad (6.11)$$

Equation (6.10) reduces to

$$-\mathbf{K}_2 \cdot \mathbf{K}_3 \mathbf{u} = \mathbf{K}_2 \times \mathbf{K}_4 \quad (6.12)$$

which yields

$$\mathbf{u}^1 = \frac{\mathbf{K}_2 \times \mathbf{K}_4}{-\mathbf{K}_2 \cdot \mathbf{K}_3} \quad (6.13)$$

Similarly, the cross products of Equations (6.8) and (6.7) will result in;

$$\mathbf{u}^2 = \frac{\mathbf{K}_2 \times \mathbf{K}_4}{\mathbf{K}_1 \cdot \mathbf{K}_4} \quad (6.14)$$

Arranging (6.1) and (6.2) in the form of (6.7) and (6.8) and comparing, one obtains;

$$\mathbf{K}_1 \Leftrightarrow \mathbf{r}_{21} \quad (6.15)$$

$$\mathbf{K}_2 \Leftrightarrow (\mathbf{v}_2 - \mathbf{v}_1) \quad (6.16)$$

$$\mathbf{K}_3 \Leftrightarrow \mathbf{r}_{31} \quad (6.17)$$

$$\mathbf{K}_4 \Leftrightarrow (\mathbf{v}_3 - \mathbf{v}_1) \quad (6.18)$$

Substituting Equations (6.15) - (6.18) into (6.13) and (6.14), one obtains (6.3) and (6.4) which completes the proof.

There exists unique solution for  $\mathbf{u}$ , due to the compatibility condition given by Equation (6.19) (obtained by equating the denominators of the right hand side of Equations (6.13) and (6.14)).

$$\mathbf{K}_1 \cdot \mathbf{K}_4 = -\mathbf{K}_2 \cdot \mathbf{K}_3 \quad (6.19)$$

which, upon substituting Equations (6.15) - (6.18), becomes;

$$\mathbf{r}_{21} \cdot (\mathbf{v}_3 - \mathbf{v}_1) = -(\mathbf{v}_2 - \mathbf{v}_1) \cdot \mathbf{r}_{31} \quad (6.20)$$

which may be simplified as

$$\mathbf{r}_{21} \cdot \mathbf{v}_3 - \mathbf{r}_{21} \cdot \mathbf{v}_1 + \mathbf{r}_{31} \cdot \mathbf{v}_2 - \mathbf{r}_{31} \cdot \mathbf{v}_1 = 0 \quad (6.21)$$

Since point 1 and point 2 are on the same rigid body, the equation,

$$\mathbf{r}_{21} \cdot \mathbf{v}_1 = \mathbf{r}_{21} \cdot \mathbf{v}_2 \quad (6.22)$$

should be satisfied. Similarly, point 1 and point 3 are on the same rigid body, satisfying the equation;

$$\mathbf{r}_{31} \cdot \mathbf{v}_1 = \mathbf{r}_{31} \cdot \mathbf{v}_3 \quad (6.23)$$

Substituting Equation (6.22) and Equation (6.23) into (6.21);

$$\mathbf{r}_{21} \cdot \mathbf{v}_3 - \mathbf{r}_{21} \cdot \mathbf{v}_2 + \mathbf{r}_{31} \cdot \mathbf{v}_2 - \mathbf{r}_{31} \cdot \mathbf{v}_3 = 0 \quad (6.24)$$

which may be written as

$$\mathbf{r}_{21} \cdot (\mathbf{v}_3 - \mathbf{v}_2) + \mathbf{r}_{31} \cdot (\mathbf{v}_2 - \mathbf{v}_3) = 0 \quad (6.25)$$

leading to

$$(\mathbf{r}_{21} - \mathbf{r}_{31}) \cdot (\mathbf{v}_3 - \mathbf{v}_2) = 0 \quad (6.26)$$

where

$$\mathbf{r}_{21} - \mathbf{r}_{31} = \mathbf{r}_{23} \quad (6.27)$$

Therefore, Equation (6.26) becomes;

$$\mathbf{r}_{23} \cdot (\mathbf{v}_3 - \mathbf{v}_2) = 0 \quad (6.28)$$

which must be true, since point 2 and point 3 are on the same rigid body. Therefore, the compatibility condition is satisfied.

The singularity, where the solutions given by (6.3) and (6.4) are not valid, can be obtained via the following approach. By rearranging (6.1) and (6.2);

$$\mathbf{v}_2 - \mathbf{v}_1 = -(\boldsymbol{\omega} \times \mathbf{r}_{21}) \quad (6.29)$$

$$\mathbf{v}_3 - \mathbf{v}_1 = -(\boldsymbol{\omega} \times \mathbf{r}_{31}) \quad (6.30)$$

Equation (6.3) can be rewritten as;

$$\boldsymbol{\omega}^1 = \frac{(\boldsymbol{\omega} \times \mathbf{r}_{21}) \times (\boldsymbol{\omega} \times \mathbf{r}_{31})}{\mathbf{r}_{31} \cdot (\boldsymbol{\omega} \times \mathbf{r}_{21})} \quad (6.31)$$

which can be simplified as

$$\boldsymbol{\omega}^1 = \frac{[\boldsymbol{\omega} \cdot (\mathbf{r}_{21} \times \mathbf{r}_{31})] \boldsymbol{\omega}}{(\mathbf{r}_{21} \times \mathbf{r}_{31}) \cdot \boldsymbol{\omega}} \quad (6.32)$$

On the other hand, by defining  $\mathbf{n}$  (i.e. the unit normal vector of the plane formed by the markers 1, 2 and 3) via the equation;

$$\mathbf{n} = \frac{(\mathbf{r}_{21} \times \mathbf{r}_{31})}{|\mathbf{r}_{21} \times \mathbf{r}_{31}|} = \frac{(\mathbf{r}_{21} \times \mathbf{r}_{31})}{c} \quad (6.33)$$

Equation (6.32) may be rewritten in terms of  $\mathbf{n}$  as;

$$\boldsymbol{\omega}^1 = \frac{[\boldsymbol{\omega} \cdot \mathbf{nc}] \boldsymbol{\omega}}{\mathbf{nc} \cdot \boldsymbol{\omega}} \quad (6.34)$$

where both the numerator and the denominator of the equation becomes zero only if

$$\boldsymbol{\omega} \cdot \mathbf{n} = 0 \quad (6.35)$$

Using the aforementioned fact, it can be said that, the angle between  $\mathbf{n}$  and  $\boldsymbol{\omega}$  should be as far as possible from  $\pi/2$  to avoid a singularity. In other words,  $\mathbf{n}$  and  $\boldsymbol{\omega}$  should be parallel or anti-parallel to each other as much as possible.

In the ideal case, the two values for the angular velocity,  $\boldsymbol{\omega}^1$  and  $\boldsymbol{\omega}^2$ , are expected to be equal to each other. However, due to the errors arising from the camera system and the curve fitting operations,... etc, the results differ in practice. Additionally,  $\mathbf{v}_1$

is expressed in terms of  $\mathbf{v}_2$  and  $\mathbf{v}_3$  (see Equations (6.1) and (6.2)) in the approach above. One can express  $\mathbf{v}_2$  in terms of  $\mathbf{v}_1$  and  $\mathbf{v}_3$  or express  $\mathbf{v}_3$  in terms of  $\mathbf{v}_1$  and  $\mathbf{v}_2$  and obtain 4 different, but similar, values for the angular velocity. Therefore, totally six different results are obtained. Let version number  $i$  refer to the case where  $\mathbf{v}_i$  is expressed in terms of the remaining velocities, as in Equations (6.1) and (6.2). Therefore, six values of angular velocities, namely  $(\boldsymbol{\omega}^{1,ver1}, \boldsymbol{\omega}^{2,ver1}, \boldsymbol{\omega}^{1,ver2}, \boldsymbol{\omega}^{2,ver2}, \boldsymbol{\omega}^{1,ver3}, \boldsymbol{\omega}^{2,ver3})$  are obtained as a function of time. In the ideal case, all values should be equal to each other.

In the software, the user is provided with the plots of all six omega values and the average. The decision of which solution(s) to consider is left to the user. Overall value of  $\boldsymbol{\omega}$  is calculated as the average of the selected values.

## 6.2 Determination of the Angular Acceleration of a Rigid Body

The following equations relate the accelerations of 3 markers on the same rigid body using the relative acceleration concept.

$$\mathbf{a}_1 = \mathbf{a}_2 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{21}) + \boldsymbol{\alpha} \times \mathbf{r}_{21} \quad (6.36)$$

$$\mathbf{a}_1 = \mathbf{a}_3 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{31}) + \boldsymbol{\alpha} \times \mathbf{r}_{31} \quad (6.37)$$

where  $\mathbf{r}_{ij}$  is the position vector from marker  $i$  to marker  $j$ ;  $\mathbf{a}_1$ ,  $\mathbf{a}_2$ , and  $\mathbf{a}_3$  are the acceleration vectors of the markers,  $\boldsymbol{\omega}$  is the angular velocity (obtained previously); and  $\boldsymbol{\alpha}$  is the angular acceleration of the rigid body. Here,  $\mathbf{r}_{ij}$ ,  $\mathbf{a}_1$ ,  $\mathbf{a}_2$ ,  $\mathbf{a}_3$  and  $\boldsymbol{\omega}$  are known, whereas  $\boldsymbol{\alpha}$  is to be found. There are two solutions, yielding the same result, for the angular acceleration of the body ( $\boldsymbol{\alpha}^1$  and  $\boldsymbol{\alpha}^2$ ) which can be obtained by using the fact proved in the previous section via Equations (6.13) and (6.14). These solutions are given by

$$\boldsymbol{\alpha}^1 = \frac{(\mathbf{a}_2 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{21})) \times (\mathbf{a}_3 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{31}))}{-(\mathbf{a}_2 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{21})) \cdot \mathbf{r}_{31}} \quad (6.38)$$

and

$$\boldsymbol{\alpha}^2 = \frac{(\mathbf{a}_2 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{21})) \times (\mathbf{a}_3 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{31}))}{\mathbf{r}_{21} \cdot (\mathbf{a}_3 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{31}))} \quad (6.39)$$

Similar to the angular velocity case, it can be shown that for the case of no errors in measurement,  $\boldsymbol{\alpha}^1$  and  $\boldsymbol{\alpha}^2$  yield the same result. It can also be shown that the singularity associated with the solutions is the same as in the angular velocity case.

Similar to the angular velocity, six different solutions are obtained for the angular acceleration. These solutions are symbolized as  $\boldsymbol{\alpha}^{1,ver1}$ ,  $\boldsymbol{\alpha}^{2,ver1}$ ,  $\boldsymbol{\alpha}^{1,ver2}$ ,  $\boldsymbol{\alpha}^{2,ver2}$ ,  $\boldsymbol{\alpha}^{1,ver3}$ ,  $\boldsymbol{\alpha}^{2,ver3}$ . In the ideal case, all values should be equal to each other. The plots for each solution are shown to the user, and it is expected that the user selects the angular acceleration(s) to be used. Overall value for  $\boldsymbol{\alpha}$  is taken to be the average of the selected data.

### 6.3 Position of the Invisible Marker

During the motion of the rigid body, a point on the body may be invisible to the cameras. Such a case occurs, for instance, when the rider is on the horse, thus blocking the saddle marker with the lower part of his body. In this section, an approach, which enables one to determine the position of such an invisible marker, is introduced.

Firstly, it is necessary to determine the relative position of the invisible marker with respect to the 3 visible markers, using a static shot which shows the 3 visible markers and the invisible marker.

In order to achieve this task, a body fixed reference frame is introduced. Once the position of the invisible marker in the body frame is obtained, the position of the same marker during the motion (while it is unseen) can be found. Figure 6.1 shows the invisible marker, S, in the motion frame, with origin at  $O_M$ , and the body fixed frame, with origin  $O_B$ .

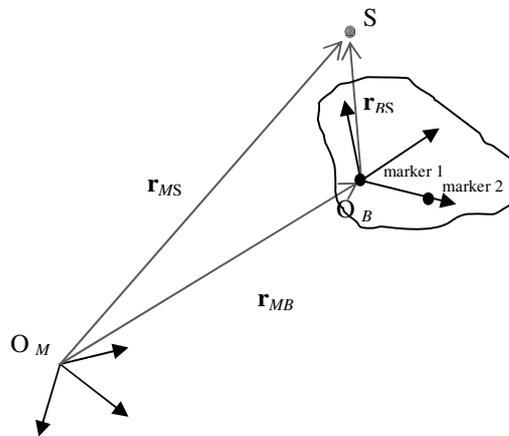


Figure 6.1 Motion and body reference frames in static shot

Referring to the figure, it is clear that

$$\mathbf{r}_{MS}^{(M)} = \mathbf{r}_{BS}^{(M)} + \mathbf{r}_{MB}^{(M)} \quad (6.40)$$

where the superscript  $(M)$  indicates that the vector is expressed in the motion reference frame. For instance,  $\mathbf{r}_{MS}^{(M)}$  refers to the vector  $\mathbf{r}_{MS}$  expressed in the motion frame.

Using the static shot, it is necessary to determine  $\mathbf{r}_{BS}$  in the body reference frame where the body reference frame is constructed such that the first marker is the origin

and the second marker is on the positive x-axis of the body frame. The unit vectors  $\mathbf{u}_1$ ,  $\mathbf{u}_2$  and  $\mathbf{u}_3$ , which are parallel to the x, y, z axes of the body frame, are determined via the following equations.

$$\mathbf{u}_1 = \frac{\mathbf{r}_2 - \mathbf{r}_1}{|\mathbf{r}_2 - \mathbf{r}_1|} \quad (6.41)$$

$$\mathbf{u}'_2 = \frac{\mathbf{r}_3 - \mathbf{r}_1}{|\mathbf{r}_3 - \mathbf{r}_1|} \quad (6.42)$$

$$\mathbf{u}_3 = \frac{\mathbf{u}_1 \times \mathbf{u}'_2}{|\mathbf{u}_1 \times \mathbf{u}'_2|} \quad (6.43)$$

$$\mathbf{u}_2 = \mathbf{u}_3 \times \mathbf{u}_1 \quad (6.44)$$

where  $\mathbf{r}_1$ ,  $\mathbf{r}_2$  and  $\mathbf{r}_3$  are the coordinates of markers 1, 2 and 3 respectively (expressed in the motion reference frame). Therefore, the rotation matrix relating the motion frame to the body frame is given by;

$$\mathbf{R}_{MB} = [\mathbf{u}_1 \mid \mathbf{u}_2 \mid \mathbf{u}_3] \quad (6.45)$$

where  $\mathbf{u}_1$ ,  $\mathbf{u}_2$  and  $\mathbf{u}_3$  are expressed in the motion frame and are known. Equation 6.40 can now be written as;

$$\mathbf{r}_{MS}^{(M)} = \mathbf{R}_{MB} \mathbf{r}_{BS}^{(B)} + \mathbf{r}_{MB}^{(M)} \quad (6.46)$$

which yields the (invisible) saddle marker's body coordinates,  $\mathbf{r}_{BS}^{(B)}$ , as

$$\mathbf{r}_{BS}^{(B)} = \mathbf{R}_{MB}^{-1} (\mathbf{r}_{MS}^{(M)} - \mathbf{r}_{MB}^{(M)}) \quad (6.47)$$

Having obtained the body fixed coordinates of the invisible marker on the saddle using the static shot, the position of the saddle marker (although it is invisible) can now be attained during the motion.

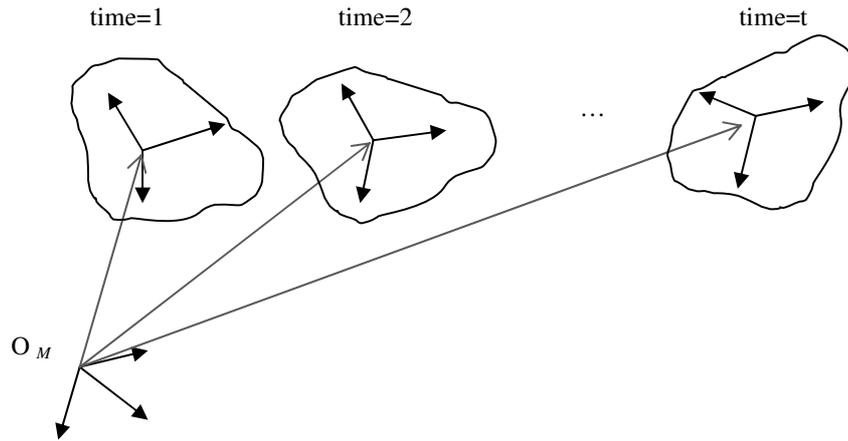


Figure 6.2 Motion and body reference frames during the motion

The position of the invisible saddle marker in the motion coordinates at any time  $t$ ,  $\mathbf{r}_{MS}^{(M)}(t)$  can now be obtained via the equation:

$$\mathbf{r}_{MS}^{(M)}(t) = \mathbf{R}_{MB}(t)\mathbf{r}_{BS}^{(B)} + \mathbf{r}_{MB}^{(M)}(t) \quad (6.48)$$

where  $\mathbf{r}_{BS}^{(B)}$  has already been obtained as given by Equation (6.47). Similar to the static case, the rotation matrix  $\mathbf{R}_{MB}(t)$  is given by

$$\mathbf{R}_{MB}(t) = [\mathbf{u}_1(t) \mid \mathbf{u}_2(t) \mid \mathbf{u}_3(t)] \quad (6.49)$$

where  $\mathbf{u}_1(t)$ ,  $\mathbf{u}_2(t)$  and  $\mathbf{u}_3(t)$  are obtained via the equations:

$$\mathbf{u}_1(t) = \frac{\mathbf{r}_2(t) - \mathbf{r}_1(t)}{|\mathbf{r}_2(t) - \mathbf{r}_1(t)|} \quad (6.50)$$

$$\mathbf{u}'_2(t) = \frac{\mathbf{r}_3(t) - \mathbf{r}_1(t)}{|\mathbf{r}_3(t) - \mathbf{r}_1(t)|} \quad (6.51)$$

$$\mathbf{u}_3(t) = \frac{\mathbf{u}_1(t) \times \mathbf{u}'_2(t)}{|\mathbf{u}_1(t) \times \mathbf{u}'_2(t)|} \quad (6.52)$$

$$\mathbf{u}_2(t) = \mathbf{u}_3(t) \times \mathbf{u}_1(t) \quad (6.53)$$

Therefore, the position vector of the invisible point in the motion reference frame at time  $t$ ,  $\mathbf{r}_{MS}^{(M)}(t)$ , having the components:  $P_x^s(t)$ ,  $P_y^s(t)$ ,  $P_z^s(t)$  has been obtained.

#### 6.4 Velocity and Acceleration of the Invisible Marker

The velocity and acceleration of the invisible marker S,  $\mathbf{v}_S$  and  $\mathbf{a}_S$ , can now be obtained in terms of the first marker velocity and acceleration,  $\mathbf{v}_1$  and  $\mathbf{a}_1$  via the equations:

$$\mathbf{v}_S = \mathbf{v}_1 + \boldsymbol{\omega} \times \mathbf{r}_{1S} \quad (6.54)$$

$$\mathbf{a}_S = \mathbf{a}_1 + \boldsymbol{\omega} \times (\boldsymbol{\omega} \times \mathbf{r}_{1S}) + \boldsymbol{\alpha} \times \mathbf{r}_{1S} \quad (6.55)$$

where all of the vectors are expressed in the motion frame. Note that,  $\mathbf{r}_{1S}$  denotes the vector from marker 1 to the saddle marker S.

If the second or the third marker is used in Equations (6.54) and (6.55), the same results are expected to be obtained. However, the results come out to be slightly different due to the measurement errors. Due to this reason, the velocity and

acceleration of the invisible marker are taken to be the average value of the 3 results obtained by using the 3 visible markers.

As a result, the velocity of the invisible saddle marker is obtained as  $\mathbf{v}_s(t)$  with the components  $V_x^s(t)$ ,  $V_y^s(t)$ ,  $V_z^s(t)$ , and the acceleration of the saddle marker is obtained as  $\mathbf{a}_s(t)$  with the components  $A_x^s(t)$ ,  $A_y^s(t)$ ,  $A_z^s(t)$ .

## CHAPTER 7

### HUMAN RESPONSE TO VIBRATION

Vibration is a mechanical wave which transfers energy. People are exposed to localized vibrations, or vibrations affecting the whole body, during transportation, or in their industrial environment (Mansfield, 2005). Similarly, horse riders are exposed to vibrations as well.

#### 7.1 Whole-Body Vibration

Whole-body vibration is a kind of vibration that affects all parts of a person's body exposed to a shaking surface. It is usually transmitted through a seat, backrests or floor from a vehicle. Most studies have been carried out in the perception of whole-body vibration in seated postures. Human response to vibration depends on the frequency, magnitude and duration of the vibration. People are most sensitive to whole-body vibration within a frequency range of 1 to 20 Hz. Vibration exposure leads to problems in the body. For instance, back pain is the mostly encountered effect of prolonged exposure to whole-body vibration. Other types of effects include motion sickness, discomfort, digestive and vascular disorders.

#### 7.2 Vibration Measurement

Investigation of vibrations should be quantified by means of objective measurements. The quantities, such as force and acceleration are to be measured at the driving point (i.e. the point at which the human is in contact with the vibration source).

### 7.2.1 Frequency Weightings

Response of people to vibration depends on the frequency of the vibration. Therefore, frequency weightings are required to simulate this dependence. These weightings are designed such that, the frequencies where the response of human body is most sensitive are not affected. On the other hand, the frequencies are attenuated where the response of the body is less sensitive. Consequently, the magnitude of a frequency weighted signal cannot be more than the unweighted signal.

A wide variety of frequency weightings can be found in the literature and in the standards (Table 7.1). For the whole body vibration, the most commonly used weightings are  $W_b$ ,  $W_d$  and  $W_k$ . For the vertical direction,  $W_b$  is applied according to BS 6841 (1987), whereas  $W_k$  is used according to ISO 2631 (1997-2003).  $W_d$  is used for the lateral and fore-aft directions, in both standards.

Table 7.1 Frequency weightings with the application areas and standards

Frequency weighting	Direction	Application area	Standard
$W_b$	Vertical	Whole-body	BS 6841
$W_c$	Fore-aft	Whole-body	ISO 2631-1, BS 6841
$W_d$	Lateral, fore-aft	Whole-body	ISO 2631-1, BS 6841
$W_e$	Roll, pitch, yaw	Whole-body (rotational)	ISO 2631-1, BS 6841
$W_f$	Vertical	Motion sickness	ISO 2631-1, BS 6841
$W_g$	Vertical	Activity interference	BS 6841
$W_h$	Lateral, fore-aft, vertical	Hand-arm	ISO 5349-1
$W_j$	Vertical	Head	ISO 2631-1
$W_k$	Vertical	Whole-body	ISO 2631-1
$W_m$	All	Building	ISO 2631-2

The weighting coefficients,  $W_b$ ,  $W_k$ , etc... couldn't be found in closed form in the available literature. Therefore, using the transfer functions provided in (Rimell and Mansfield, 2007) closed form expressions, where  $W_b$  and  $W_k$  are expressed in terms of frequency  $f$ , have been obtained for these coefficients.

The plot of  $W_b$  and  $W_k$  versus  $f$ , obtained with closed form expressions, is shown in Figure 7.1 (a). In the saddle motion analysis, the frequency range of interest is between 0.8 Hz and 1.8 Hz. Therefore Figure 7.1 (b) gives a close look at the graph for this frequency range.

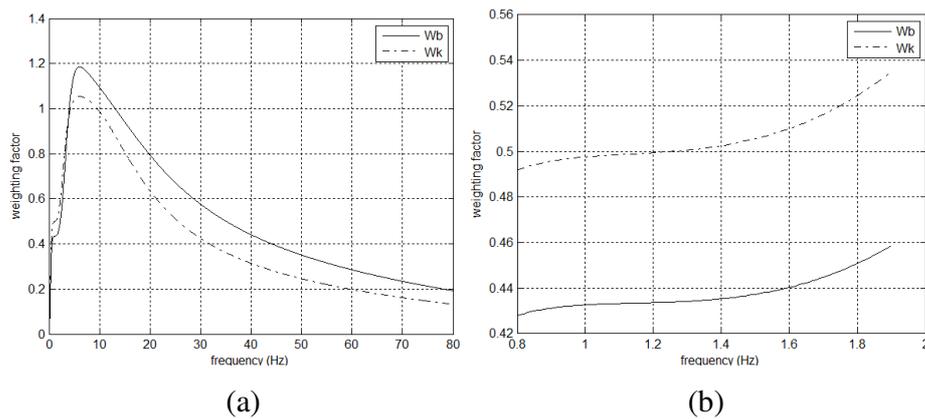


Figure 7.1 Weighing coefficients vs frequency

In the saddle motion analysis, the dominant direction is the vertical axis, corresponding to the  $y$ -motion direction.  $W_b$  and  $W_k$  are used for the whole-body vibration, according to the British Standards and International Organization for Standardization, respectively. The formulations of the necessary weighting functions,  $W_b(f)$  and  $W_k(f)$  have been obtained as explained above and inserted in

MATLAB<sup>®</sup> for usage. The weighted acceleration data in the vertical direction are indicated by  $A_{wb}(t)$  and  $A_{wk}(t)$  which are defined below.

$$A_{wb}(t) = W_b \cdot A_y(t) \quad (7.1)$$

$$A_{wk}(t) = W_k \cdot A_y(t) \quad (7.2)$$

### 7.2.2 Root Mean Square

Since the acceleration data is taken for a complete cycle(s), the mean of the acceleration cannot indicate the magnitude of the signal. Instead, the root mean square of the values is used. Root mean square acceleration, expressed in  $[m/s^2]$ , is defined via Equations (7.3-7.4), where  $\tau$  is the measurement duration which corresponds to two periods of a gait in horse motion analysis.

$$a_{wb, rms} = \sqrt{\frac{1}{\tau} \int_0^{\tau} A_{wb}^2(t) dt} \quad (7.3)$$

$$a_{wk, rms} = \sqrt{\frac{1}{\tau} \int_0^{\tau} A_{wk}^2(t) dt} \quad (7.4)$$

### 7.2.3 Peak Acceleration

Peak acceleration can be defined as the maximum instantaneous acceleration value of the entire acceleration data, in the measurement duration. The values of the peak weighted acceleration are defined as:

$$a_{wb, peak} = \max(\text{abs}(A_{wb}(t))) \quad (7.5)$$

$$a_{wk, peak} = \max(\text{abs}(A_{wk}(t))) \quad (7.6)$$

#### 7.2.4 Crest Factor (CF)

Ratio of the peak acceleration to the root mean square acceleration value is defined as the crest factor. It is a dimensionless value and its definition is given by Equation (7.7). Either  $W_b$  or  $W_k$  weighted acceleration ratios give the same value.

$$CF = \frac{a_{Wk, peak}}{a_{Wk, rms}} = \frac{a_{Wb, peak}}{a_{Wb, rms}} \quad (7.7)$$

#### 7.2.5 Vibration Dose Value (VDV)

Vibration dose value differs from the root mean square value such that, it emphasizes the shocks during the vibration. The unit of VDV is  $m/s^{1.75}$  and it is defined as (Mansfield, 2005):

$$VDV_{Wb} = \sqrt[4]{\int_0^{\tau} A_{Wb}^4(t) dt} \quad (7.8)$$

$$VDV_{Wk} = \sqrt[4]{\int_0^{\tau} A_{Wk}^4(t) dt} \quad (7.9)$$

#### 7.2.6 Calculation of the Daily Exposures

Once the vibration measurements are available, the time to reach the threshold value can be calculated. The total daily exposure times for the r.m.s. measurements is termed as  $A(8)$ , and the corresponding value in terms of VDV is  $VDV_{threshold}$  (Mansfield, 2005). The time to reach the  $A(8)$  threshold is denoted by  $T_{A(8)}$  and for VDV it is denoted by  $T_{VDV}$ .

Table 7.2 EAV and ELV

	EAV	ELV
A(8)	0.5 m/s <sup>2</sup>	1.15 m/s <sup>2</sup>
VDV <sub>threshold</sub>	9.1 m/s <sup>1.75</sup>	21 m/s <sup>1.75</sup>

The limit of the acceleration value that a person can be exposed to vibration without health damage is named as the Exposure Action Value. On the other hand, Exposure Limit Value is the acceleration value above which the person is exposed to health risks. Exposure Action Values (EAV) and Exposure Limit Values (ELV) are given in Table 7.2 as specified in the EU Physical Agents Directive. The corresponding exposure time thresholds for each of the weighting coefficients, in terms of r.m.s. accelerations, are calculated with the following equations:

$$T_{A(8),Wb}^{\text{action}} = 8 \left( \frac{A(8)^{\text{action}}}{a_{Wb, \text{rms}}} \right)^2 \quad (7.10)$$

$$T_{A(8),Wk}^{\text{action}} = 8 \left( \frac{A(8)^{\text{action}}}{a_{Wk, \text{rms}}} \right)^2 \quad (7.11)$$

$$T_{A(8),Wb}^{\text{limit}} = 8 \left( \frac{A(8)^{\text{limit}}}{a_{Wb, \text{rms}}} \right)^2 \quad (7.12)$$

$$T_{A(8),Wk}^{\text{limit}} = 8 \left( \frac{A(8)^{\text{limit}}}{a_{Wk, \text{rms}}} \right)^2 \quad (7.13)$$

where the allowable time values are obtained in hours. Also, exposure time thresholds for each of the weighting coefficients, in terms of VDV, can be calculated as follows:

$$T_{VDV,Wb}^{action} = \tau \left( \frac{VDV_{threshold}^{action}}{VDV_{Wb}} \right) \quad (7.14)$$

$$T_{VDV,Wk}^{action} = \tau \left( \frac{VDV_{threshold}^{action}}{VDV_{Wk}} \right) \quad (7.15)$$

$$T_{VDV,Wb}^{limit} = \tau \left( \frac{VDV_{threshold}^{action}}{VDV_{Wb}} \right) \quad (7.16)$$

$$T_{VDV,Wk}^{limit} = \tau \left( \frac{VDV_{threshold}^{action}}{VDV_{Wk}} \right) \quad (7.17)$$

For the case of vibrations induced by the horse on the rider, the accelerations in the vertical direction refer to the accelerations of the saddle point. The saddle point is a point on the saddle which is marked by the invisible marker S, which is assumed to be in contact with the rider at all times. In this study, the allowable daily distances (based on an 8 hour/ day riding) that can be covered by a specific horse at a specific gait are defined by;

$$D_{A(8),Wb}^{action} = V_{x,avg} T_{A(8),Wb}^{action} \quad (7.18)$$

$$D_{A(8),Wk}^{action} = V_{x,avg} T_{A(8),Wk}^{action} \quad (7.19)$$

$$D_{A(8),Wb}^{limit} = V_{x,avg} T_{A(8),Wb}^{limit} \quad (7.20)$$

$$D_{A(8),Wk}^{limit} = V_{x,avg} T_{A(8),Wk}^{limit} \quad (7.21)$$

where distances are in km and  $V_{x,avg}$  denotes the velocity of the horse in the motion direction in km/hr. In Equations (7.18)–(7.21), it is assumed that A(8) limits are employed.

Finally, the action and limit values of the distances when the  $VDV_{threshold}$  is preferred are defined as:

$$D_{\text{VDV},W_b}^{\text{action}} = V_{x,\text{avg}} T_{\text{VDV},W_b}^{\text{action}} \quad (7.22)$$

$$D_{\text{VDV},W_k}^{\text{action}} = V_{x,\text{avg}} T_{\text{VDV},W_k}^{\text{action}} \quad (7.23)$$

$$D_{\text{VDV},W_b}^{\text{limit}} = V_{x,\text{avg}} T_{\text{VDV},W_b}^{\text{limit}} \quad (7.24)$$

$$D_{\text{VDV},W_k}^{\text{limit}} = V_{x,\text{avg}} T_{\text{VDV},W_k}^{\text{limit}} \quad (7.25)$$

The daily allowable time of riding and distances based on both A(8) and VDV (both  $W_b$  and  $W_k$  weighted) can be used as a measure of the horse comfort level (see chapter 8).

## CHAPTER 8

### CASE STUDIES

In this chapter, the applications of the three camera vision system to the horse motion analysis and the details of the developed algorithms are presented. The following studies are based on various trials and experiments which have been performed in the early stages of the thesis study.

#### 8.1 The Camera Set Up

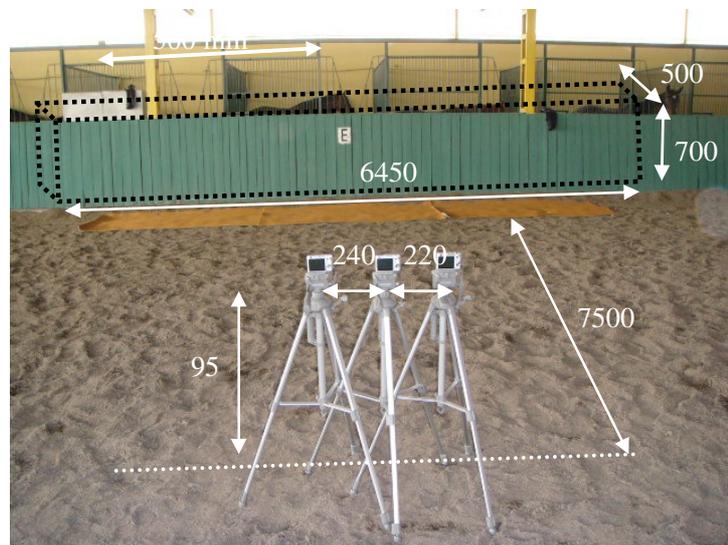


Figure 8.1 The system for the experiment (dimensions are in mm)

The scene is shown in Figure 8.1. To cover the required scene fully, the cameras are placed as parallel as possible. The perpendicular distance from the middle camera to the center of the scene is approximately 7.5 meters.

## 8.2 Calibration of the 3-Camera System

In the first step, for the synchronization of the cameras, the flash light is shown with the off and on modes and the frame differences between the cameras are obtained as  $\Delta F_{LM} = 53$  and  $\Delta F_{MR} = 73$ .

For the calibration process, the calibration board of size (800 x 1500 mm), with grid dimensions of 50x50 mm mounted on the calibration mechanism, is used. The total calibration volume is required as 500x700x6450 mm since the markers that are attached on the saddle and the body of the horses should be within the calibration volume for at least two periods of the motion. The calibration object is made to slide six times during the recording. For each slide, an equal number of images namely, 15, is selected to have a homogeneous distribution. Therefore, totally 90 different position and orientations of the calibration grid are chosen as the calibration images. Six of the images corresponding to each camera are shown in Figure 8.2.

By processing the 90 images with the MATLAB ® toolbox of Bouguet (2008), the intrinsic parameters (with the uncertainties) of the left, middle and right cameras are obtained. The results are tabulated in Table 8.1, Table 8.2 and Table 8.3, respectively. The extrinsic parameters of each camera pair, on the other hand, are given in Table 8.4.

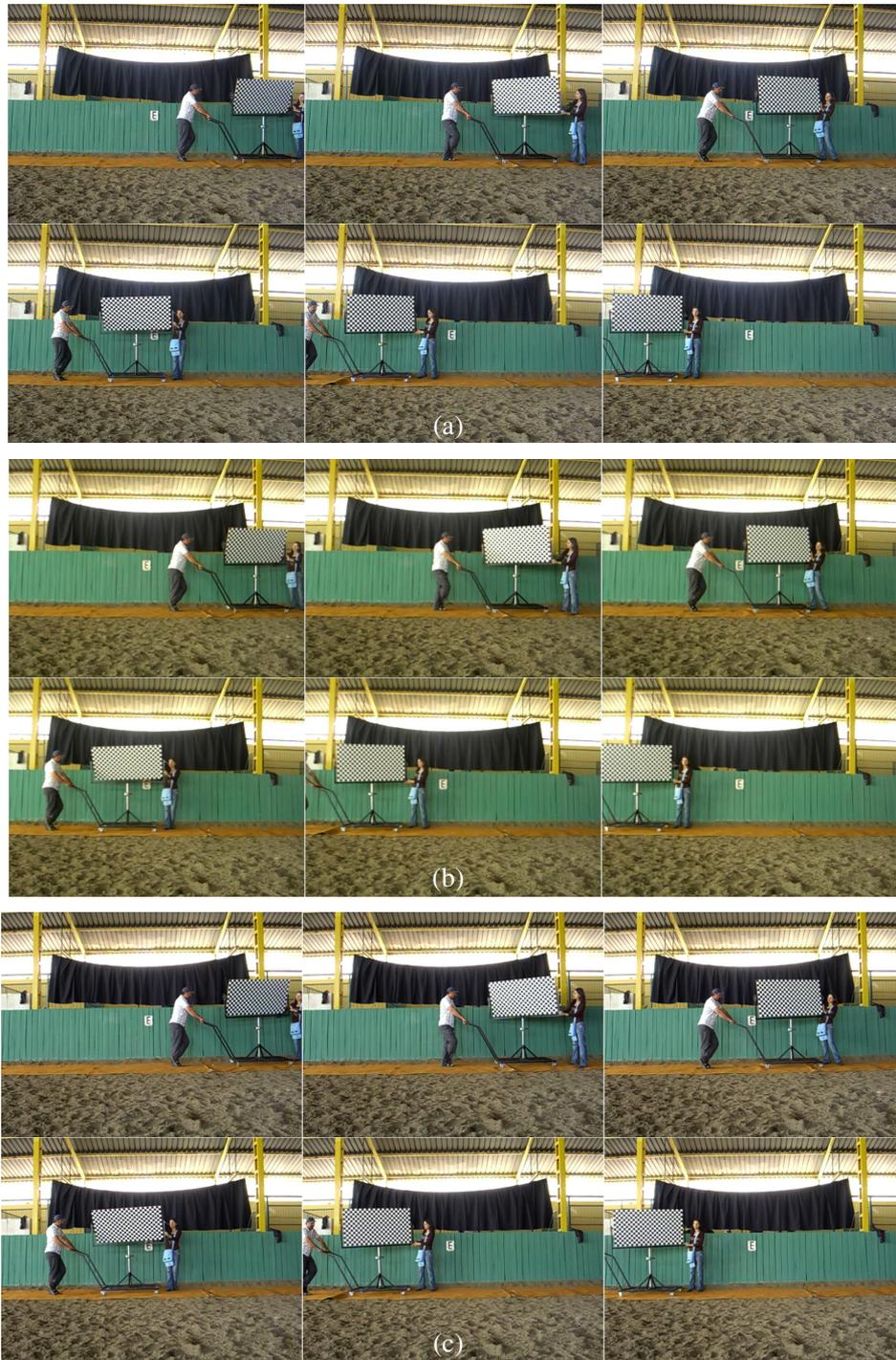


Figure 8.2 Images of the calibration object in different positions as viewed from the (a) left camera (b) middle camera (c) right camera

Table 8.1 Intrinsic parameters of the left camera

(mm)	Left and middle camera calibration	Left and right camera calibration
<b>Focal length</b>	[ 729.84486 728.16638 ]	[ 723.61911 722.43205 ]
<b>Focal length uncertainty</b>	± [ 2.38296 2.36505 ]	± [ 2.23818 2.23192 ]
<b>Principal point</b> [ $u_0$ $v_0$ ]	[ 326.62643 234.60562 ]	[ 329.24776 234.55332 ]
<b>Principal point uncertainty</b>	± [ 2.57650 1.97176 ]	± [ 2.19256 1.85860 ]
<b>Distortion</b> [ $k_1$ $k_2$ $p_1$ $p_2$ ]	[ -0.12752 0.01808 -0.00097 0.00176 ]	[ -0.13068 0.02705 -0.00147 0.00287 ]
<b>Distortion uncertainty</b>	± [ 0.00359 0.01252 0.00042 0.00060 ]	± [ 0.00266 0.00962 0.00037 0.00051 ]

Table 8.2 Intrinsic parameters of the middle camera

(mm)	Left and middle camera calibration	Middle and right camera calibration
<b>Focal length</b>	[ 734.75323 733.19973 ]	[ 736.99696 735.78218 ]
<b>Focal length uncertainty</b>	± [ 2.41647 2.39100 ]	± [ 2.62135 2.60006 ]
<b>Principal point</b> [ $u_0$ $v_0$ ]	[ 314.71229 232.66709 ]	[ 316.70410 229.73023 ]
<b>Principal point uncertainty</b>	± [ 2.45180 1.98786 ]	± [ 2.66114 2.13031 ]
<b>Distortion</b> [ $k_1$ $k_2$ $p_1$ $p_2$ ]	[ -0.13507 0.00775 -0.00087 0.00122 ]	[ -0.13766 0.00772 -0.00191 0.00238 ]
<b>Distortion uncertainty</b>	± [ 0.00350 0.01195 0.00040 0.00053 ]	± [ 0.00309 0.01056 0.00042 0.00058 ]

Table 8.3 Intrinsic parameters of the right camera

(mm)	Left and right camera calibration	Middle and right camera calibration
<b>Focal length</b>	[ 717.21907 715.47318 ]	[722.57069 720.80392 ]
<b>Focal length uncertainty</b>	± [ 2.27227 2.25533 ]	± [ 2.61329 2.58251 ]
<b>Principal point</b> [ $u_0$ $v_0$ ]	[ 309.66787 230.92801 ]	[309.43864 228.05798 ]
<b>Principal point uncertainty</b>	± [ 2.07494 1.83745 ]	± [ 2.61090 2.07332 ]
<b>Distortion</b> [ $k_1$ $k_2$ $p_1$ $p_2$ ]	[ -0.13740 0.06109 -0.00216 -0.00039 ]	[ -0.13461 0.04621 -0.00266 -0.00006 ]
<b>Distortion uncertainty</b>	± [ 0.00261 0.00936 0.00037 0.00050 ]	± [ 0.00315 0.01101 0.00044 0.00062 ]

Table 8.4 Extrinsic parameters of each camera pair

	Left and middle cameras	Left and right cameras	Middle and right cameras
<b>Translation vector</b> ( $\mathbf{t}_{ML}$ , $\mathbf{t}_{RL}$ , $\mathbf{t}_{RM}$ )	[ -238.05656 -4.25963 -13.06120 ]	[ -446.97074 -3.85081 10.56836 ]	[ -217.29207 -6.69612 -18.79435 ]
<b>Translation vector uncertainty</b>	± [ 2.46538 2.34808 10.88922 ]	± [ 2.56818 2.26584 10.65862 ]	± [ 2.62213 2.50664 11.48718 ]
<b>Rotation matrix</b> ( $\mathbf{R}_{ML}$ , $\mathbf{R}_{RL}$ , $\mathbf{R}_{RM}$ )	$\begin{bmatrix} 0.9998 & 0.0046 & 0.0208 \\ -0.0043 & 0.9999 & -0.0108 \\ -0.0208 & 0.0107 & 0.9997 \end{bmatrix}$	$\begin{bmatrix} 0.9963 & 0.0363 & 0.078 \\ -0.0355 & 0.9993 & -0.0123 \\ -0.0784 & 0.0095 & 0.9969 \end{bmatrix}$	$\begin{bmatrix} 0.9979 & 0.0314 & 0.0573 \\ -0.0314 & 0.9995 & -0.0007 \\ -0.0573 & -0.0011 & 0.9984 \end{bmatrix}$

### **8.3 Determination of the Gravitational Acceleration via the Vision Setup**

In the analysis of saddle motion, it is necessary to obtain the acceleration of the saddle marker by using the position data of the saddle marker. In order to test the accuracy of such a case (where position data is used to obtain acceleration data) using the vision setup, an experiment is designed. In this experiment, an object is tracked with the vision setup during a free fall motion and the acceleration of the object in the gravity direction is computed. Neglecting the effect of air resistance, the resulting calculated acceleration should be equal to  $g$ , i.e.,  $9.81 \text{ m/s}^2$ . The details of the experiment are given below.

The object whose gravitational acceleration is to be found is a tennis ball of diameter 65 mm. It is covered by the 3M<sup>®</sup> reflective material for tracking in order to extract it from the background.

#### **8.3.1 Operations on Images**

In the experiment, the ball is thrown upwards from the lower edge of the calibration volume. Having reached the peak, it freely falls from a height corresponding to the upper edge. A total of 21 frames (corresponding to 0.7 seconds) are obtained from each camera. Only area thresholding and windowing operations are applied to the frames with an area threshold of [10, 35] pixels and with a window size of 12x20 pixels. 10 images of the original and filtered images are shown in Figure 8.3, as viewed from the left camera. Note that the images are cropped to give more details in the figure. Using the procedure described in chapter 4, the image coordinates of the ball center in each of the camera reference frames are obtained.

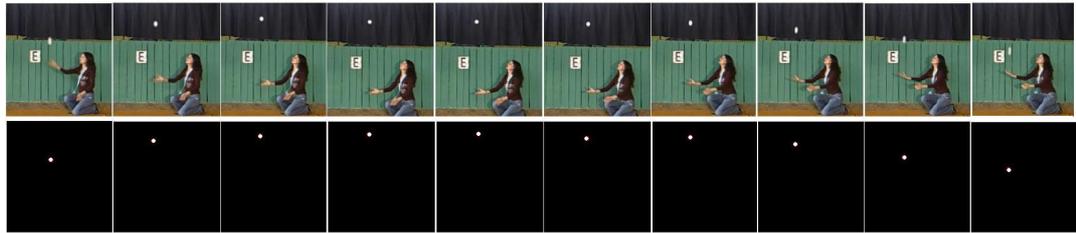


Figure 8.3 The original and area filtered left camera images of the ball

Since the acceleration in the gravity direction is required; the unit vector showing the gravity direction is to be found. To that purpose, the images of a plumb are extracted and processed. A sample image as viewed from the left camera is given in Figure 8.4.



Figure 8.4 The original and filtered left camera image of the plumb

### 8.3.2 Operations in 3D

The position of the ball in the left camera reference frame is found with the approach explained in Chapter 5. Since only one visible point is tracked, there is no need to apply the rigidity or the invisible marker tracking algorithms. The 3D coordinates of the ball in the left camera frame is represented by  $CM_L$ . The position of the ball

along the gravity direction is obtained by taking the dot product of the position vector of the ball with the gravity direction (obtained via the plumb). Therefore, the position of the ball in the gravitational direction is obtained ( $\mathbf{CM}_g$ ). The curve to be fit to the position data is chosen as a quadratic polynomial, since the acceleration (obtained by differentiating the position twice) is expected to be constant. The results of the curve fitting and the derived velocity and acceleration data are presented in Figure 8.5. In this application, the gravitational acceleration is obtained to be approximately  $9.5 \text{ m/s}^2$ , with an error of 3.6 %.

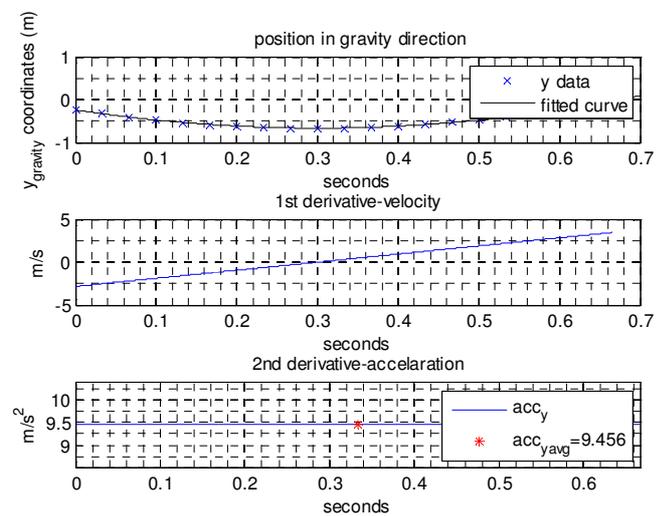


Figure 8.5 Motion of the ball in gravity direction

Having calculated the errors involved in the acceleration, obtained from position data, errors in the position data itself are also calculated. In order to perform this, various points on the calibration grid are considered and the distances between these points are measured with the vision system. The errors between the distances obtained using the vision system and the actual distances are compared. For instance, in Figure 8.6, 4 points (A, B, C, D) are shown on the calibration grid. In Table 8.5, the measured and actual distances are shown with the corresponding percent errors.

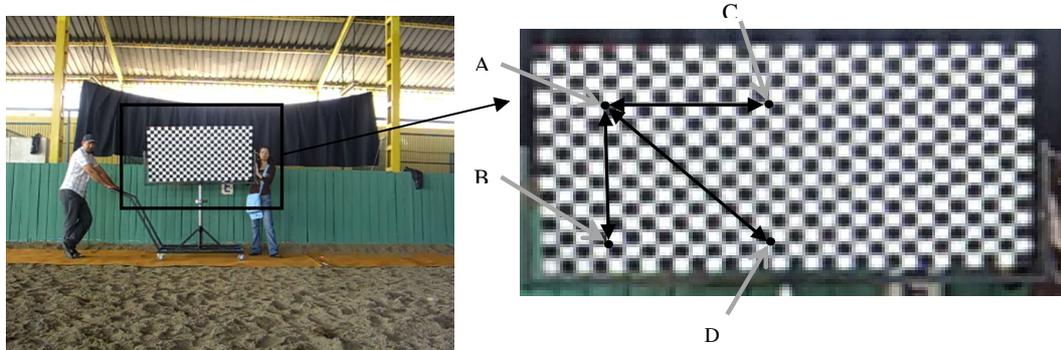


Figure 8.6 The points between which the distances are calculated

Table 8.5 The comparison of the actual and measured distances for the executed experiments

	<b>Actual distance (mm)</b>	<b>Measured distance (mm)</b>	<b>Percent error in distance (%)</b>
AB	500	489.1015	2.1797
AC	500	522.9729	4.5946
AD	707.1	726.5618	2.7514

## 8.4 Determination of the Kinematics of Horse Saddle

### 8.4.1 Detailed Application to Reach Kinematics of Horse Saddle

As a case study, the saddle motions of four horses in four different gait styles have been investigated using the developed vision system. In order to illustrate the details of the involved steps, the whole process used to reach the kinematic data of one horse (Horse 2) in one gait (canter) will be presented in detail. There are four horses with four riders who took part in the experiments. The horses are labeled as Horse 1, Horse 2, Horse 3 and Horse 4; whereas the riders are named as A, B, C and D.

Table 8.6 The horses features taking part in the experiments

	<b>Horse 1</b>	<b>Horse 2</b>	<b>Horse 3</b>	<b>Horse 4</b>
<b>Age</b>	20	10	19	13
<b>Breed</b>	German	Dutch	French	English
<b>Height at withers (m)</b>	1.70	1.73	1.58	1.57

Firstly, the markers' center coordinates, in the image reference frames must be obtained. The movie files obtained from the left, middle and right cameras are converted to image files. The files have to be in sequence and put in the necessary directory. There are 38 images (frames) that cover two periods of the motion. The left camera image centroids are found as described below:

While running `left_level_info_from_all_images_1.m`; the RGB to gray level threshold is entered as `left_levelinput= 115`. The 3 markers are checked such that all of them are visible and isolated from the background during the whole motion. Then, using the first image, necessary inputs are entered by running

left\_data\_from\_first\_image\_2.m. The RGB and black & white versions of the first image is shown in Figure 8.7.



Figure 8.7 RGB and black & white versions of the first image

The inputs which are entered are shown below:

```
left_labels= [598 607 613]
left_area_threshold_min= 15
left_area_threshold_max= 240
left_window_dx= 84
left_window_dy= 40
left_distance_percent_error= 0.09
```

The results of the filtering operations are shown in Figure 8.8:

where

- i) Area filtered image
- ii) The regions in the pre-defined window.
- iii) The (desired) markers and their centers, after application of the distance filter

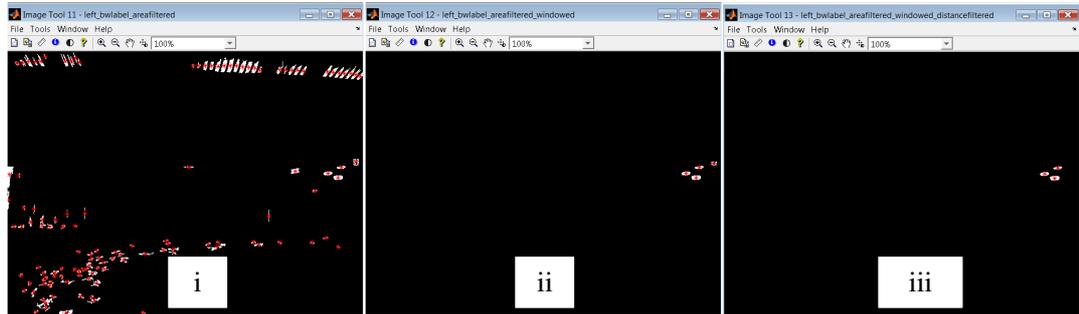


Figure 8.8 Filtering on the first image

Therefore, the centers of the markers are obtained in the first image. Using the outputs of this step, one can proceed to the next stage. Then, only by running the file `left_image_centroids_3.m` one obtains the results of the filtering operations for the totality of the left images. Similar operations are performed for the middle and right sequence of images. After these operations, the markers center coordinates are obtained for the 3 cameras. Next, this data will be combined with the calibration parameters to reach the markers locations with respect to the left camera reference frame. The components of the position vectors of the markers are presented in Figure 8.9 (a). A smoothing operation is performed for each component. Furthermore, value for the period (`period= 18`) and the title for the figures (`titles= 'Horse2CCWcenter'`) are entered. In Figure 8.9 (b), only the first marker's smoothed results are shown.

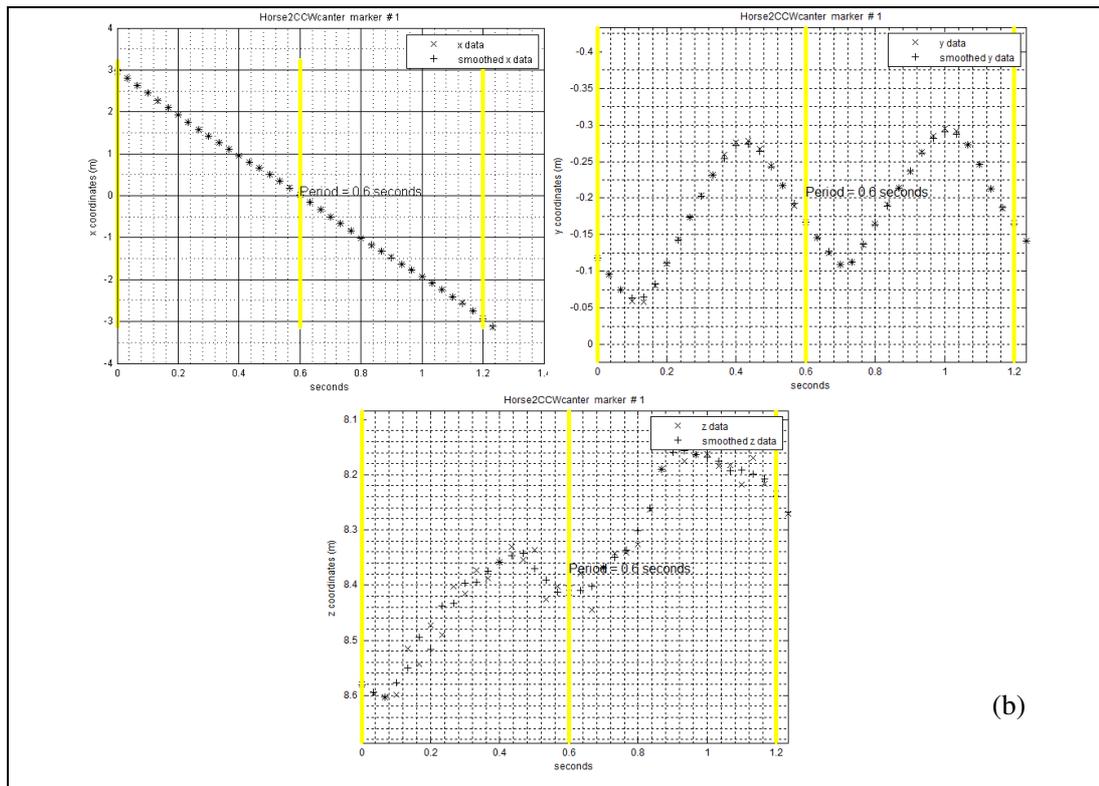
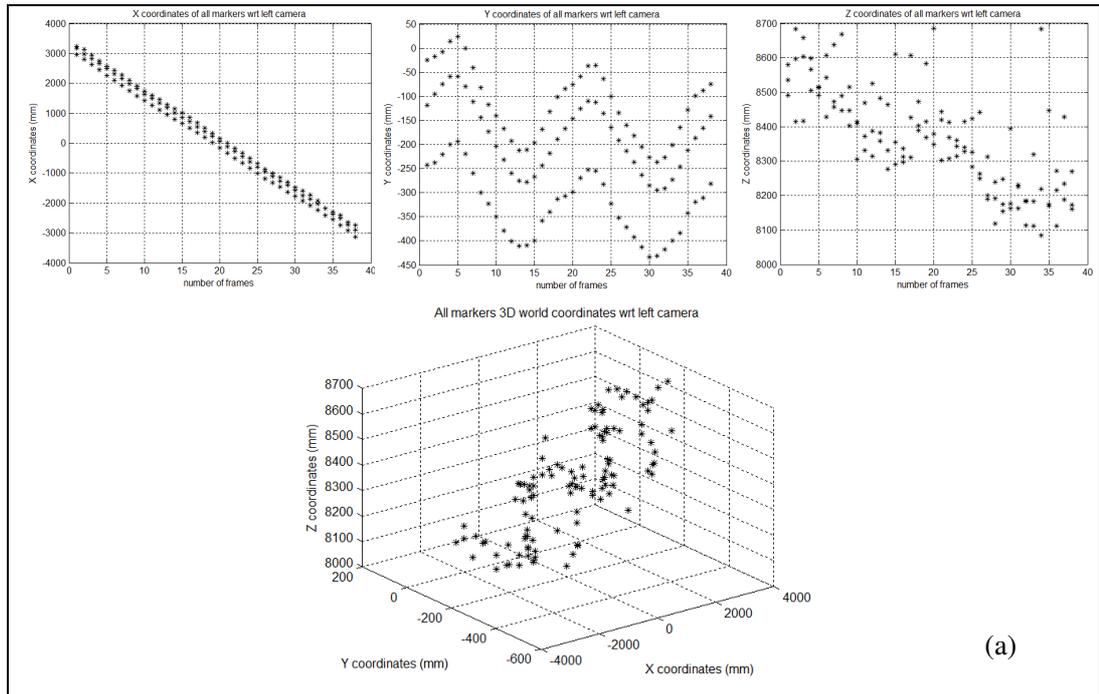


Figure 8.9 (a) Position of the markers (b) Smoothed results of the 1<sup>st</sup> marker

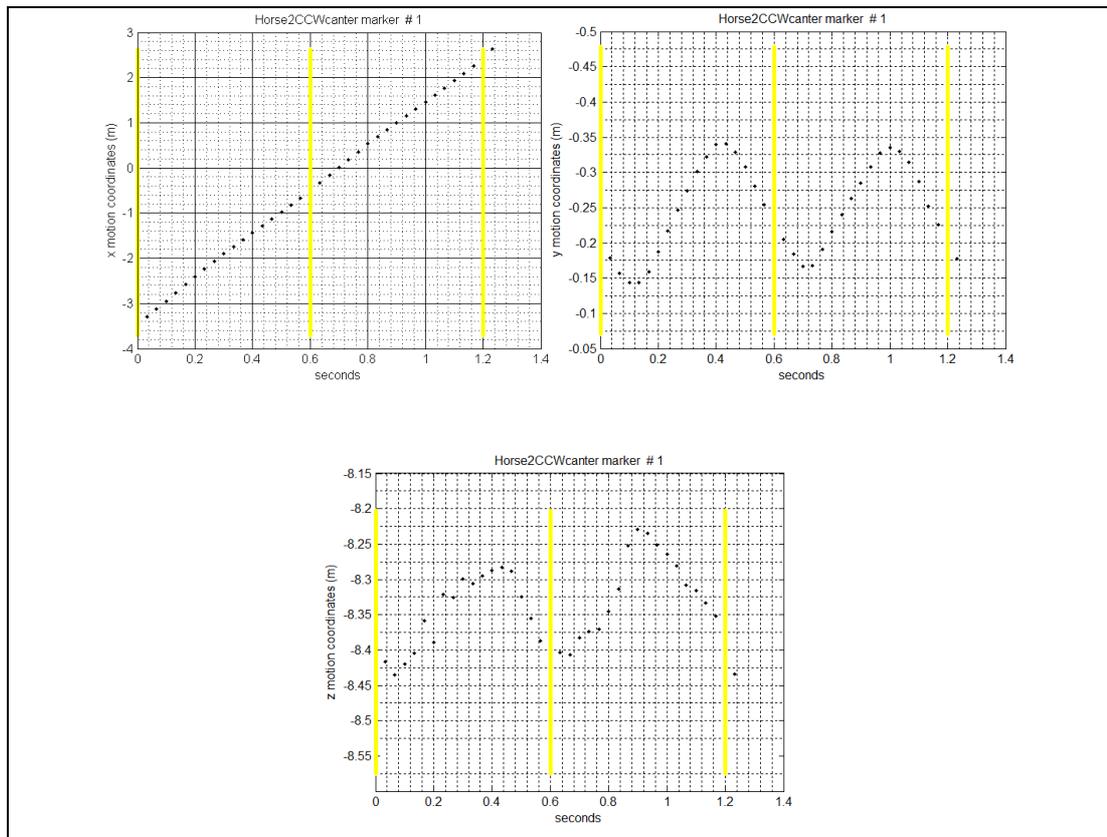


Figure 8.10 Position of 1<sup>st</sup> marker in motion reference frame

In the next step, the reference frame is changed from the camera to the motion reference frame. As seen in Figure 8.10, the plots in the motion frame (provided here for only marker 1) are more useful than the camera frame plots.

The assumption of rigidity for the three markers will be satisfied in this stage with the registration algorithm. First of all, the image centers of the markers are obtained from the three cameras. The 3 images from the 3 cameras are put in the required directory. By running `one_image_centroids.m`, the left image of the horse in the static position opens. The necessary inputs as `left_levelinput=115,`

`left_labels=[332 346 353 ]` are entered. Then, the middle and right images are processed, similarly. The images of this procedure are given in Figure 8.11.



Figure 8.11 Static shot of the markers (in left camera)

Next, the 3D static position of the markers in the motion frame is obtained (corresponding to the model points). The coordinates before and after the registration are shown to the user for the whole motion. As an example, the 27th frame is provided in Figure 8.12. Therefore, the distances between markers remain the same (satisfying the rigid body constraint) during the whole motion.

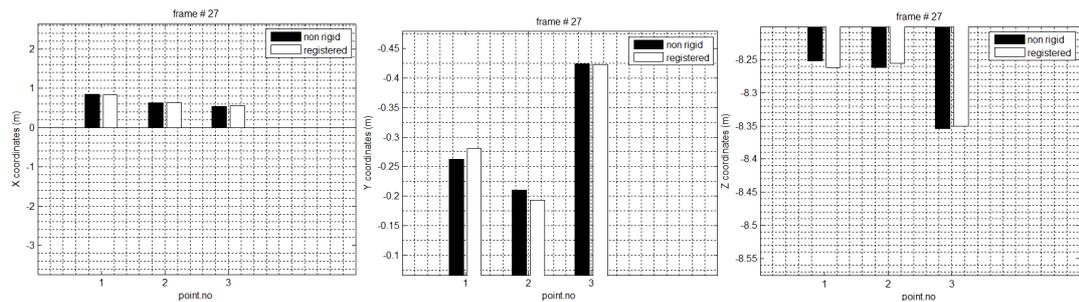


Figure 8.12 Coordinates before and after rigid body correction

In order to reach the velocity and acceleration of each marker, a curve fitting operation is performed on the recently obtained position data. However, after applying curve fitting to the position data, the rigid body constraints will be deteriorated, which is undesired and insensible. In the following stages, original position data obtained after registration is retained, while the velocity and acceleration data is obtained by differentiating the curve fitted position. For the x component of the data, the curve fitting process is as follows. Firstly, the user is shown the unfitted data. Then, the appropriate curve type (`fit_type_x`) is expected from the user (A suitable fit can be chosen from the MATLAB library or the recommended fit (linear) is applied.). In the fitting process for the y component of the data, the unfitted data is provided to the user firstly and the number of extremes is required. For instance, by entering `no_of_extreme=4` for this example, two choices of Fourier fits are provided (Fourier2 and Fourier3). In this example `fit_choice=1` (Fourier2) has been selected by observing the graphs and the numerical goodness of the fit data provided on the screen. Finally, while fitting a curve to the z component of the data, `no_of_extreme=2` (although it seems to be 4) is entered, since it seems to be more appropriate for the data. Two choices of Fourier fits are then provided (Fourier1 and Fourier2) and `fit_choice=1` (Fourier1) is entered. For illustrative purposes, the first marker's data is presented in Figure 8.13. All three markers are processed similarly.

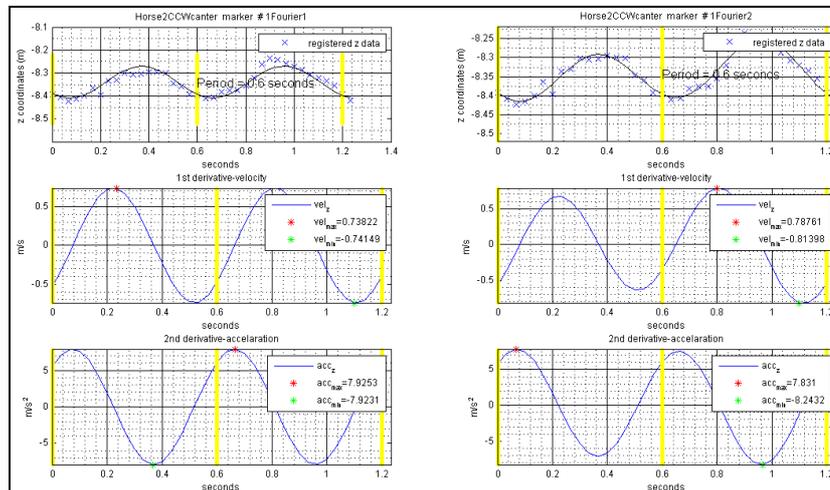
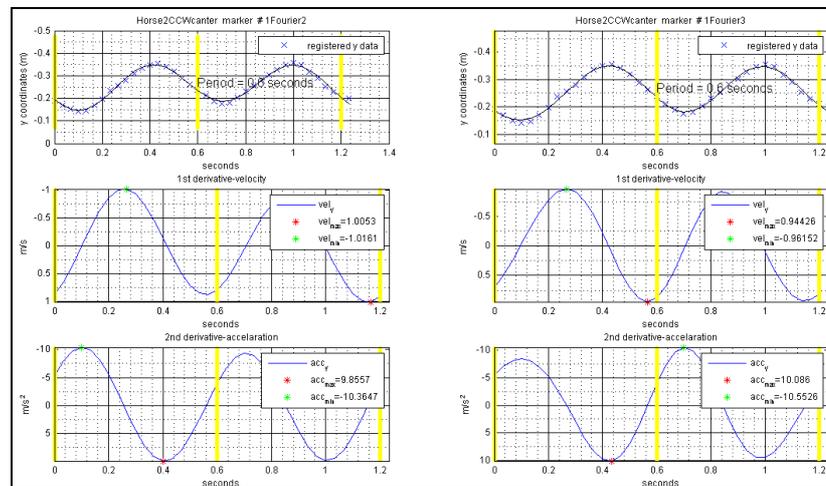
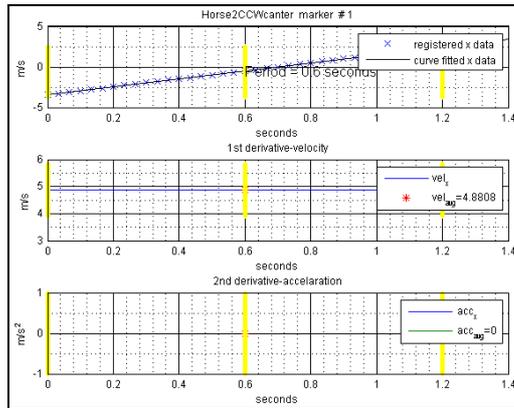


Figure 8.13 Curve fitted kinematic data of 1<sup>st</sup> marker

In the next step, the angular velocity and acceleration of the rigid body, consisting of the 3 visible markers and 1 visible (saddle) marker are calculated. The three components of the original and smoothed angular velocity data (in the six versions) plots are presented in Figure 8.14, Figure 8.15 and Figure 8.16. All smoothed versions of omega and the average of the 6 versions are presented in Figure 8.17. The user is asked the versions of omega that are to be used in the determination of the final angular velocity. In this experiment, the questions asked by the software and the corresponding answers are given below:

```
Is omega1_ver1 to be in the calculation of final omega? y/n: y  
Is omega2_ver1 to be in the calculation of final omega? y/n: y  
Is omega1_ver2 to be in the calculation of final omega? y/n: n  
Is omega2_ver2 to be in the calculation of final omega? y/n: y  
Is omega1_ver3 to be in the calculation of final omega? y/n: n  
Is omega2_ver3 to be in the calculation of final omega? y/n: y
```

The overall omega is calculated as the mean of the selected versions. The resultant omega for this example is shown in Figure 8.18.

The angular acceleration is obtained in a similar manner. All smoothed versions of alfa and the average of them are presented in Figure 8.19. The suitable versions are selected by the user leading to the final value of alfa, shown in Figure 8.20.

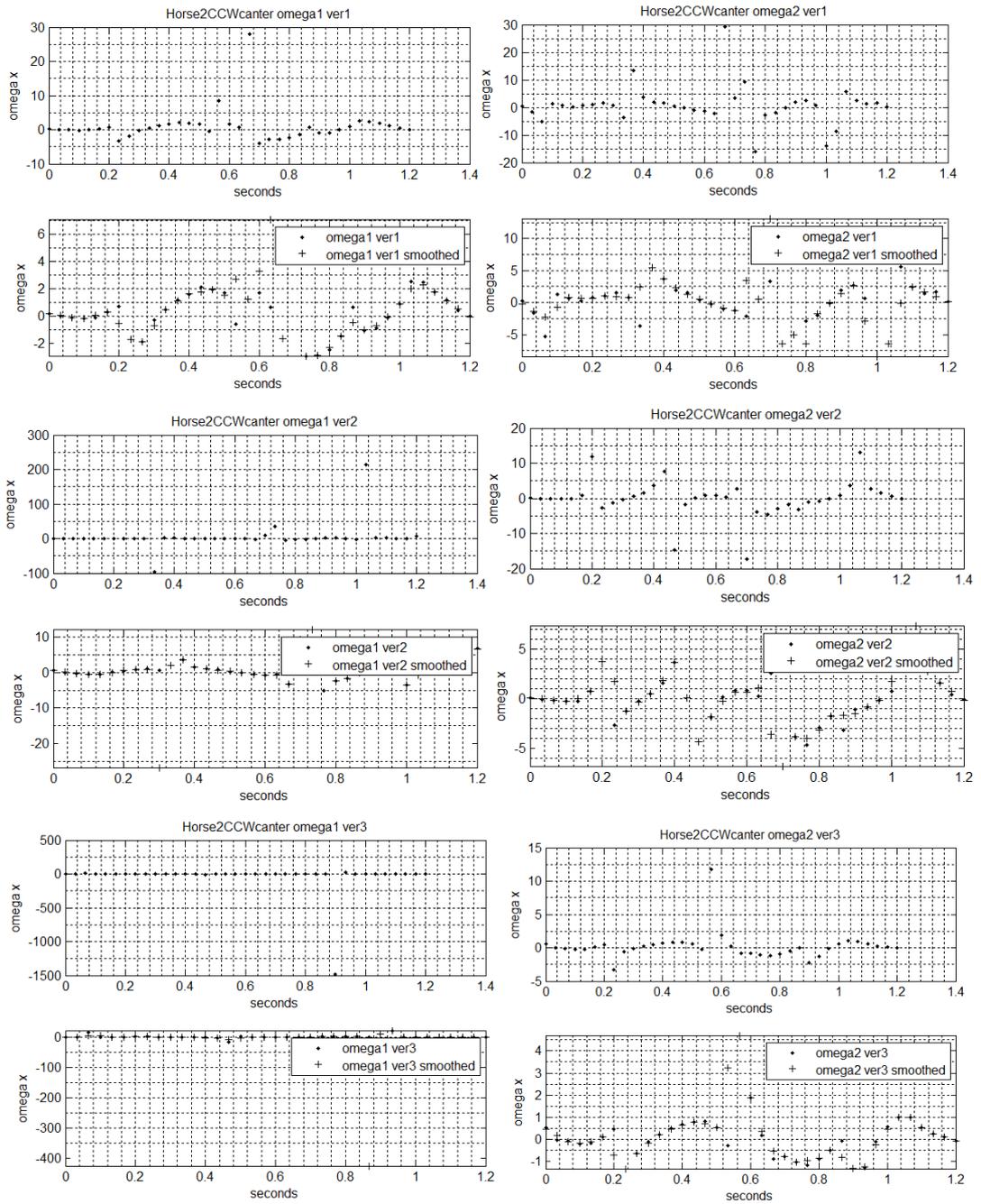


Figure 8.14 All versions of omega around the x axis

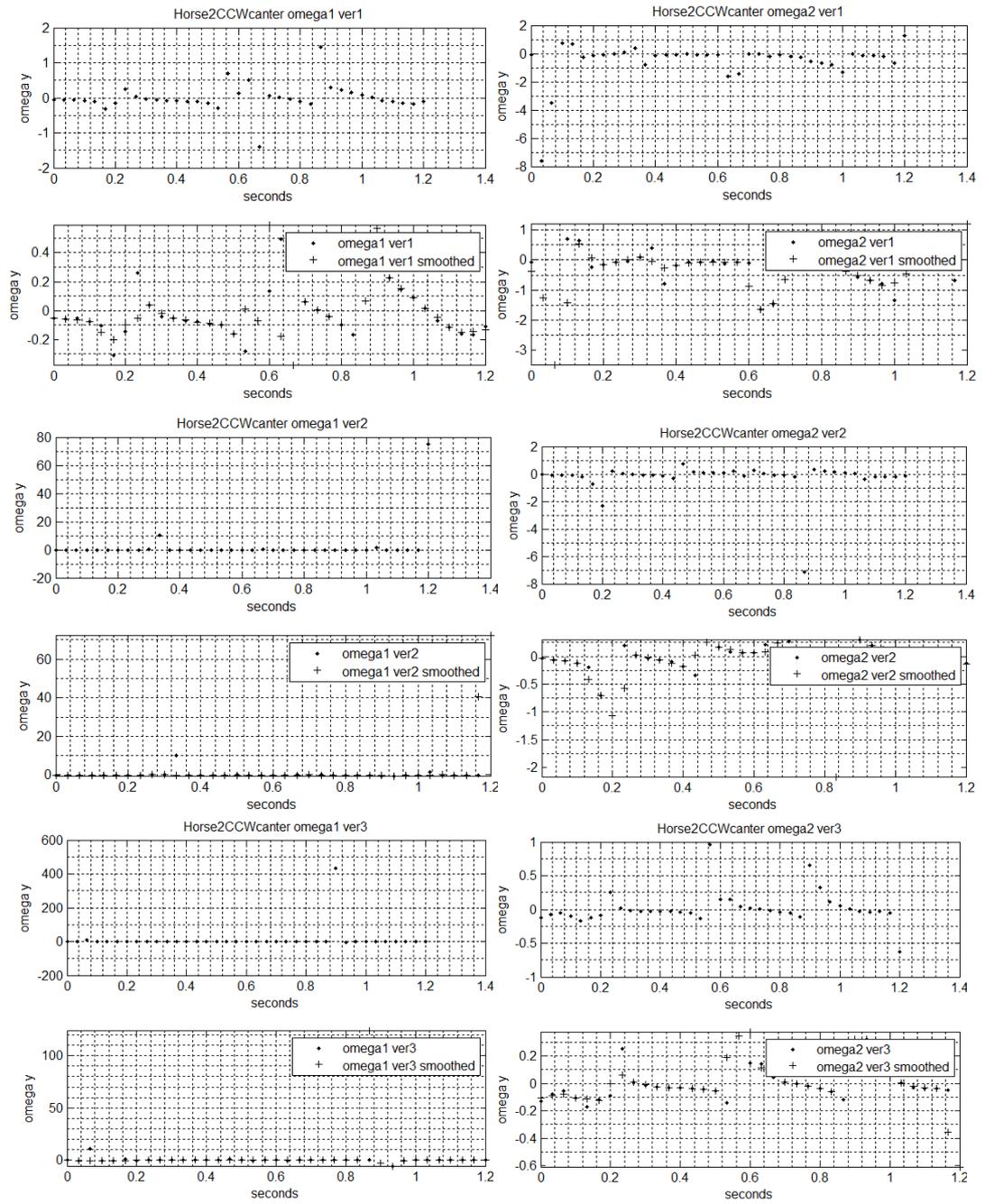


Figure 8.15 All versions of omega around the y axis

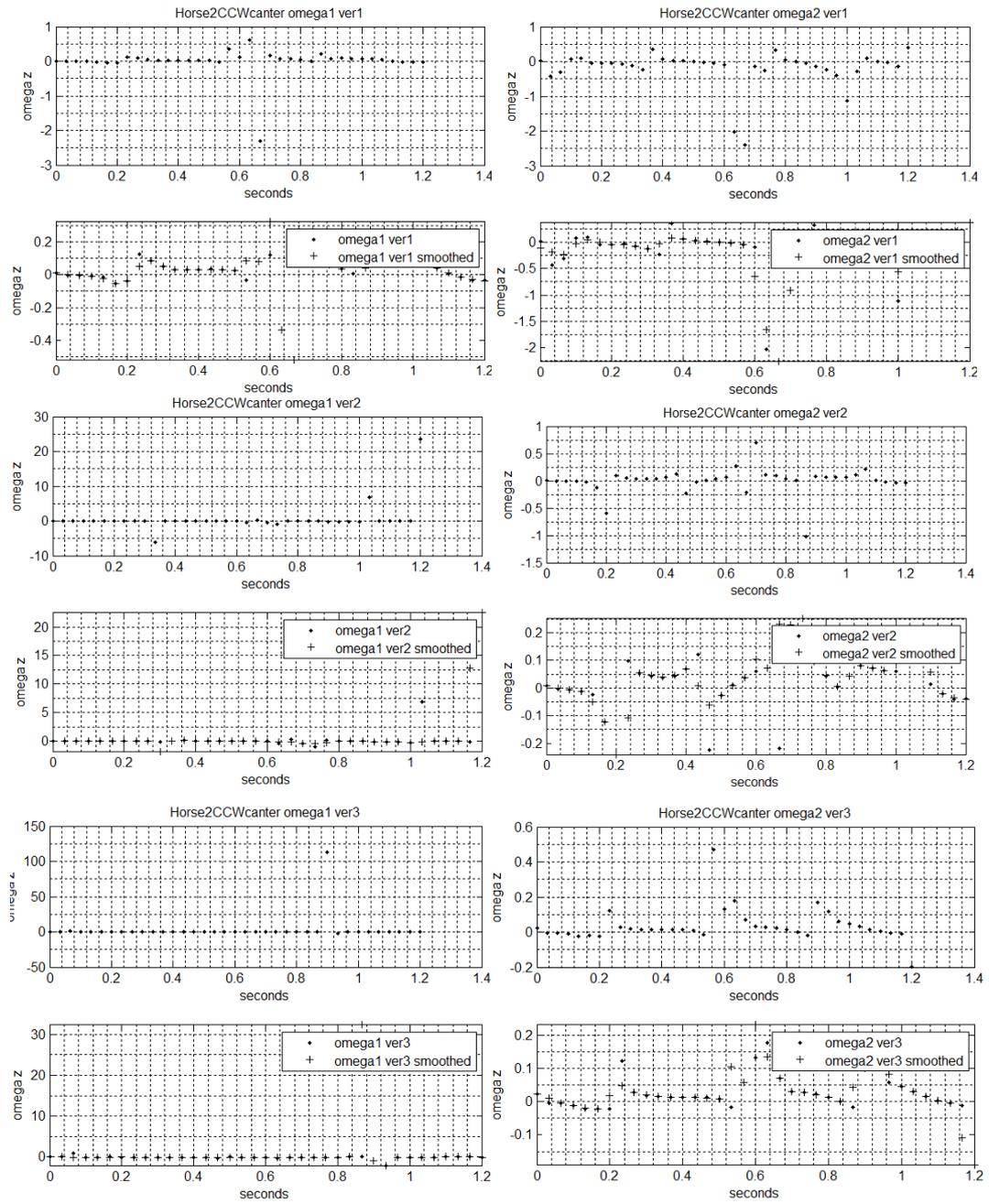


Figure 8.16 All versions of omega around the z axis

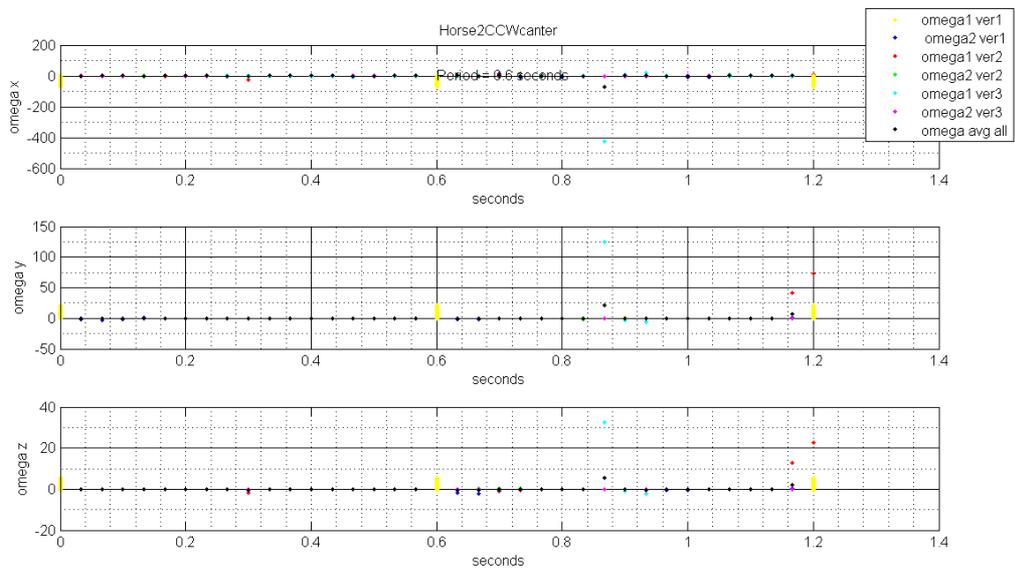


Figure 8.17 All smoothed versions and average of omega

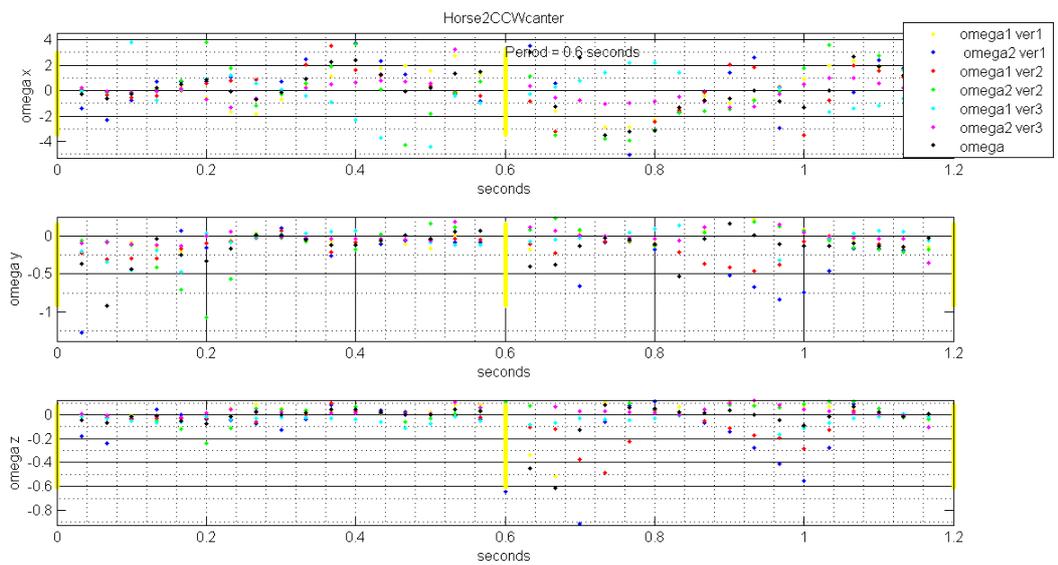


Figure 8.18 The resultant omega

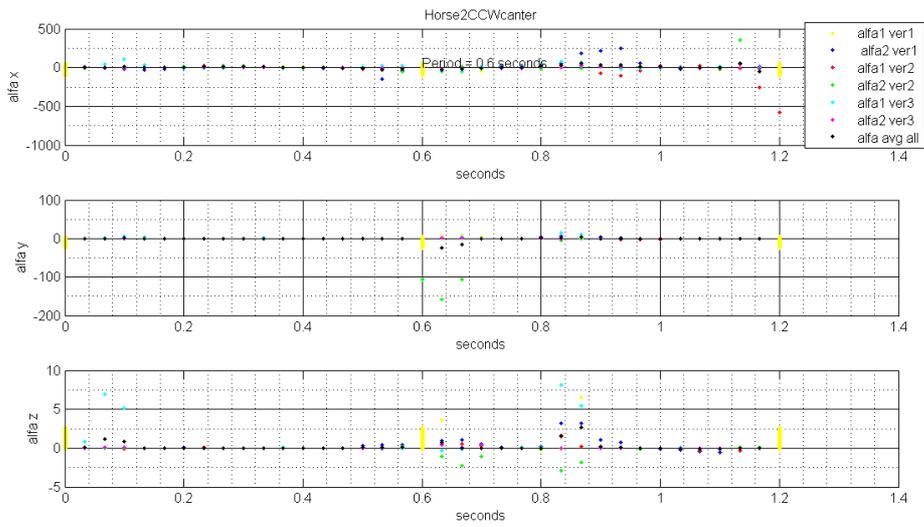


Figure 8.19 All smoothed versions and average of alfa

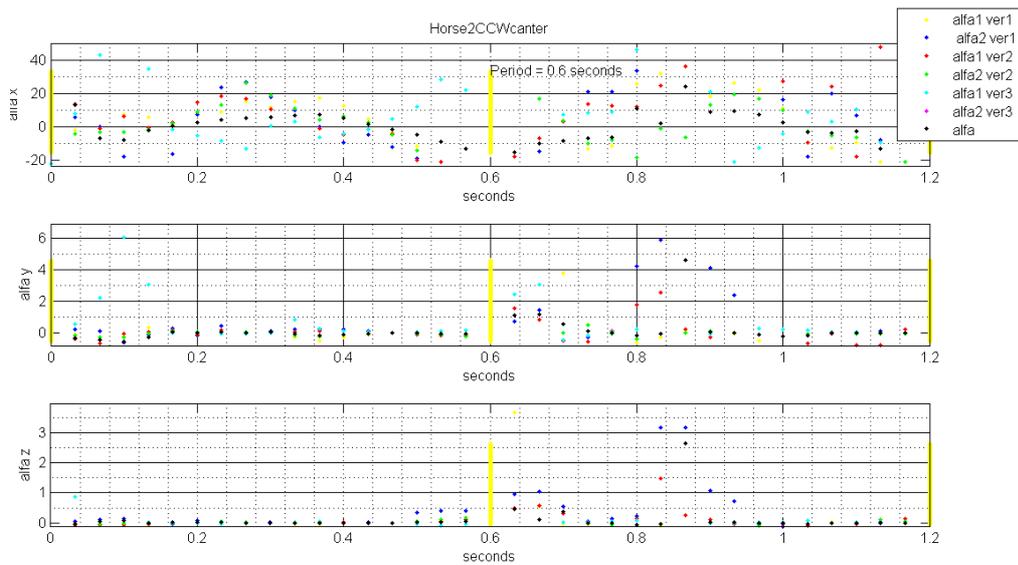


Figure 8.20 The resultant alfa

The angular velocity and accelerations obtained in this example are not very reliable due to 2 main reasons. The first source of error is due to the simple and cheap, but unsophisticated cameras (with a low frame rate and resolution) used in the vision system. The main source of error, however, is that the angular velocity of the body

(which is roughly parallel to the x-motion axis) is approximately perpendicular to the normal plane formed by the 3 visible markers (which is roughly parallel to the z-motion axis). Therefore, the angular velocity and acceleration are calculated in a configuration which is very close to the singularity (see chapter 6) which, naturally, makes the results quite unreliable. In order to obtain the angular velocity and acceleration in a reliable manner (which is of secondary importance in this study), the locations of the 3 markers should be changed.

After reaching the data of the visible markers, the data for the invisible saddle marker (during riding) attached to the saddle is to be found. Firstly, the images from the three static camera views have to be processed (as in the registration step). The user is shown the images of the static horse viewed in the left camera (see Figure 8.21) and the following data is entered as inputs: `left_levelinput= 115`, `left_labels= 325`. Similarly, the marker is identified from the middle and right camera and the 3D coordinates of the saddle marker is obtained in the motion frame. The position of the invisible saddle marker during the motion can now be found with the procedure explained in chapter 6. The position of the saddle marker is depicted in Figure 8.22.

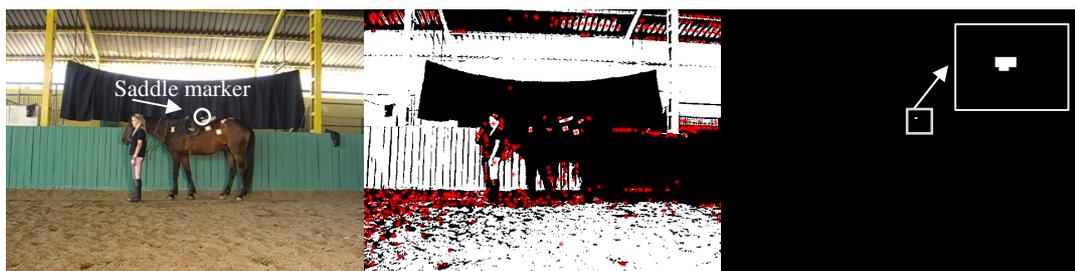
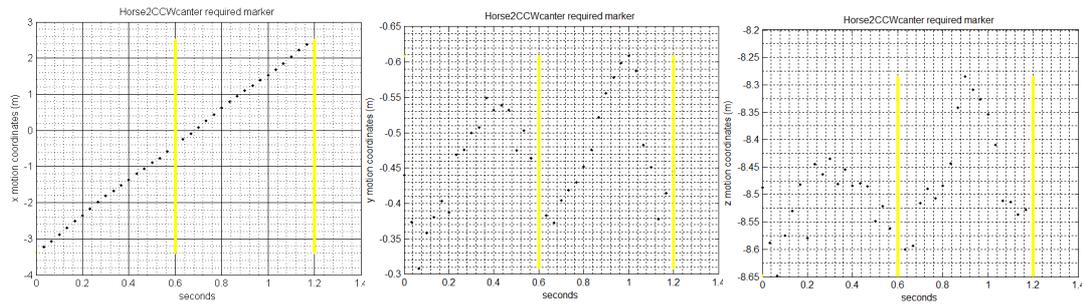
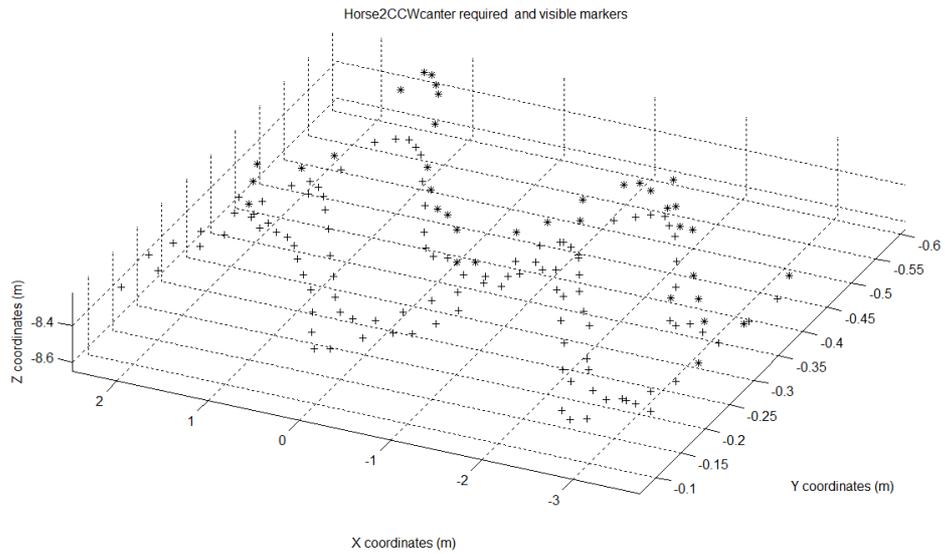


Figure 8.21 Saddle marker viewed from the left camera



(a)



(b)

Figure 8.22 Position of (a) saddle marker (b) visible and saddle markers

Having obtained the position, in the subsequent part, the velocity and acceleration of the saddle marker are obtained with the three different approaches explained in chapter 6. It can be clearly seen in Figure 8.23 that the three approaches lead to similar results. The average of the three solutions is taken as the resultant value of velocity and acceleration (of the required, invisible saddle marker). The resultant plots are shown in Figure 8.24.

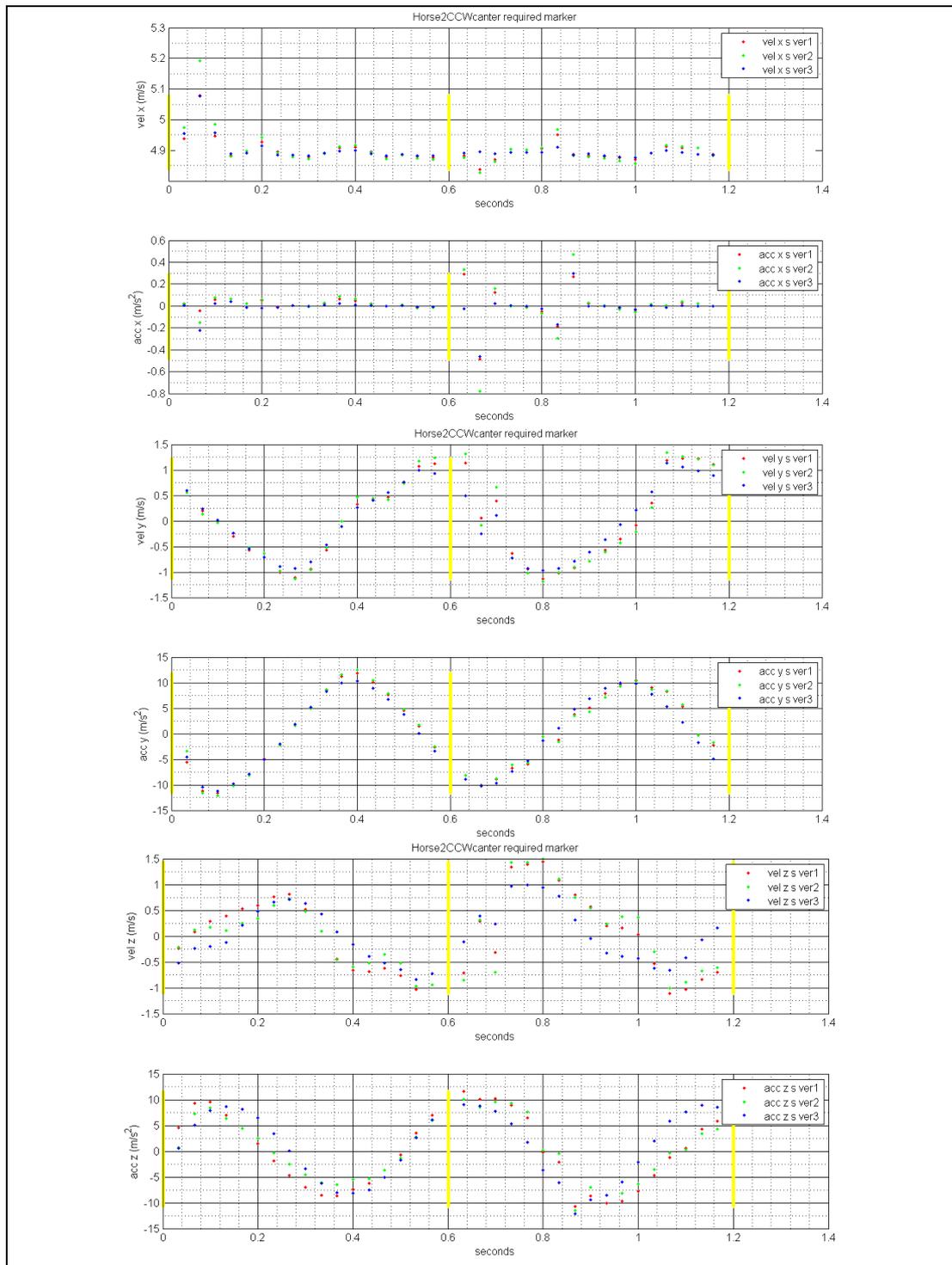


Figure 8.23 Velocity and acceleration of the invisible saddle marker

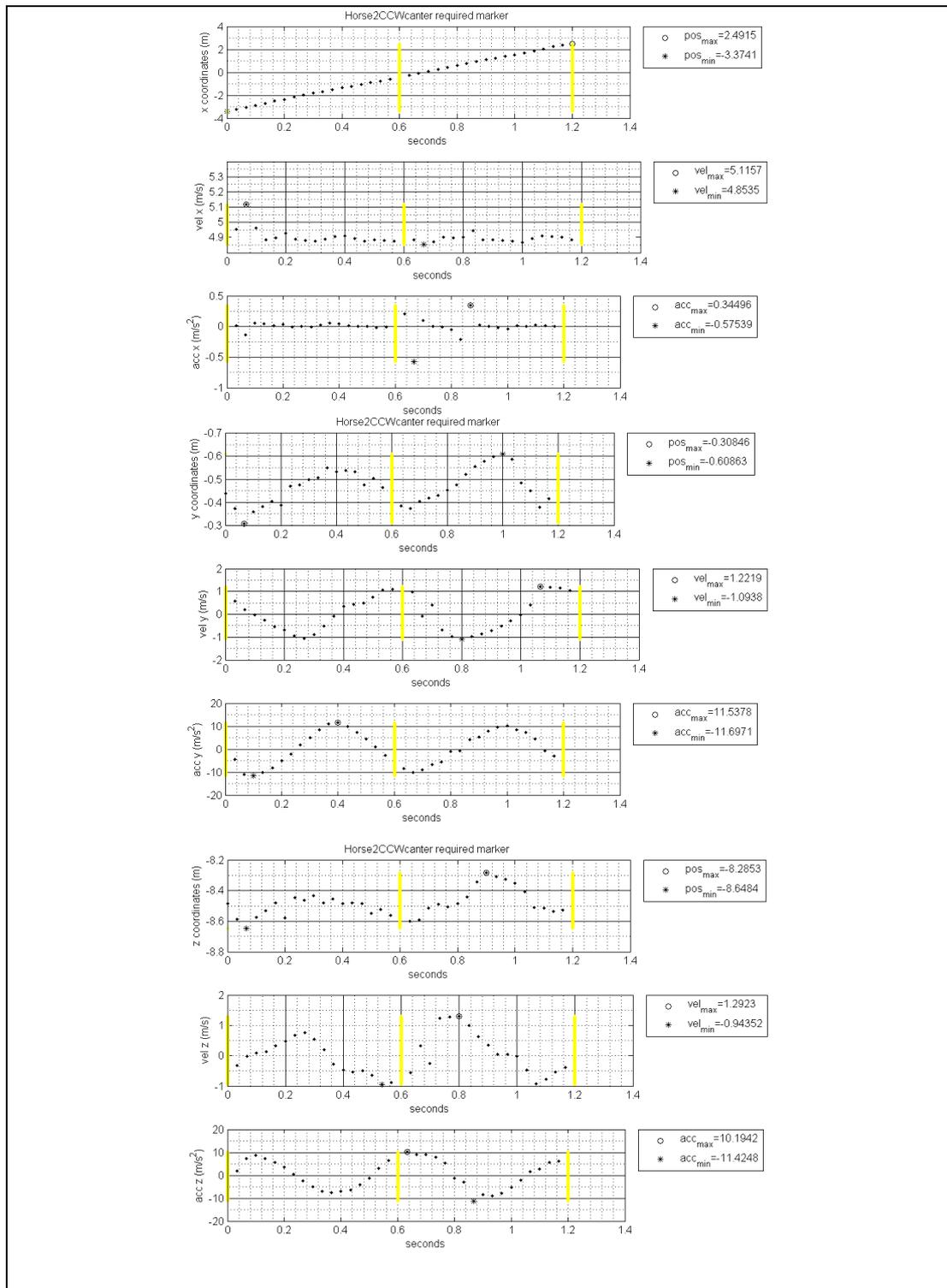


Figure 8.24 Kinematic data of the invisible saddle marker

By running the file `performance_measure_required_markers.m`, the horse comfort indicators are obtained (see chapter 7 for details). In Table 8.7 the results are tabulated for horse 2 when it is cantering. The outputs, which are the horse comfort indicators, may be saved in a format which is compatible with EXCEL®.

Table 8.7 Various horse comfort indicators

<code>acc_y_Wb_rms_s_Horse2CCWcanter</code> (m/s <sup>2</sup> )	3.1906
<code>acc_y_Wk_rms_s_Horse2CCWcanter</code> (m/s <sup>2</sup> )	3.7007
<code>VDV_y_Wb_s_Horse2CCWcanter</code> (m/s <sup>1.75</sup> )	3.7932
<code>VDV_y_Wk_s_Horse2CCWcanter</code> (m/s <sup>1.75</sup> )	4.3997
<code>acc_y_max_s_Horse2CCWcanter</code> (m/s <sup>2</sup> )	11.6971
<code>crest_factor_y_s_Horse2CCWcanter</code>	1.6245
<code>TA_8_action_Wb_s_Horse2CCWcanter</code> (hr)	0.1965
<code>TA_8_action_Wk_s_Horse2CCWcanter</code> (hr)	0.1460
<code>TA_8_limit_Wb_s_Horse2CCWcanter</code> (hr)	1.0393
<code>TA_8_limit_Wk_s_Horse2CCWcanter</code> (hr)	0.7725
<code>TVDV_T_action_Wb_s_Horse2CCWcanter</code> (hr)	0.0110
<code>TVDV_T_action_Wk_s_Horse2CCWcanter</code> (hr)	0.0061
<code>TVDV_T_limit_Wb_s_Horse2CCWcanter</code> (hr)	0.3131
<code>TVDV_T_limit_Wk_s_Horse2CCWcanter</code> (hr)	0.1730
<code>Distance_TA_8_action_Wb_s_Horse2CCWcanter</code> (km)	3.4653
<code>Distance_TA_8_action_Wk_s_Horse2CCWcanter</code> (km)	2.5758
<code>Distance_TA_8_limit_Wb_s_Horse2CCWcanter</code> (km)	18.3313
<code>Distance_TA_8_limit_Wk_s_Horse2CCWcanter</code> (km)	13.6259
<code>Distance_TVDV_T_action_Wb_s_Horse2CCWcanter</code> (km)	0.1947
<code>Distance_TVDV_T_action_Wk_s_Horse2CCWcanter</code> (km)	0.1076
<code>Distance_TVDV_T_limit_Wb_s_Horse2CCWcanter</code> (km)	5.5228
<code>Distance_TVDV_T_limit_Wk_s_Horse2CCWcanter</code> (km)	3.0514
<code>vel_x_avg_s_Horse2CCWcanter</code> (km/hr)	17.6376
<code>amplitude_y_s_Horse2CCWcanter</code> (m)	0.3002

#### **8.4.2 Saddle Marker Data of Four Horses in Each Gait**

In Figure 8.25, the x-motion, y-motion and z-motion coordinates of the position and acceleration vectors of the invisible saddle marker are shown for 4 horses while they are walking. Figure 8.26 to Figure 8.28 depict similar data for the cases of sitting trot, rising trot and canter, respectively.

It should be noted that in each figure, the graphs in columns 1, 2 and 3 correspond to the x-motion, y-motion and z-motion data, respectively. Similarly, rows 1, 2, 3 and 4 correspond to horse 1, horse 2, horse 3 and horse 4, respectively.

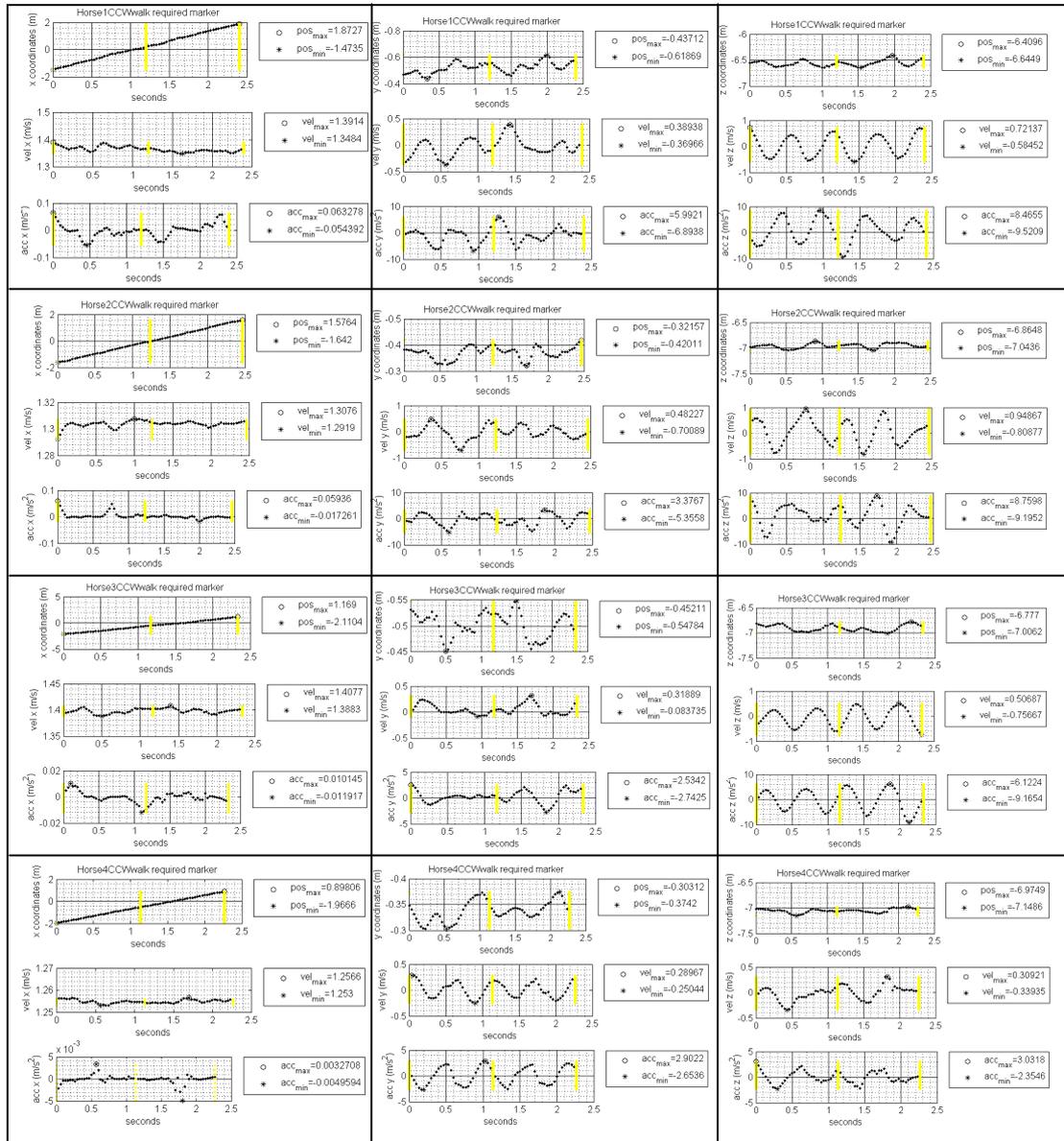


Figure 8.25 Walk data of the saddle marker for the 4 horses



Figure 8.26 Sitting trot data of the saddle marker for the 4 horses

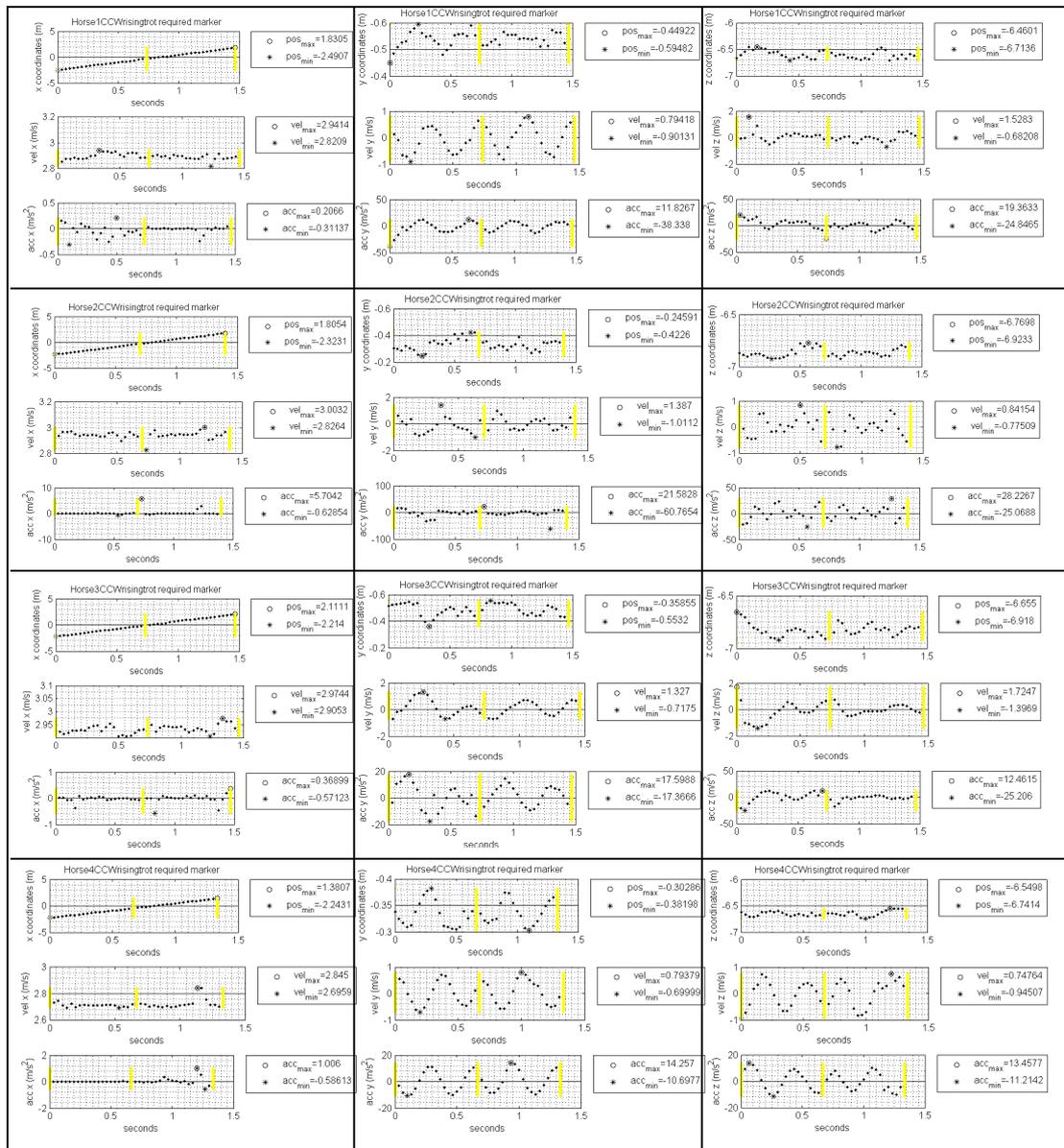


Figure 8.27 Rising trot data of the saddle marker for the 4 horses



Figure 8.28 Canter data of the saddle marker for the 4 horses

## 8.5 Comparison of Horse Comfort Levels

The 8 comfort indicators (that may be used to evaluate horses) in this study are  $acc\_y\_max$ ,  $acc\_y\_Wb\_rms$ ,  $crest\_factor$ ,  $Distance\_TA\_8\_limit\_Wb$ ,  $TA\_8\_limit\_Wb$ ,  $VDV\_y\_Wb$ ,  $vel\_x\_avg$  and  $amplitude\_y$ . In these indicators ISO standards, rather than British standards have been preferred (i.e.  $W_b$  is preferred over  $W_k$ ). While using these indicators to evaluate a given horse, the following comments should be taken into account.

- The indicators may not be as accurate as desired due to the errors arising from the low frame rate and resolution of the simple and cheap commercial cameras used in the vision setup.
- Other than  $vel\_x\_avg$ , all indicators assume that the rider is in continuous contact with the saddle (as if he was rigidly connected to the saddle). This assumption is not true for the rising trot. Therefore, although rising trot is also considered in Figure 8.29 - Figure 8.32, this specific gait should be somewhat disregarded in these plots.
- The two indicators  $Distance\_TA\_8\_limit\_Wb$  and  $TA\_8\_limit\_Wb$  are extremely over-safe since they are based upon the exposure limit values specified by the EU Physical Agents Directive (see Table 7.2). The aforementioned limit values assume that the person subjected to vibrations is totally passive (in regard to taking any counter measures against vibrations). In the case of riding, however, the rider somewhat compensates for the induced vibrations. For instance, in sitting trot, the rider moves his belly like a belly dancer. In the case of canter, on the other hand, the rider moves as if he is swinging. Therefore, the values of the indicators  $Distance\_TA\_8\_limit\_Wb$  and  $TA\_8\_limit\_Wb$  should be interpreted to be “relative” comfort indicators, rather than absolute limits. For instance, let  $TA\_8\_limit\_Wb$  values of horses A and B

1.5 hours and 3 hours, respectively. In this case, one should think of horse B to be  $(3 \div 1.5 = 2)$  times more comfortable than horse A.

### **8.5.1 Different Gaits of the Same Horse**

Referring to Figure 8.29, it is observed that, for all horses maximum acceleration value is higher during trot and canter than walk; and rms acceleration increases as walking style changes from walk to sitting trot and sitting trot to canter. However, referring to Figure 8.30, each horse has a relatively different behavior in the crest factor value. Furthermore, it is noticed that the allowable distance that can be travelled is much higher in the walking style compared to the other styles, while trot and canter shows changes. Similarly, the time that limits the rider has a much greater values in walk than the other gaits, while it shows changes in trot and canter (Figure 8.31). VDV value of acceleration is lower in walk, but it varies in trot and canter.

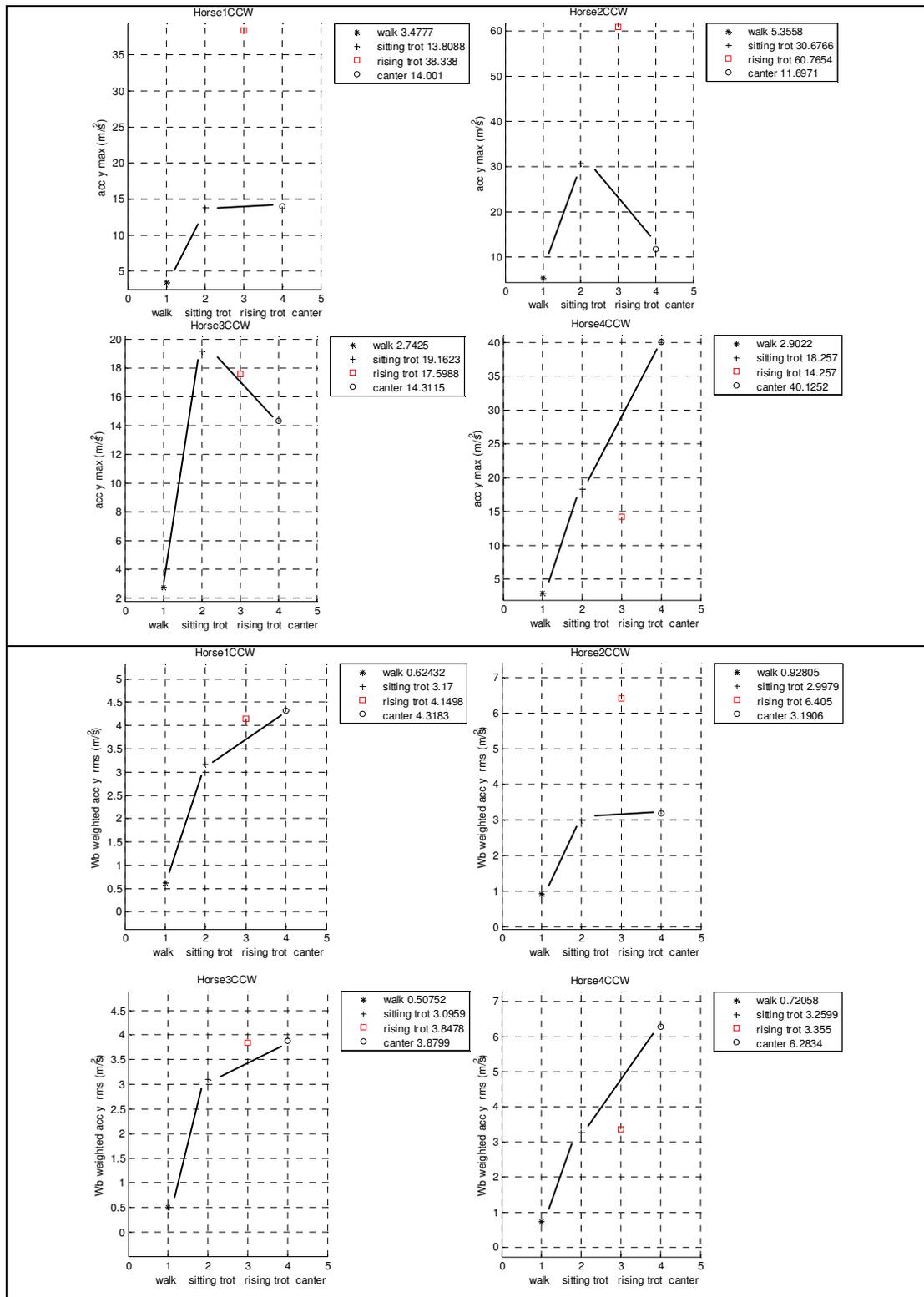


Figure 8.29 acc\_y\_max, acc\_y\_Wb\_rms versus gait style for 4 different horses

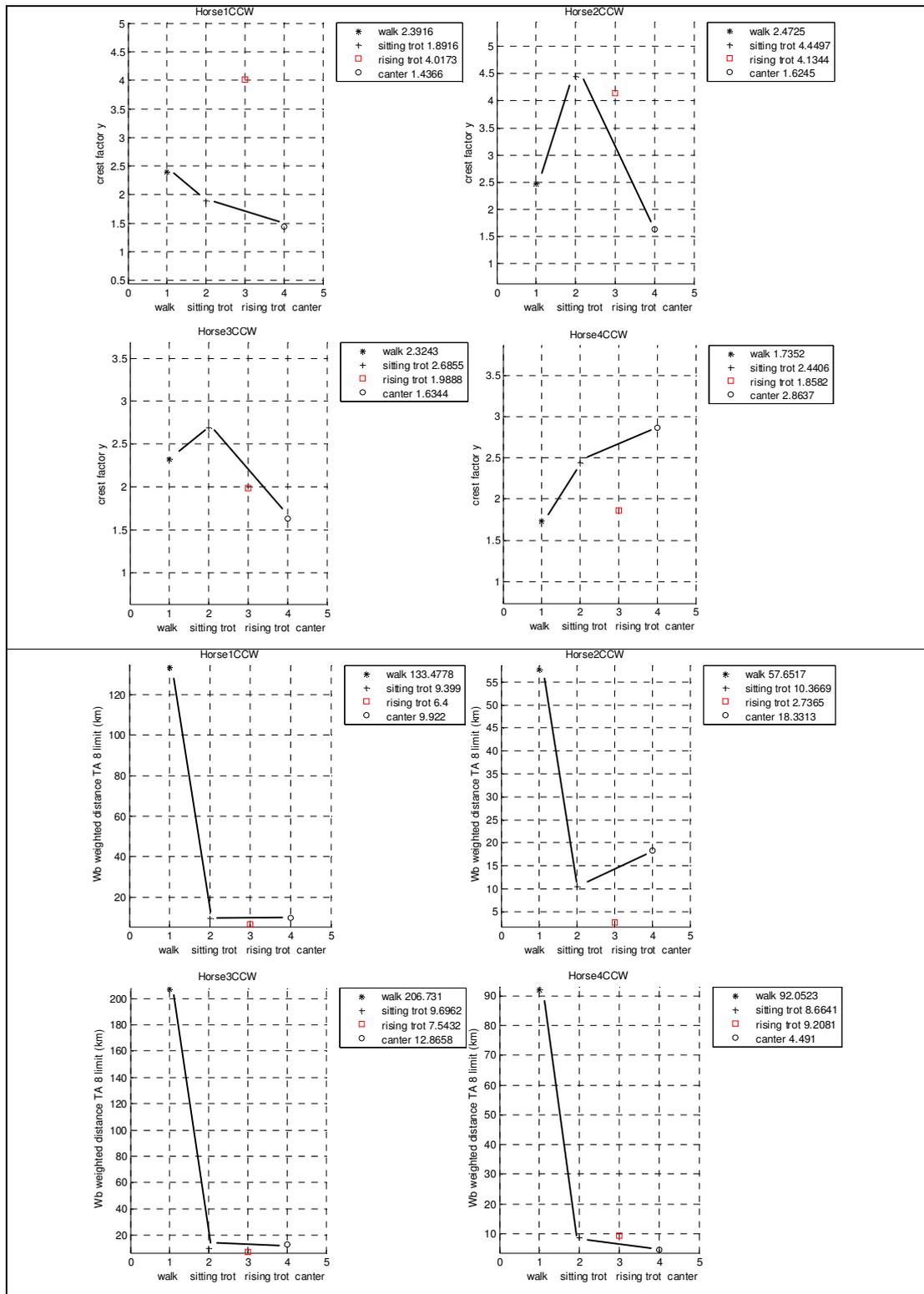


Figure 8.30 crest\_factor, Distance\_TA\_8\_limit\_Wb versus gait style for 4 different horses

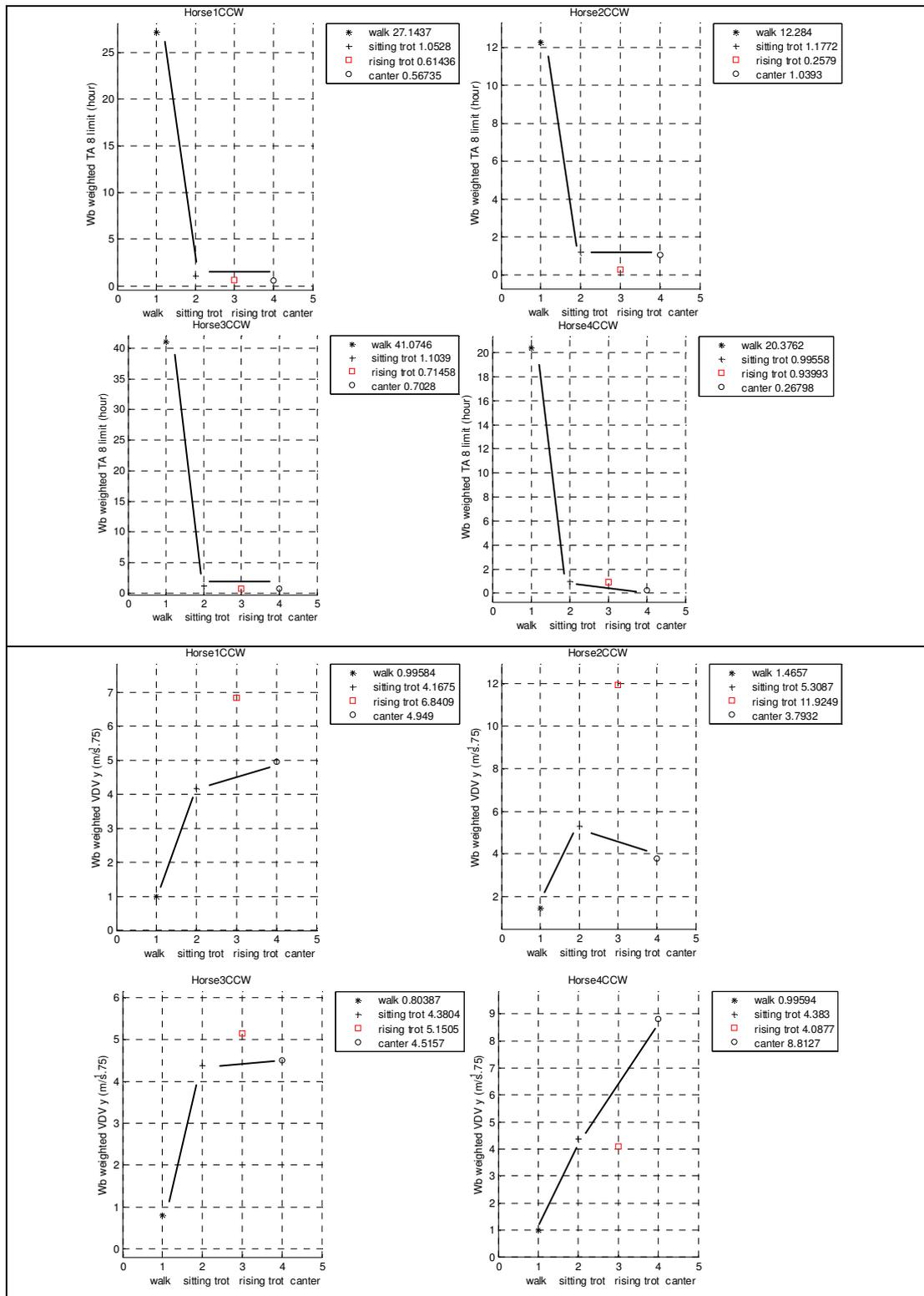


Figure 8.31 TA\_8\_limit\_Wb, VDV\_y\_Wb versus gait style for 4 different horses

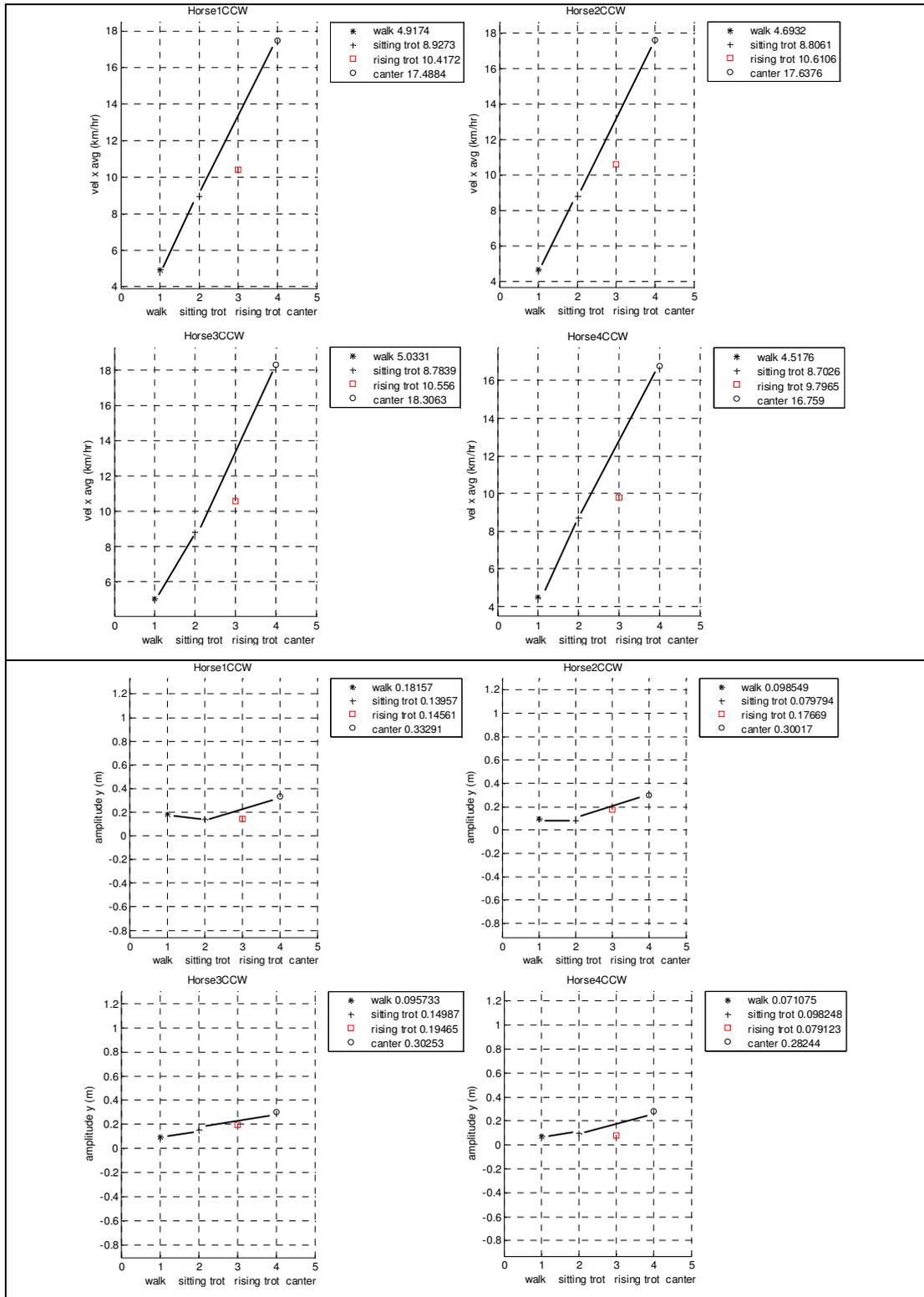


Figure 8.32  $vel\_x\_avg$ ,  $amplitude\_y$  versus gait style for 4 different horses

As can be seen clearly in Figure 8.32, the value of the average velocity in the motion direction (x axis) increases (for all horses) as in the following order: walk, sitting trot, rising trot, and canter. Finally, it is observed that the amplitude has the highest value in canter and lowest value in walk for all horses.

### 8.5.2 Different Horses Performing Each Gait

In the video recordings realized by the developed vision setup, 4 horses have been ridden by 4 experienced riders. The identifying features of the 4 horses were given in Table 8.6. All riders were asked to ride all horses in all gaits and they were asked to grade the comfort level of the horse in each gait. The grades used were 1, 2, 3, 4 and 5 corresponding to very bad, bad, medium, good and very good, respectively. The grades assigned by the riders are tabulated in Table 8.8.

Table 8.8 The riders' grades for horse gaits

WALK (CCW)					SITTING TROT (CCW)				
	Horse 1	Horse 2	Horse 3	Horse 4		Horse 1	Horse 2	Horse 3	Horse 4
Rider A	3	4	4	2	Rider A	2	3	4	2
Rider B	4	5	4	3	Rider B	3	4	4	2
Rider C	4	5	5	4	Rider C	3	4	4	2
Rider D	4	4	4	2	Rider D	2	5	4	2
avg	3.75	4.5	4.25	2.75	avg	2.5	4	4	2

RISING TROT (CCW)					CANTER (CCW)				
	Horse 1	Horse 2	Horse 3	Horse 4		Horse 1	Horse 2	Horse 3	Horse 4
Rider A	3	4	4	3	Rider A	2	5	5	3
Rider B	3	5	4	3	Rider B	3	5	5	3
Rider C	3	4	5	3	Rider C	3	5	5	2
Rider D	3	5	4	3	Rider D	3	4	5	1
avg	3	4.5	4.25	3	avg	2.75	4.75	5	2.25

	Horse 1	Horse 2	Horse 3	Horse 4
overall avg ccw	3	4.4375	4.375	2.5

In order to investigate the relationships between the comfort level indicators and the grades, the average grade assigned by the riders to a specific horse in a specific gait has been used.

In the case of an acceleration value (maximum, rms or VDV); as the grade gets higher, the acceleration is expected to get lower. Additionally, the allowable duration of riding and the distance travelled are expected to get longer, as the grades get higher.

By examining the plots in Figure 8.33 for the walk, Figure 8.34 for the sitting trot, Figure 8.35 for the rising trot and Figure 8.36 for the canter, the following results are deduced.

For the case of walk, the relationship between the acceleration data and grades do not turn out to be as expected. The reason for this discrepancy could be that, walk is the slowest gait with small displacements which may not be correctly measured using the low resolution cameras and there is a small difference in the comfort level to feel for the riders. The time and distance indicators are quite as expected.

In the sitting trot, the rms acceleration is inversely proportional to the grades as expected. Remarkably, time and distance indicators are directly proportional to the grades awarded.

Canter results are in accordance with the expectations the most in the sense that, the rms and VDV acceleration, time and distance indicators vary as expected.

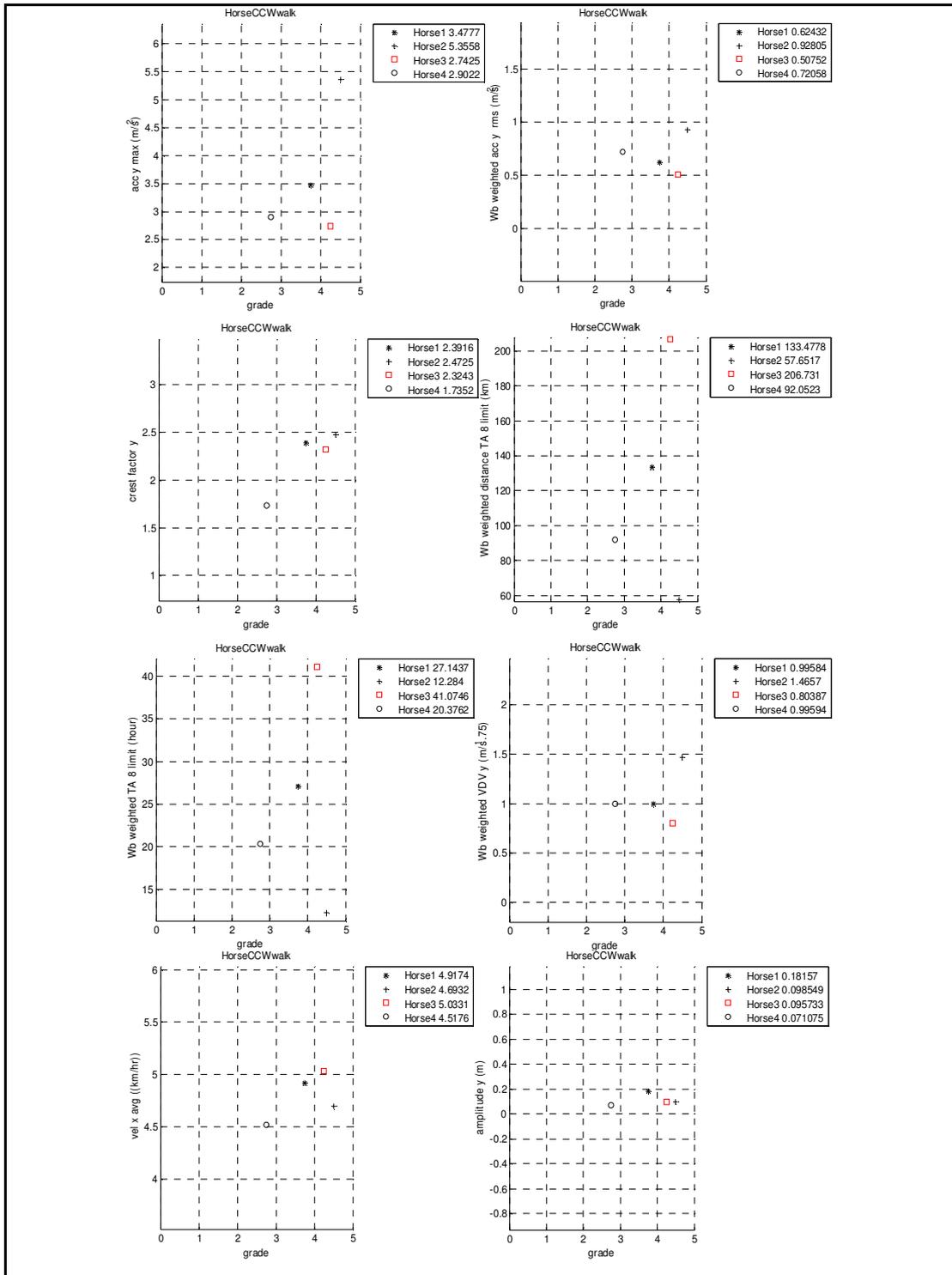


Figure 8.33 Relations between comfort indicators and grades (walk)

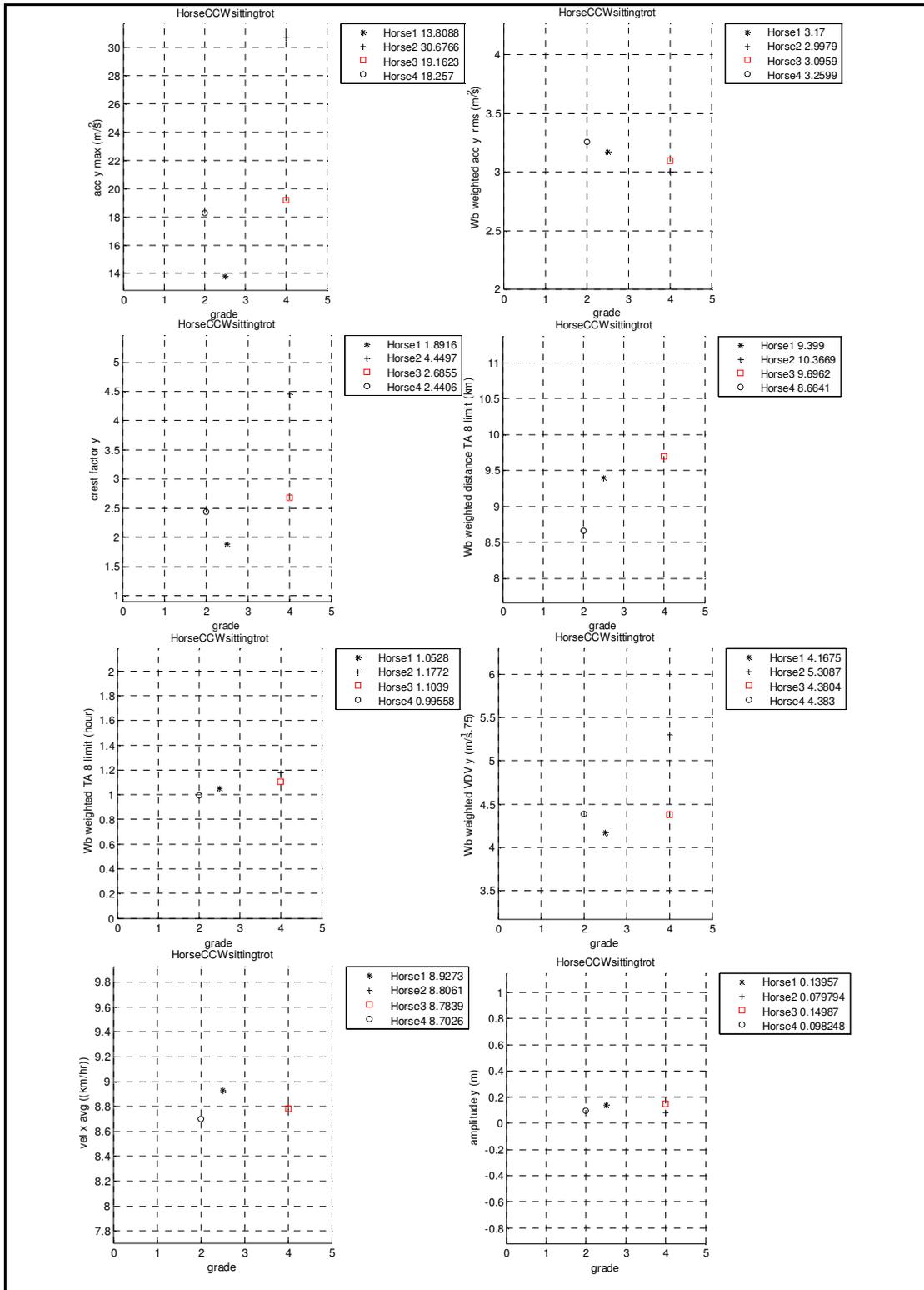


Figure 8.34 Relations between comfort indicators and grades (sitting trot)

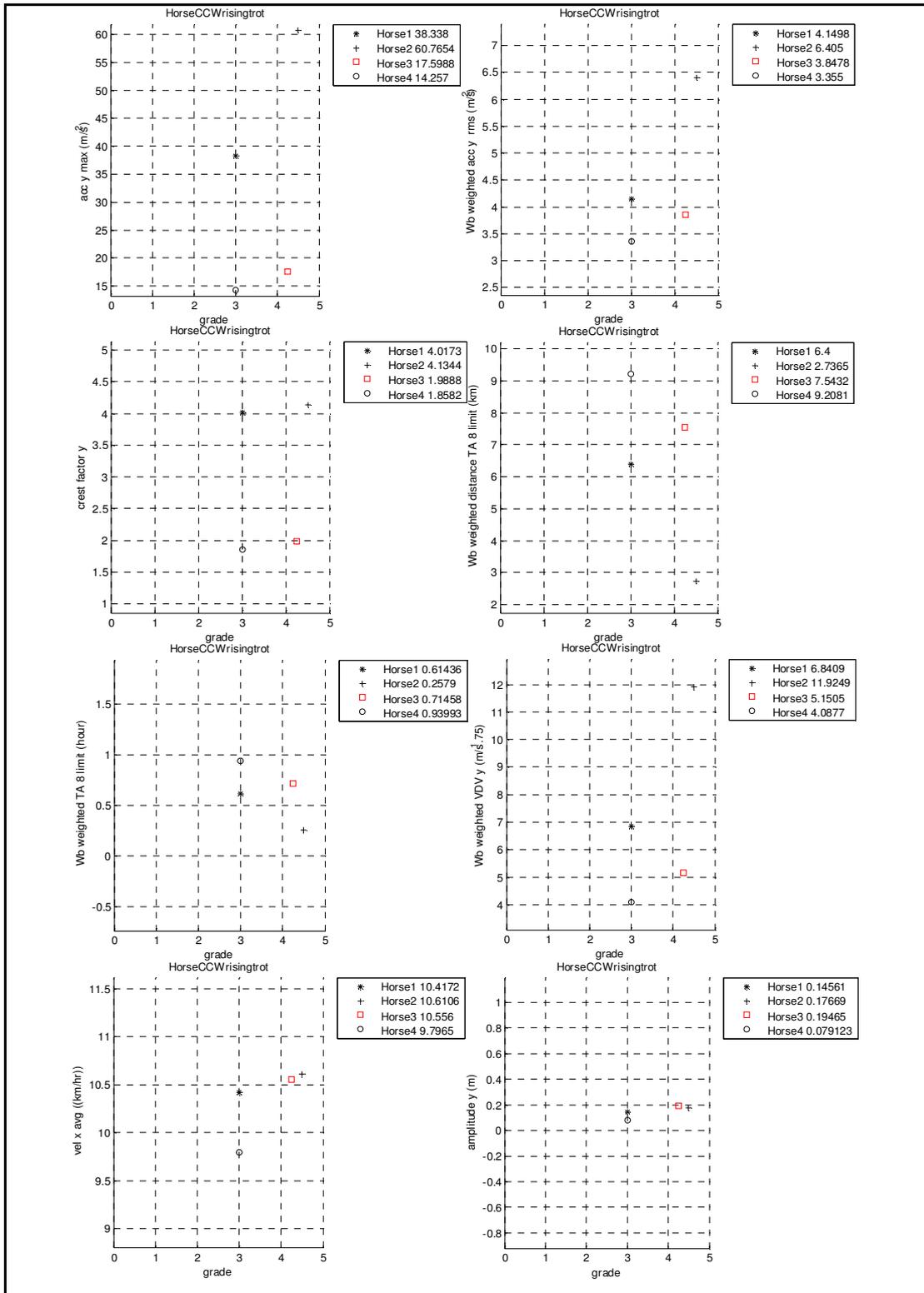


Figure 8.35 Relations between comfort indicators and grades (rising trot)

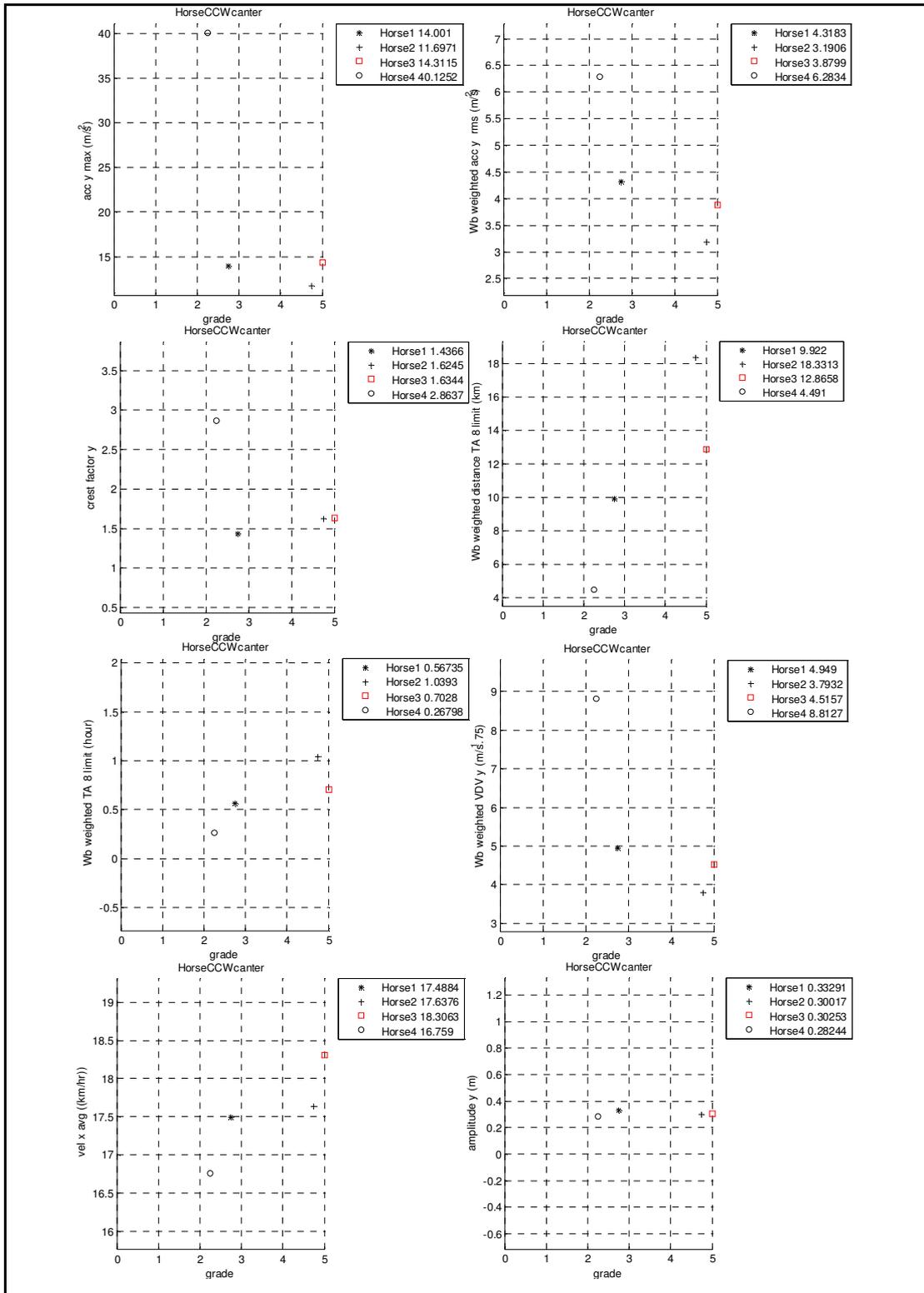


Figure 8.36 Relations between comfort indicators and grades (center)

## CHAPTER 9

### CONCLUSION AND FUTURE WORK

#### 9.1 Conclusion

The contributions of this thesis study can be summarized as given below:

- A modular vision setup, which consists of 3 inexpensive, commercial cameras, is developed to track passive markers.
- A MATLAB<sup>®</sup> toolbox available for the camera calibration (Bouguet, J. Y. (2008) is examined and utilized for calibrating the cameras of the developed setup.
- The vision setup developed in this study is intended to be used for tracking objects, or markers, that move in a pretty large calibration volume. Typically, this calibration volume is 6.5 m x 0.7 m x 0.5 m of size. Hence, a suitable calibration pattern is moved in the calibration volume via a 2 degree of freedom mechanism, attached to a chassis with wheels, which is designed and manufactured for this study.
- In order to extract the image coordinates of the markers, various filters have been developed. The images are firstly filtered with respect to the areas of regions. Then a dynamic window is introduced and a filter, based on the fixed distances between markers on a rigid body, is used. Finally, the so-called cross product filter is used to extract the image coordinates of the markers.

- Using the vision system, it is possible to obtain the position vectors of 3 markers attached on a rigid body. Based on this data, an algorithm is developed to determine all kinematic properties of a rigid body. These properties include the position, velocity, acceleration of any point (including invisible points) on the body as well as the angular velocity and acceleration of the body. It is proved that the algorithm fails at the singular positions. These positions occur when the normal vector of the plane formed by the 3 markers and the angular velocity (or acceleration) are perpendicular to each other.
- Using the vision system setup and the developed algorithm (for tracking the kinematics of a rigid body) the motion of the saddle of a horse is analyzed. A convenient, so called motion reference frame, is introduced for this purpose. The similarities and differences between the motions of different horses and gaits are investigated.
- Using the limitations induced by the whole body vibration of humans, for the first time in the world, daily, allowable riding time and riding distances are determined for horses. These limiting times and distances differ from horse to horse and gait to gait. Furthermore, several novel, quantitative horse comfort indicators are proposed. Some experiments are performed to assess the validity of these indicators. The results indicate that the indicators are consistent with the comfort assessment of experienced riders.
- For the implementation of the aforementioned ideas, a computer code, in MATLAB<sup>®</sup>, is developed.

## **9.2 Future Work**

Based on the results of this work, various recommendations can be suggested as future work.

The frame rate (30 fps) and the resolution (640x480) of the inexpensive, readily available cameras used are the basic factors that affect the accuracy of the results. An increase in the frame rate and/or the resolution will lead to much more accurate results.

The calibration of cameras is a necessary step for the mapping of the image and world coordinates of points. However, at each time of utilization of the system, it is required to re- calibrate the cameras (if the positions of cameras are changed). In order to minimize the effort for the calibration process, pre-calibration of cameras can be achieved. Research on this topic has not been met; therefore a study would be very useful.

Having made suggestions on the improvement of the vision system, future work including the application and results of the system can be diversified as follows:

The application areas of the system can be improved. Some other fields of application can be classified as follows: Gait analysis of humans and animals, reaching kinematic data of machines, mechanisms and industrial robots (in a fast and practical manner), experimental set-ups in education institutions and crash tests.

The data obtained at the end of this study can be used as the inputs for a horse simulator. The simulator may have the capability of simulating each gait of each different horse. Four different horses in four different gaits have been analyzed and the necessary linear and angular kinematic data of the saddle have been obtained as

outputs. Hence, it can be said that the necessary data for designing a horse simulator is now available.

Furthermore, this study can be extended to detect lameness in horses, in a quantitative, objective manner. Such a tool would definitely be helpful to the veterinarians in diagnosing and curing lameness.

## REFERENCES

- Armen, S. A. (1990). Training horse simulator. United States Patent. US 4,957,444.
- Besl, P. J. & McKay, N. D. (1992). A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12. 239-256.
- Bouguet, J. Y. (2008, June). Camera calibration toolbox for MATLAB<sup>®</sup>. [online] [http://www.vision.caltech.edu/bouguetj/calib\\_doc/](http://www.vision.caltech.edu/bouguetj/calib_doc/), last accessed date: 02 /06/2008.
- BS 6841 (1987). Guide to measurement and evaluation of human exposure to whole-body mechanical vibration and repeated shock. British Standards Institution. London.
- Cannavino D. T. (2001). Posting trot and canter simulator for horseback riders. United States Patent. US 6,264,569.
- Chapra, S.C. & Canale, R.P. (2003). *Numerical Methods for Engineers: With Software and Programming Applications*. USA: McGraw-Hill.
- Forsyth, D. A. & Ponce, J. (2003). *Computer Vision: A Modern Approach*. USA: Pearson Education, Inc.
- Gonzalez, R. C. & Woods, R. E. (2002). *Digital Image Processing*. USA: Prentice Hall.
- Greenberg, M. D. (1998). *Advanced Engineering Mathematics*. New Jersey: Prentice Hall.

Heikkilä, J. & Silvén, O. (1997). A four-step camera calibration procedure with implicit image correction. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1106–1112.

ISO 2631 (1997-2003). Mechanical vibration and shock- evaluation of human exposure to whole-body vibration. International Organization for Standardization. Geneva.

Keegen, K. G., Wilson, D. A., Kramer, J., Howard, R., Johnson, P. J., Pai, P. F., et al. (2003). Computer-Assisted Lameness Diagnosis in the Horse [online] <http://www.equinelameness.org/technique.htm>, last accessed date: 2003.

Koenig, P. & Bekey, G. (1993) Generation and control of gait patterns in a simulated horse. International Conference on Intelligent Robots and Systems, 572-579. Japan.

Kijima, R., Kouno, M., Hashimoto, K., Jiang, Y., Aoki, T. & Ojika, T. (2003). Karakuri horse riding therapy. International Conference on Rehabilitation Robotics. Japan: Gifu University.

Mansfield, N.J. (2005). *Human Response to Vibration*. Florida: CRC Press.

Mansson, J. (1998). Stereovision: a model of human stereopsis. Lund University Cognitive Studies, Sweden.

Marr, D. & Poggio, T. (1979). A computational theory of human stereo vision. Proc. R. Soc. London Ser. B 204, 301-28.

Mathworks, Inc. (2002). Curve Fitting Toolbox for use with MATLAB®. User's guide. Version 5.

Mathworks, Inc. (2006). Image processing toolbox for use with MATLAB®. User's guide. Version 1.

Micklem W. (2003). *Complete Horse Riding Manual*. London: Dorling Kindersley Limited.

Muybridge, E. (1887). *Animals in Motion*. New York: Dover.

Poggio, G. F. & Poggio, T. (1984). The analysis of stereopsis. *Annual Reviews Neuroscience*, 379-412.

Rimell, A. N. & Mansfield, N.J. (2007). Design of digital filters for frequency weightings required for risk assessments of workers exposed to vibration. *Industrial Health*, 45, 512-519.

Sato, M., Terajima, M., Ojika, T., Kijima, R., Kobayashi, T., Inoue, Y., et al. (1999). Development of horse-riding simulator by "Karakuri" technique. *International Conference on Virtual System and Multimedia*, 98-106. Dundee.

Shinomiya, Y., Nomura, J., Yoshida, Y. & Kimura, T. (1997). Horseback riding therapy simulator with VR technology. *ACM Symposium on Virtual Reality Software and Technology*, 9-14.

Shinomiya, Y., Ozawa, T., Hosaka, Y., Wang, S., Ishida, K. & Kimura, T. (2003). Development and physical training evaluation of horseback riding therapeutic equipment. *IEEE\ASME International Conference on Advanced Intelligent Mechatronics*, 1239-1243.

Wei, G. Q. & Ma, S. D. (1993). A complete two-plane camera calibration method and experimental comparisons. Proc. 4th International Conference on Computer Vision. 439-446. Germany.

Yamaguchi, M., Kito, M., Iguchi, N., Yoshida, S. & Ishigure, A. (1991). Riding simulator. United States Patent. US 4,988,300.

Zhang, Z. (2000). A flexible new technique for camera calibration, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1330-1334.

## APPEDIX A

### USER'S GUIDE

Please read the comments in the codes for the sake of understanding and follow the instructions below. The codes in sequence are placed under the All procedure\_Codes\_Vision\_System folder. Each data is saved as a MAT-file and each file has the extension of .mat.

#### 1. Create Results Folder

##### a. create\_results\_folder.m

Creates an empty folder named Results\_Vision\_System in the location: "C:\ ", i.e. "C:\Results\_Vision\_System". All variables are saved in this folder. It is recommended that the files are viewed as date modified sorted for following the steps.

#### 2. Calibration of each camera

##### a. calib left

##### i. image names padding with zeros

*left\_image\_names\_padding\_with\_zeros.m*

Image file names used for calibration should be proper such that they should be sorted since sequence of images is important for the applications in MATLAB<sup>®</sup>. For this, first count the maximum number of digits of image file names. Pad the image names with zeros if necessary according to the maximum number of digits counted. Ex: If left.AVI15.tif, left.AVI112.tif and left.AVI1212.tif images should be used for calibration, their image names should be modified according to the maximum number of number digits (i.e. 4). Therefore, pad left.AVI15.tif with 2 zeros, left.AVI112.tif with 1 zero.

left.AVI15.tif → left.AVI0015.tif

left.AVI112.tif → left.AVI0112.tif

left.AVI1212.tif → left.AVI1212.tif (remains)

Now all filenames have equal number of digits.

! Put only the image files in the current folder to be padded with same number of zeros. Run the code, and move the files in temp folder after the zero padding procedure. Repeat if necessary.

##### ii. renaming and renumbering image names

*left\_renaming\_renumbering1.m*

Put all the images in the current folder whose names are properly modified by padding with zeros before. Run the code to rename the files as

left.AVI\*.tif → left\*.tif and then,

to renumber starting from 1 to the number of images:

Ex: left.AVI0015.tif → left0015.tif → left0001.tif

left.AVI0112.tif → left0112.tif → left0002.tif

left.AVI1212.tif → left1212.tif → left0003.tif

! If one does not do the padding process, the images are sorted as;

left.AVI112.tif → → left0001.tif

left.AVI1212.tif → → left0002.tif

left.AVI15.tif → → left0003.tif

which is a totally wrong sorting and causes false image matching in the stereo calibration step.

Begin to calibrate the left camera. Firstly read the comments in *left\_read\_me\_calib\_save2.m* without running it. Write on the command window; *calib\_gui*

After the calibration results obtained, run the code;

*left\_read\_me\_calib\_save2.m*

Therefore, the calibration results are saved as; C:\Results\_Vision\_System\Calib\_Results\_left

b. calib middle (similar to part 2.a.)

i. image names padding with zeros

*middle\_image\_names\_padding\_with\_zeros.m*

ii. renaming and renumbering image names

*middle\_renaming\_renumbering1.m*

*calib\_gui*

*middle\_read\_me\_calib\_save2.m*

c. calib right (similar to part 2.a.)

i. image names padding with zeros

*right\_image\_names\_padding\_with\_zeros.m*

ii. renaming and renumbering image names

*right\_renaming\_renumbering1.m*

*calib\_gui*

*right\_read\_me\_calib\_save2.m*

∴ 3 files are created;

I. C:\Results\_Vision\_System\Calib\_Results\_left

II. C:\Results\_Vision\_System\Calib\_Results\_middle

III. C:\Results\_Vision\_System\Calib\_Results\_right

**3.** Stereo calibration of each pair of cameras

a. stereo calib left middle

Begin to calibrate each pair of cameras. Firstly read the comments in *left\_middle\_read\_me\_save.m* and follow the instructions without running it. (Put the files named Calib\_Results\_left and Calib\_Results\_middle that were created in step 2. in current directory). Write on the command window;

*stereo\_gui*

After the stereo calibration results obtained, run the code;

*left\_middle\_read\_me\_save.m*

Therefore, the stereo calibration results are saved as;

C:\Results\_Vision\_System\Calib\_Results\_stereo\_left\_middle

i. *stereo\_gui*

ii. *left\_middle\_read\_me\_save.m*

b. stereo calib left right (similar to part 3.a.)

i. *stereo\_gui*

ii. *left\_right\_read\_me\_save.m*

c. stereo calib middle right (similar to part 3.a.)

i. *stereo\_gui*

ii. *middle\_right\_read\_me\_save.m*

∴ 3 files are created (totally 6 files);

IV. C:\Results\_Vision\_System\Calib\_Results\_stereo\_left\_middle

V. C:\Results\_Vision\_System\Calib\_Results\_stereo\_left\_right

VI. C:\Results\_Vision\_System\Calib\_Results\_stereo\_middle\_right

4. All image centroids (filtering via areas and distances with window) from each camera

a. left image centroids

i. *left\_level\_info\_from\_all\_images\_1.m*

Do not forget to do the application of padding zeros to the image names similarly as in part 2.a.i. if necessary. You don't need to rename and renumber the image files for this application.

Put all the image files related to the motion in the current folder. Run the code and convert the rgb colormap to black and white by entering the threshold value (left\_levelinput) out of 255. Check the results such that the markers are visible and isolated in all motion images. If they are not, change the level input accordingly and check the results again until they are satisfying. The level information is saved as; C:\Results\_Vision\_System\left\_level\_info\_from\_all\_images\_visible\_markers

ii. *left\_data\_from\_first\_image\_2.m*

First image is the input for this file, run this file and enter the required data as explained in the code. (Enter the marker labels (left\_labels), minimum area value (left\_area\_threshold\_min), maximum area value (left\_area\_threshold\_max), length of marker tracking window (left\_window\_dx), width of

marker tracking window (left\_window\_dy), distance percent error out of 1 (left\_distance\_percent\_error). The result of the process should be checked from the figure such that only the desired markers should be seen in the figure. Necessary information from the first image is saved as;

C:\Results\_Vision\_System\left\_data\_from\_first\_image\_visible\_markers

iii. *left\_image\_centroids\_3.m*

Run the code and track the figures that open and close simultaneously to see if only the desired markers are seen. The visible markers' image centroids are saved as;

C:\Results\_Vision\_System\left\_image\_centroids\_visible\_markers

b. middle image centroids (similar to part 4.a.)

i. *middle\_level\_info\_from\_all\_images\_1.m*

ii. *middle\_data\_from\_first\_image\_2.m*

iii. *middle\_image\_centroids\_3.m*

c. right image centroids (similar to part 4.a.)

i. *right\_level\_info\_from\_all\_images\_1.m*

ii. *right\_data\_from\_first\_image\_2.m*

iii. *right\_image\_centroids\_3.m*

∴ 9 files are created (totally 15 files);

VII. C:\Results\_Vision\_System\left\_level\_info\_from\_all\_images\_visible\_markers

VIII. C:\Results\_Vision\_System\left\_data\_from\_first\_image\_visible\_markers

IX. C:\Results\_Vision\_System\left\_image\_centroids\_visible\_markers

X. C:\Results\_Vision\_System\middle\_level\_info\_from\_all\_images\_visible\_markers

XI. C:\Results\_Vision\_System\middle\_data\_from\_first\_image\_visible\_markers

XII. C:\Results\_Vision\_System\middle\_image\_centroids\_visible\_markers

XIII. C:\Results\_Vision\_System\righ\_level\_info\_from\_all\_images\_visible\_markers

XIV. C:\Results\_Vision\_System\righ\_data\_from\_first\_image\_visible\_markers

XV. C:\Results\_Vision\_System\righ\_image\_centroids\_visible\_markers

**5.** Combination of image centroids results and stereo calibration parameters

i. *combine\_image\_centroids\_and\_calibration.m*

Run the code and the position of the markers in motion with respect to the left most camera is to be saved by combining the stereo calibration and markers' image centroids. The resultant files are;

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_middle\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_right\_calib

C:\Results\_Vision\_System\XYZ\_middle\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates

ii. *define\_visible\_markers.m*

Run this code to change the variable names.

(XYZ\_left\_camera\_coordinates→XYZ\_left\_camera\_coordinates\_visible\_markers)

The resultant file is saved as:

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_visible\_markers

∴ 6 files are created (totally 21 files);

XVI. C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_middle\_calib

XVII. C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_right\_calib

XVIII. C:\Results\_Vision\_System\XYZ\_middle\_camera\_coordinates\_middle\_right\_calib

XIX. C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_middle\_right\_calib

XX. C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates

XXI. C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_visible\_markers

#### 6. Smoothing the position of visible markers

i. *smooth\_x\_left\_camera\_coordinates\_markers.m*

Run the code, enter the period (as obtained from the images as frames) and the title for the graphs.

The smoothed position results are saved as;

C:\Results\_Vision\_System\X\_smoothed\_results\_visible\_markers

ii. *smooth\_y\_left\_camera\_coordinates\_markers.m*

iii. *smooth\_z\_left\_camera\_coordinates\_markers.m*

∴ 3 files are created (totally 24 files);

XXII. C:\Results\_Vision\_System\X\_smoothed\_results\_visible\_markers

XXIII. C:\Results\_Vision\_System\Y\_smoothed\_results\_visible\_markers

XXIV. C:\Results\_Vision\_System\Z\_smoothed\_results\_visible\_markers

#### 7. Converting position of markers from the leftmost camera to motion reference frame

a. one image centroids of gravity showing vector

i. *one\_image\_centroids.m*

In order to convert the position of markers with respect to left camera reference frame to the motion reference frame, tip and tail coordinates of the gravity showing vector should be found. Therefore, put necessary images of the plumb (showing the gravity) from left, middle and right cameras in the current folder. Three images are the inputs for this file, run this file and enter the required data as explained in the code.

After entering the threshold value (left\_levelinput), the marker labels (left\_labels), similarly (middle\_levelinput, middle\_labels, right\_levelinput, right\_labels) check the results to see that only the

desired points are seen. Therefore, the centers of the tip and tail of the gravity showing vector are saved as;

C:\Results\_Vision\_System\left\_image\_centroids

C:\Results\_Vision\_System\middle\_image\_centroids

C:\Results\_Vision\_System\right\_image\_centroids

b. combination of image centroids and calibration gravity showing vector

i. *combine\_image\_centroids\_and\_calibration.m*

Run the code and the tip and tail points of the gravity showing vector with respect to the left most camera is to be saved by combining the stereo calibration and markers' image centroids. (Note that these files are written over the old files having the same file name)

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_middle\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_right\_calib

C:\Results\_Vision\_System\XYZ\_middle\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates

ii. *define\_gravity.m*

Run this code to define the estimated unit gravity vector.

The resultant file is saved as:

C:\Results\_Vision\_System\y\_axis\_gravity\_unit\_estimated

c. convert from camera to motion reference frame

The smoothed position of markers and the estimated unit gravity vector are obtained to convert the markers in left-most camera reference frame to motion reference frame.

i. *convert\_smoothed\_camera\_to\_motion\_reference\_frame.m*

Run the code to convert the position markers in left-most camera reference frame to motion reference frame. Enter marker number (1, 2 or 3) which defines the x-motion direction. The result is saved as;

C:\Results\_Vision\_System\XYZ\_visible\_markers\_smoothed\_motion\_coordinates

∴ 5 files are created (totally 29 files);

XXV.C:\Results\_Vision\_System\left\_image\_centroids

XXVI.C:\Results\_Vision\_System\middle\_image\_centroids

XXVII.C:\Results\_Vision\_System\right\_image\_centroids

XXVIII.C:\Results\_Vision\_System\y\_axis\_gravity\_unit\_estimated

XXIX.C:\Results\_Vision\_System\XYZ\_visible\_markers\_smoothed\_motion\_coordinates

**8.** Registration of the position of markers (in case of rigidity of markers is required)

a. one image centroids of static shot of markers

i. *one\_image\_centroids.m*

In order the rigid body constraints to be assumed for the markers, images from each camera of the static position of the markers are required. Therefore, put necessary images of static shot from left, middle and right cameras in the current folder. Three images are the inputs for this file, run this file and enter the required data as explained in the code. After entering the threshold value (left\_levelinput), the marker labels (left\_labels), similarly (middle\_levelinput, middle\_labels, right\_levelinput, right\_labels) check the results to see that only the desired markers are seen.

Therefore, image centroids of markers that are assumed to be rigid are saved as (Note that these files are written over the old files having the same file name);

C:\Results\_Vision\_System\left\_image\_centroids

C:\Results\_Vision\_System\middle\_image\_centroids

C:\Results\_Vision\_System\right\_image\_centroids

b. combination of image centroids and calibration static shot of markers

i. *combine\_image\_centroids\_and\_calibration.m*

Run the code and the static position of the markers with respect to the left most camera is to be saved by combining the stereo calibration and markers' image centroids. (Note that these files are written over the old files having the same file name)

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_middle\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_right\_calib

C:\Results\_Vision\_System\XYZ\_middle\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates

ii. *define\_static\_shot\_markers2.m*

Run this code to change the variable name:

(XYZ\_left\_camera\_coordinates→XYZ\_static\_shot\_markers\_left\_camera)

The resultant file is saved as:

C:\Results\_Vision\_System\XYZ\_left\_camera\_static\_shot\_markers

iii. *convert\_camera\_motion\_static\_shot3.m*

Run the code to convert the position of static shot markers in left-most camera reference frame to motion reference frame. The resultant file is saved as:

C:\Results\_Vision\_System\XYZ\_motion\_static\_shot\_markers

c. registration of the position of markers

Smoothed position of markers in motion reference frame (non-rigid) and the static shot of same markers (rigidly assumed) are obtained to register the non-rigid coordinates of the markers.

i. *registration\_of\_markers.m*

Run the code to register the position of markers accordingly. The result is saved as;

C:\Results\_Vision\_System\XYZ\_motion\_coordinates\_registered

∴ 3 files are created (totally 32 files);

XXX. C:\Results\_Vision\_System\XYZ\_left\_camera\_static\_shot\_markers

XXXI. C:\Results\_Vision\_System\XYZ\_motion\_static\_shot\_markers

XXXII. C:\Results\_Vision\_System\XYZ\_motion\_coordinates\_registered

#### 9. Linear velocity and acceleration markers by fits

Fitting suitable curves to the position of the markers that are registered in motion reference frame is to be done. Note that, although curves are fitted to the position, only velocity and acceleration data will be used whereas the original registered position data will remain in the following steps.

##### i. *fit\_vel\_acc\_x\_motion\_visible\_markers.m*

Run the code, enter the type of curve (*fit\_type\_x*) to be fit from the MATLAB® Library (default type: *poly1*). The original position data, fitted position curve, velocity and acceleration curves for each marker are shown in the figures. The results are saved as;

C:\Results\_Vision\_System\x\_marker1fitdata

C:\Results\_Vision\_System\x\_marker2fitdata

C:\Results\_Vision\_System\x\_marker3fitdata

C:\Results\_Vision\_System\X\_cfitted\_results\_visible\_markers

##### ii. *fit\_vel\_acc\_y\_motion\_visible\_markers.m*

Run the code and enter the number of extreme points seen in the position data (*no\_of\_extreme*) for each marker, choose a fit type from the two recommended fit types (*fit\_choice*).

The results are saved as;

C:\Results\_Vision\_System\Y\_cfitted\_results\_visible\_markersFourier (n)

C:\Results\_Vision\_System\Y\_cfitted\_results\_visible\_markersFourier (n+1)

C:\Results\_Vision\_System\Y\_cfitted\_results\_visible\_markers

C:\Results\_Vision\_System\Y\_amplitudes\_marker1

C:\Results\_Vision\_System\Y\_amplitudes\_marker2

C:\Results\_Vision\_System\Y\_amplitudes\_marker3

##### iii. *fit\_vel\_acc\_z\_motion\_visible\_markers.m*

Run the code and enter the number of extreme points seen in the position data (*no\_of\_extreme*) for each marker, choose a fit type from the two recommended fit types (*fit\_choice*).

C:\Results\_Vision\_System\Z\_cfitted\_results\_visible\_markersFourier (n)

C:\Results\_Vision\_System\Z\_cfitted\_results\_visible\_markersFourier (n+1)

C:\Results\_Vision\_System\Z\_cfitted\_results\_visible\_markers

C:\Results\_Vision\_System\Z\_amplitudes\_marker1

C:\Results\_Vision\_System\Z\_amplitudes\_marker2

C:\Results\_Vision\_System\Z\_amplitudes\_marker3

∴ 16 files are created (totally 48 files)

#### **10. Performance measure of the motion of markers**

i. *performance\_measure.m*

Necessary information about the performance measures of the motion of the markers is obtained; the results are finalized if the markers are the required points to be tracked. The inputs have the default values and only pressing enter button is sufficient. The results are saved as;

C:\Results\_Vision\_System\performance\_results\_markers (title)

∴ 1 file is created (49 files);

#### **11. Reaching angular velocity and acceleration**

The three markers' linear velocity and acceleration information is used to obtain the angular velocity and acceleration of the rigid body.

i. *angular\_vel\_and\_acc\_all\_avg\_smooth.m*

Run the code to visualize the smoothed values for the omega (angular velocity) in six different but similar types and the average value of them. Decide the ones to include while obtaining the overall omega value. Similarly, visualize the smoothed values for the alfa (angular acceleration) in six different but similar behaviors and the average value of them. Decide the ones to include while obtaining the overall omega value. The results are saved as;

C:\Results\_Vision\_System\omega\_and\_alfa\_info

∴ 1 file is created (totally 50 files)

#### **12. Reaching required (unseen) markers data**

a. one image centroids of static shot of required (unseen) markers

i. *one\_image\_centroids.m*

In order to reach the position of required (unseen) markers, images from each camera of the static position of the unseen markers are required. Therefore, put necessary images of static shot from left, middle and right cameras in the current folder. Three images are the inputs for this file, run this file and enter the required data as explained in the code. (It is recommended to use the same files used in part 6.) After entering the threshold value (*left\_levelinput*), the marker labels (*left\_labels*), similarly (*middle\_levelinput*, *middle\_labels*, *right\_levelinput*, *right\_labels*) check the results to see that only the desired markers are seen.

So, static image centroids of markers that are required (unseen) are saved as:

C:\Results\_Vision\_System\left\_image\_centroids\_required\_markers

C:\Results\_Vision\_System\middle\_image\_centroids\_required\_markers

C:\Results\_Vision\_System\right\_image\_centroids\_required\_markers

b. combination of image centroids and calibration ss

i. *combine\_image\_centroids\_and\_calibration1.m*

Run the code and the static position of the required (unseen) markers with respect to the left most camera is to be saved by combining the stereo calibration and markers' image centroids. (Note that these files are written over the old files having the same file name)

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_middle\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_left\_right\_calib

C:\Results\_Vision\_System\XYZ\_middle\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates\_middle\_right\_calib

C:\Results\_Vision\_System\XYZ\_left\_camera\_coordinates

ii. *define\_static\_shot\_markers\_convert\_camera\_to\_motion2.m*

Run this code to change the variable names.

(XYZ\_left\_camera\_coordinates → XYZ\_static\_shot\_required\_markers\_motion)

The resultant file is saved as:

C:\Results\_Vision\_System\XYZ\_static\_shot\_required\_markers\_motion

iii. *convert\_static\_shot\_markers\_motion\_to\_body\_frame3.m*

Markers position in the motion frame is represented in the defined body reference frame. The resultant file is saved as:

C:\Results\_Vision\_System\XYZ\_static\_shot\_required\_markers\_body

c. reaching position velocity and acceleration of required marker

i. *convert\_required\_body2motion1.m*

Position of the required marker in the motion reference frame is to be obtained and saved as;

C:\Results\_Vision\_System\XYZ\_required\_markers\_motion

ii. *vel\_and\_acc\_required\_markers2.m*

Three different but similar values and the averages of the velocity and acceleration data of the required marker are to be obtained. The results are saved as;

C:\Results\_Vision\_System\vel\_acc\_info\_required\_markers

∴ 7 files are created (totally 57 files)

**13.** Performance measure of the motion of required markers

i. *performance\_measure\_required\_markers.m*

Necessary information about the performance measures of the motion of the required marker (marker on the saddle) is obtained. The inputs have the default values and only pressing enter button is sufficient. The results are saved as;

C:\Results\_Vision\_System\performance\_results\_required\_markers (title)

∴ 1 file is created (totally 58 files)

## APPEDIX B

### FLOW CHARTS FOR THE CODES

Flow diagram of the whole procedure

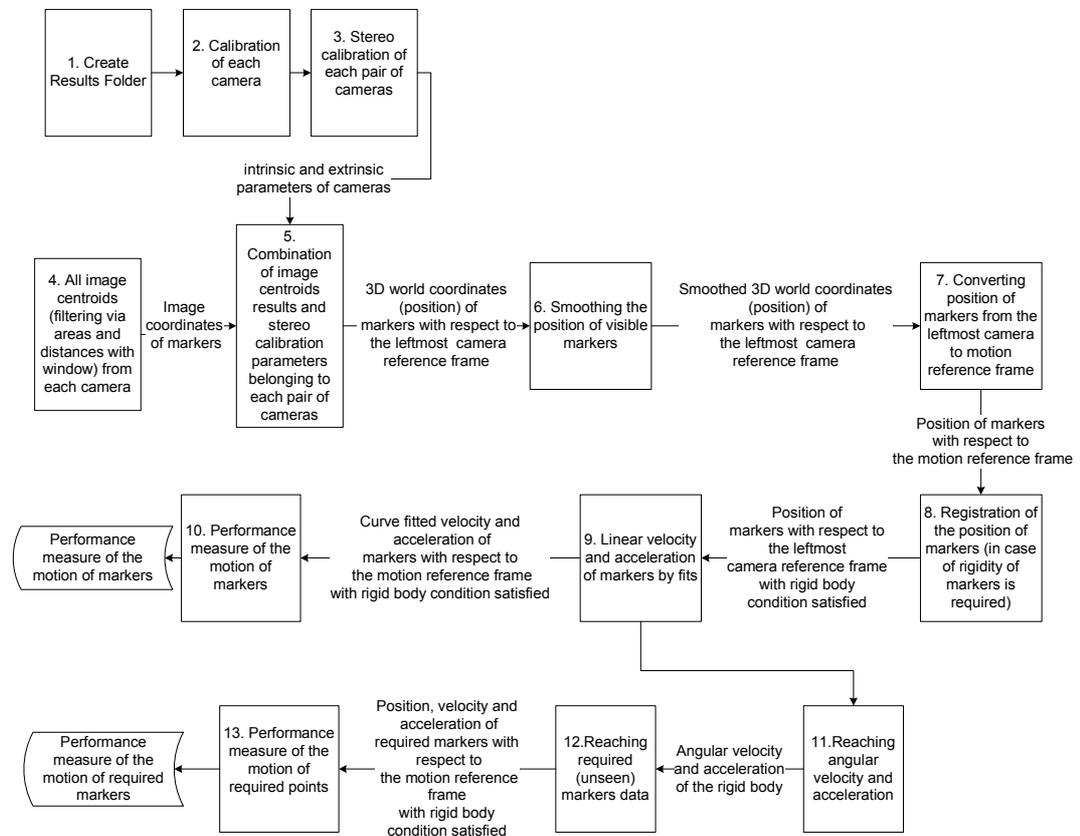


Figure B.1 Flow diagram of the whole procedure

1. Create results folder

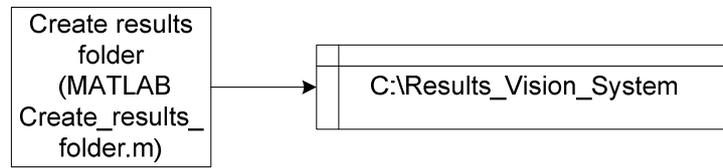


Figure B.2 Create results folder

## 2. Calibration of each camera

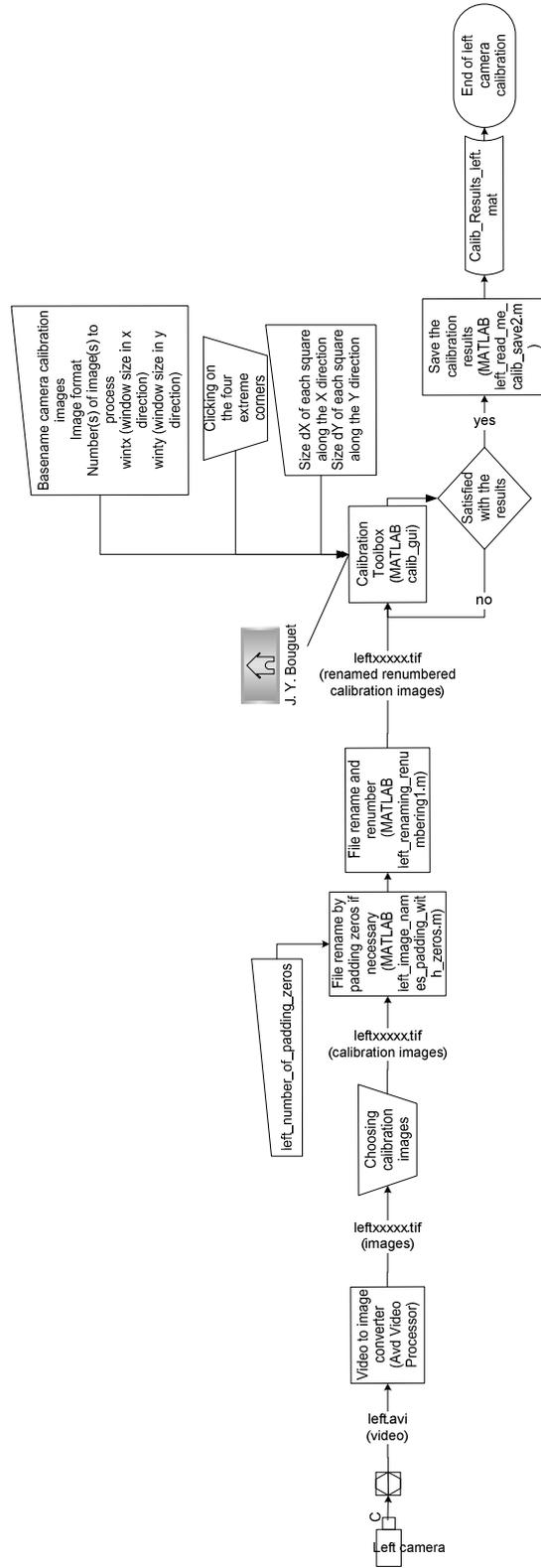
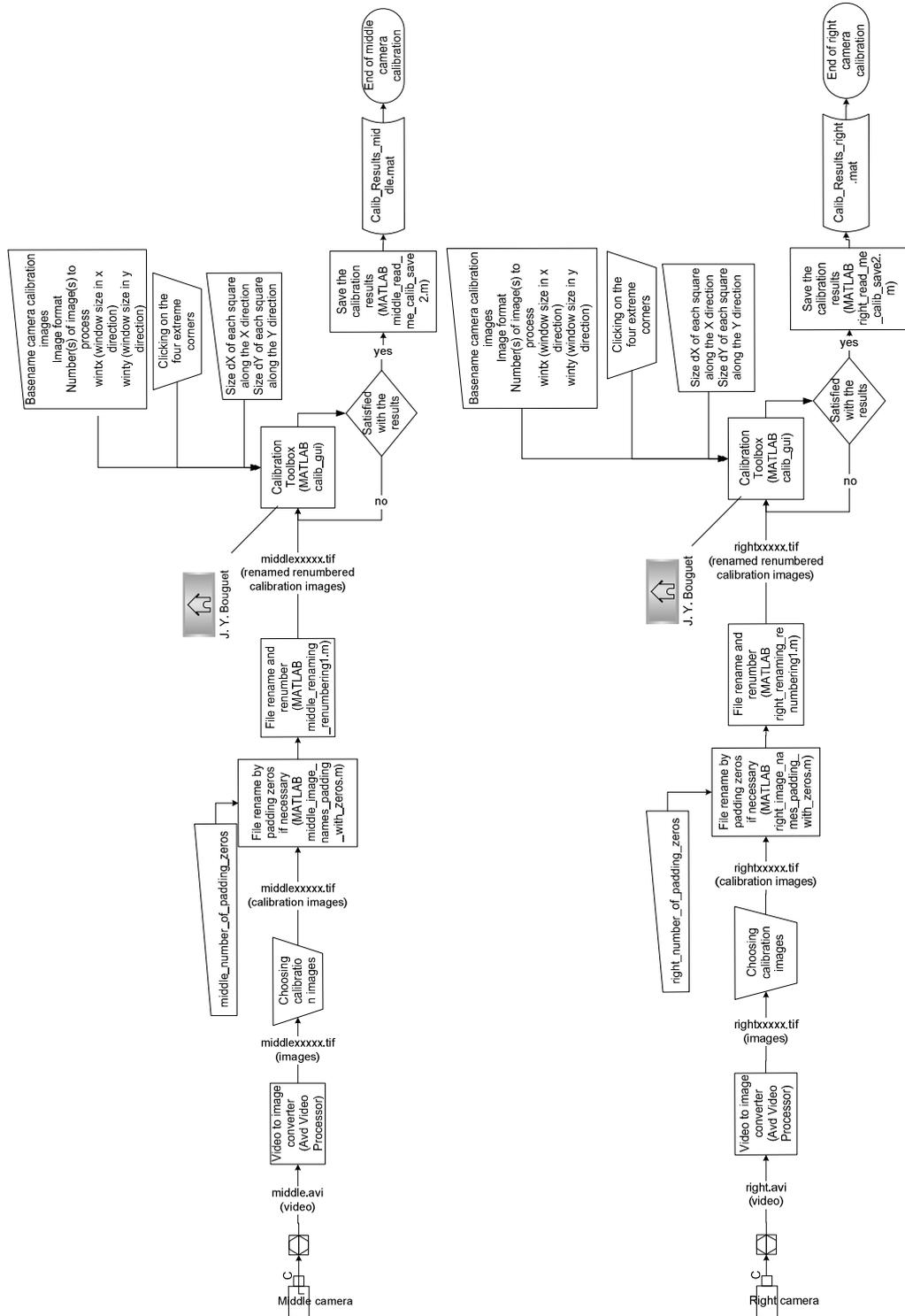


Figure B.3 Calibration of each camera



### 3. Stereo calibration of each pair of cameras

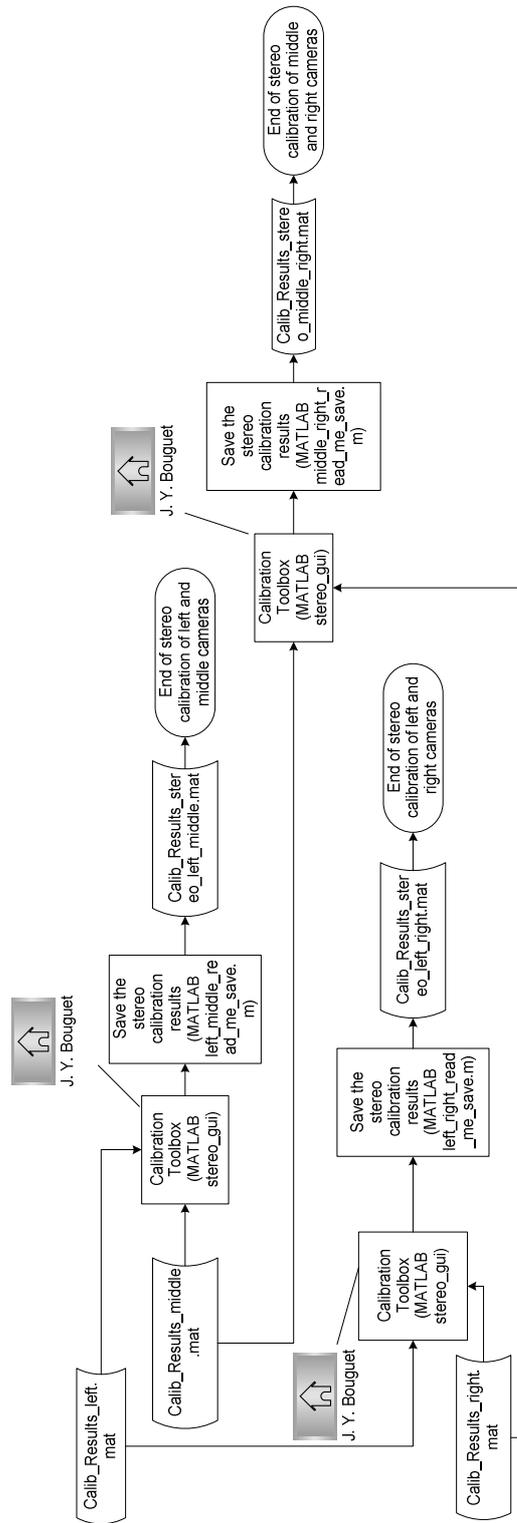


Figure B.4 Stereo calibration

4. All image centroids (filtering via areas and distances with window) from each camera

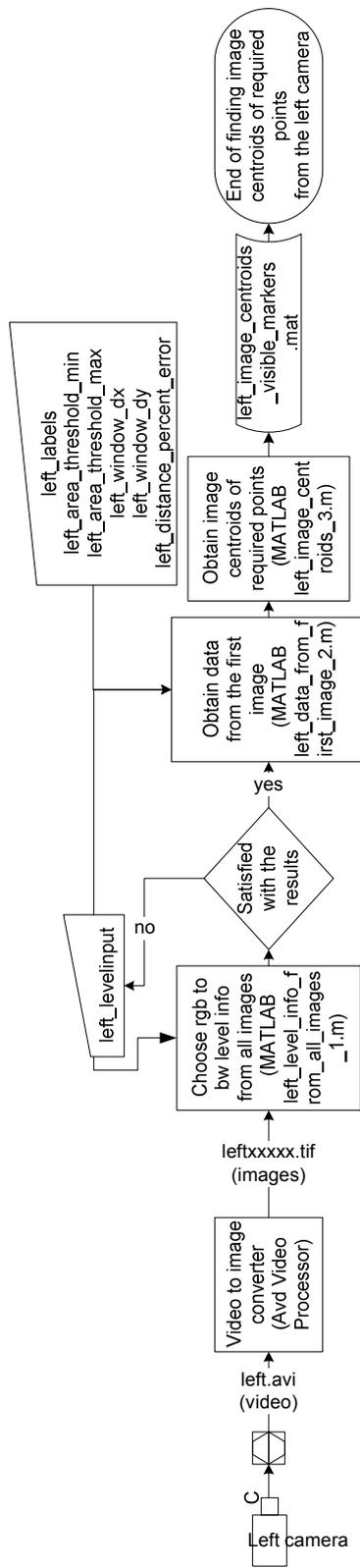
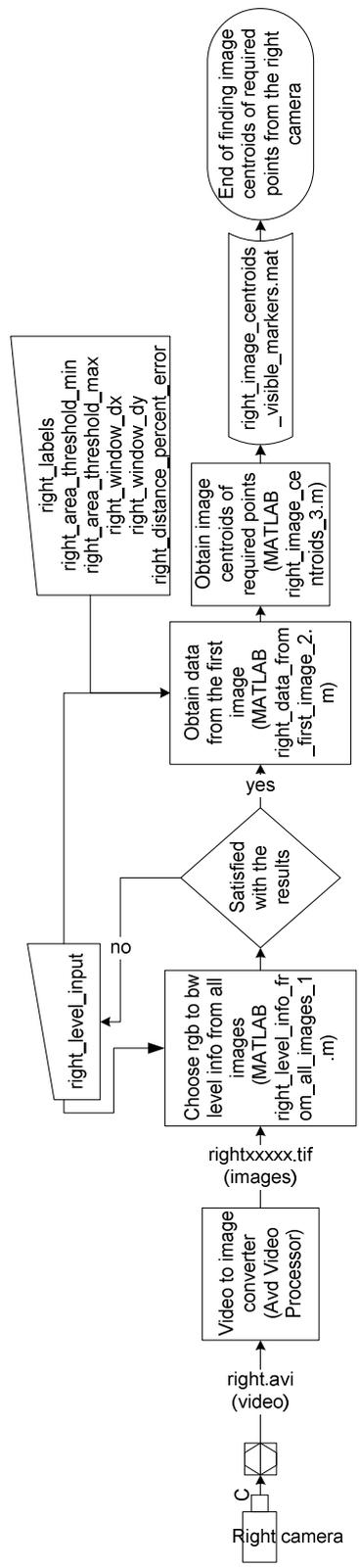
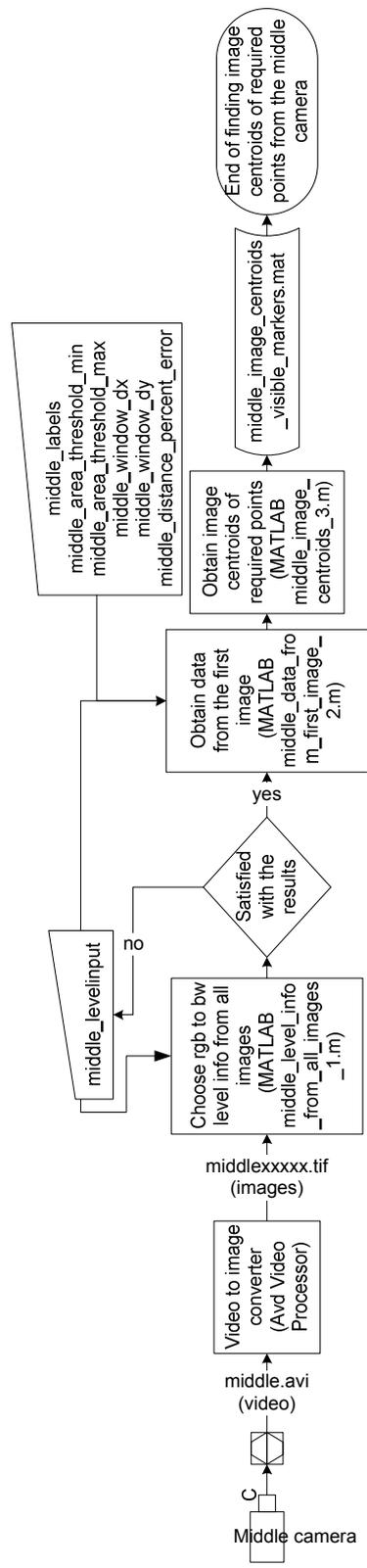
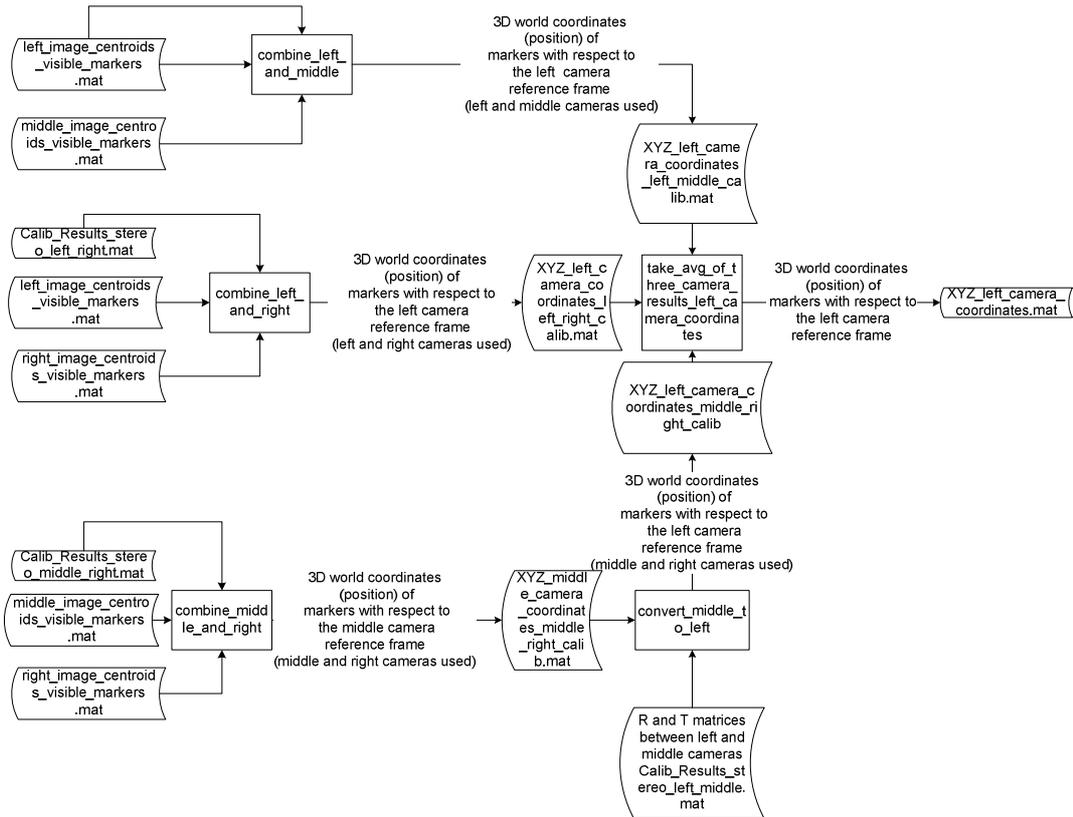


Figure B.5 Image centroids



## 5. Combination of image centroids results and stereo calibration parameters

combine\_image\_centroids\_and\_calibration.m



define\_visible\_markers.m

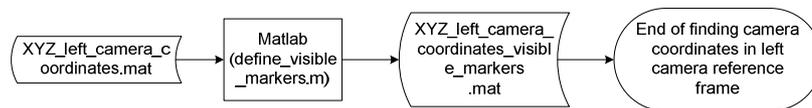


Figure B.6 Combine image centroids and calibration

## 6. Smoothing the position of visible markers

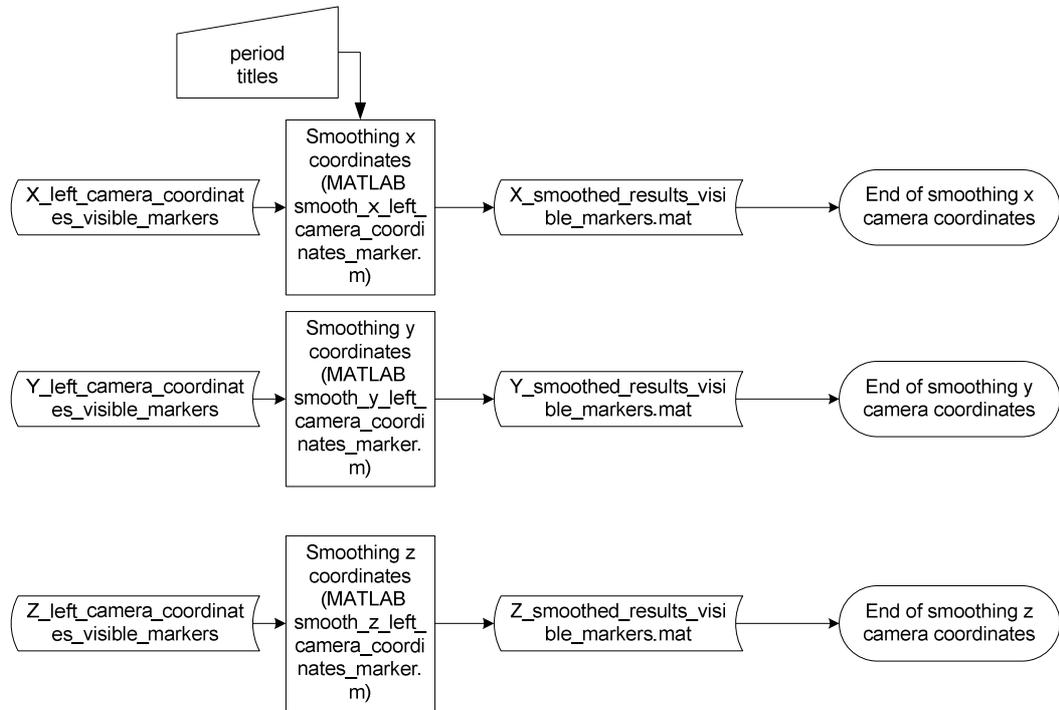
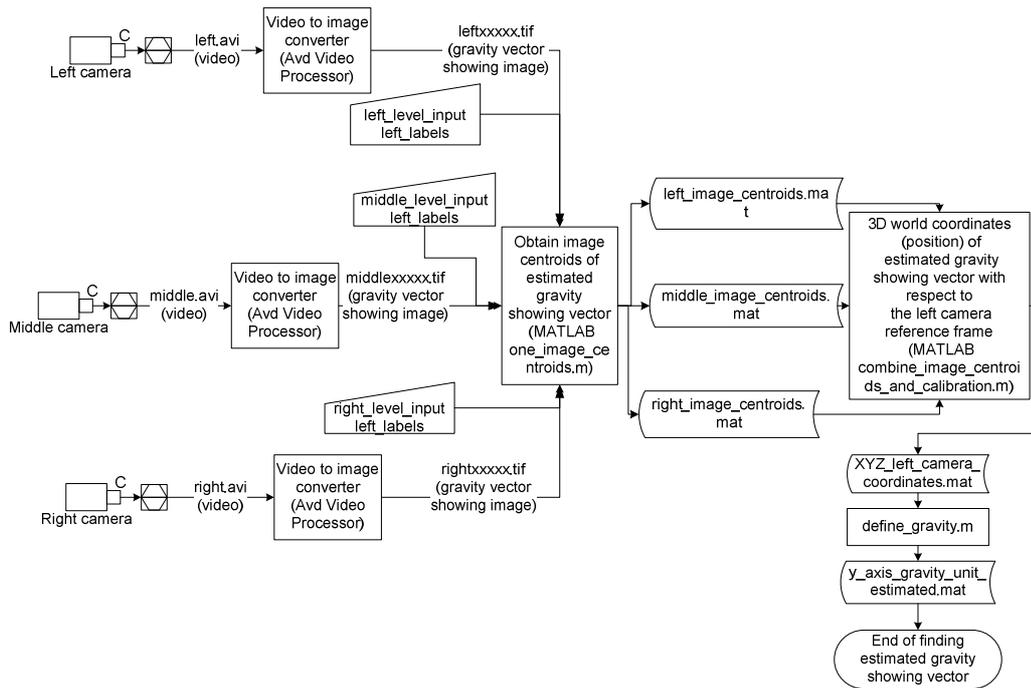


Figure B.7 Smoothing

## 7. Converting position of markers from the leftmost camera to motion reference frame

Defining estimated gravity showing unit vector on image and left camera reference frame



## Converting position of markers from the leftmost camera reference frame to motion reference frame

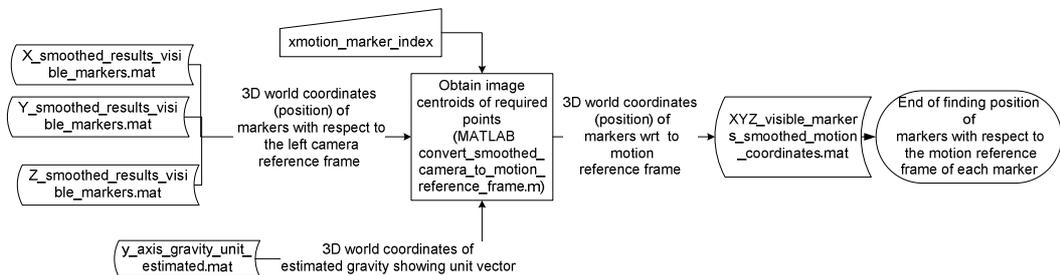
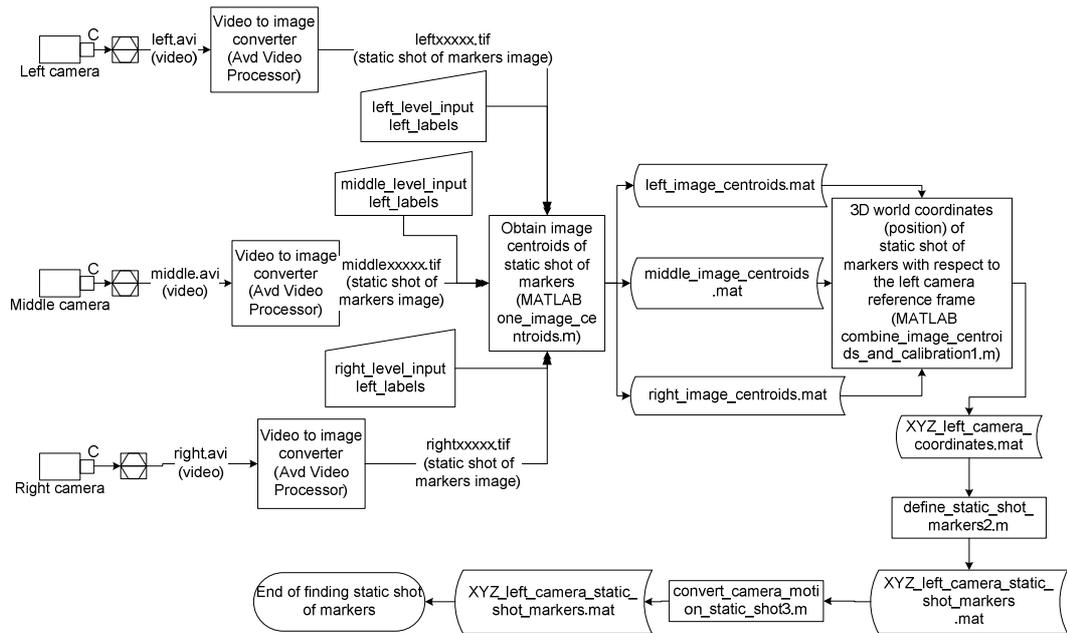


Figure B.8 Camera to motion reference frame

## 8. Registration of the position of markers (in case of rigidity of markers is required)

### Defining markers statically shot



### Registration of the position of markers in motion

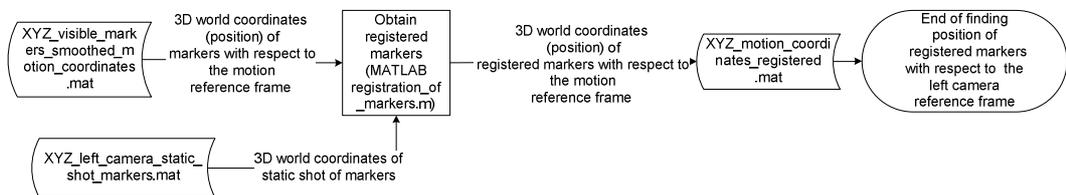


Figure B.9 Registration

## 9. Curve fitting to the position of required points with velocity and acceleration data



Figure B.10 Curve fitting

## 10. Performance measure of the motion of markers

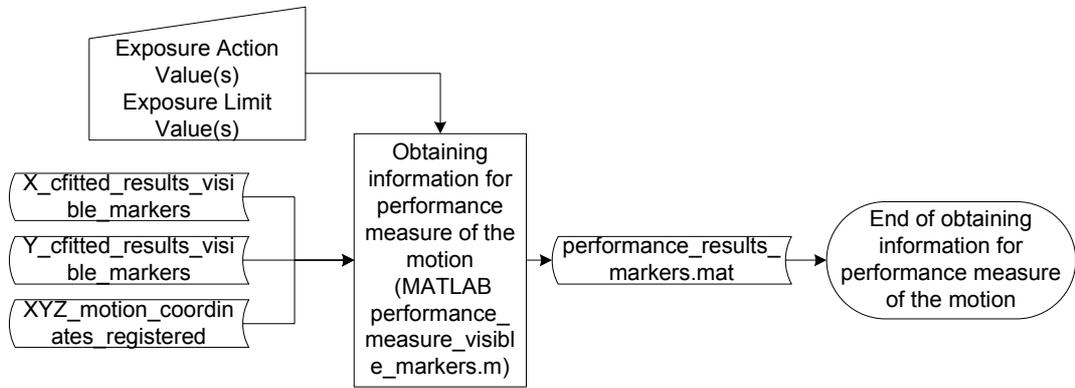


Figure B.11 Comfort measure

## 11. Reaching angular velocity and acceleration

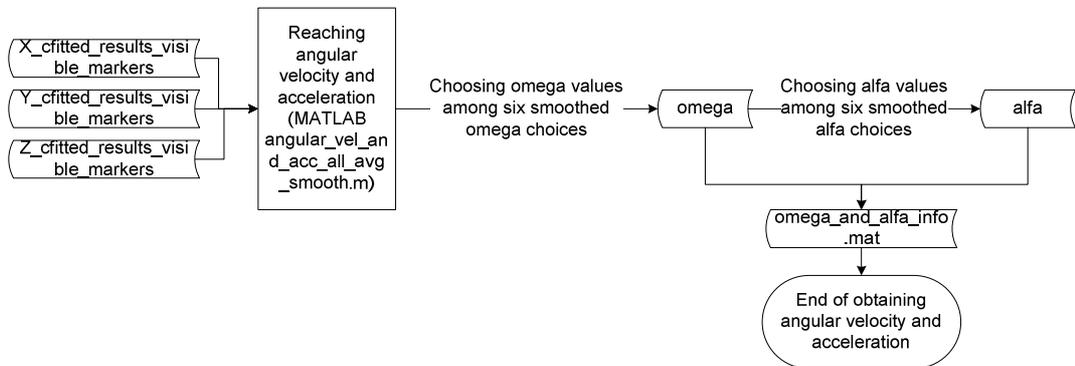
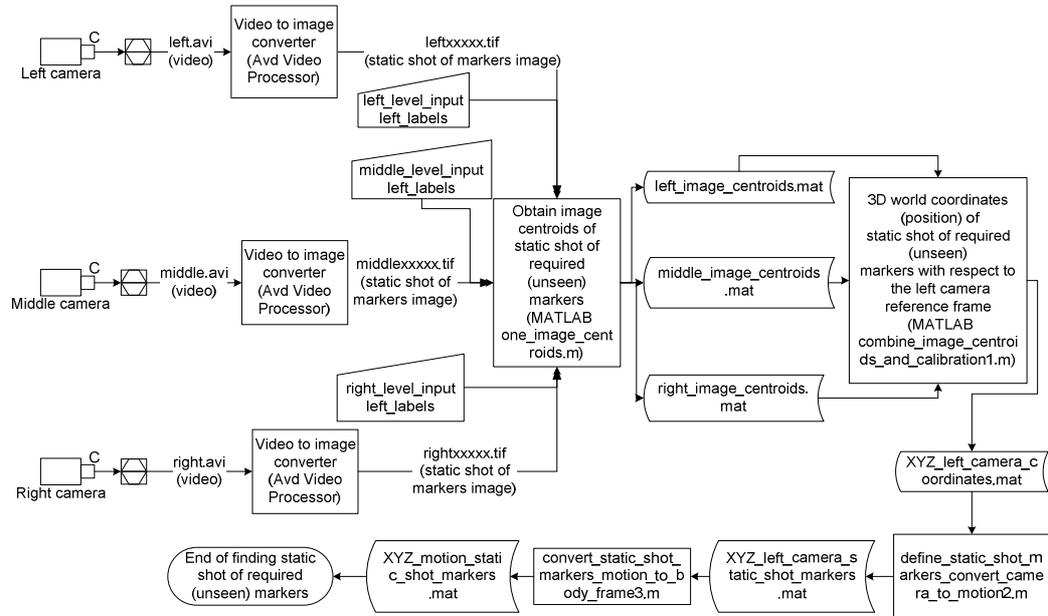


Figure B.12 Angular velocity and acceleration

## 12. Reaching required (unseen) markers data

### Defining required (unseen) markers statically shot



### Reaching position velocity and acceleration of required marker

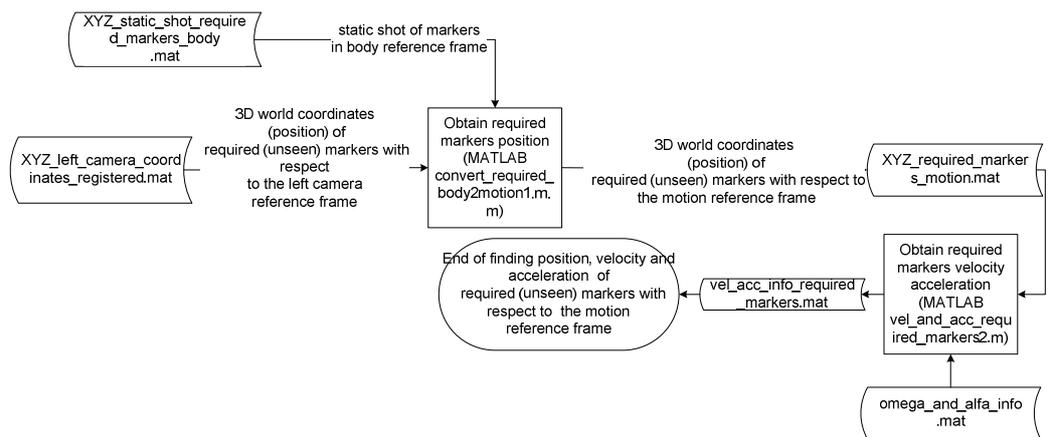


Figure B.13 Required (unseen) markers data

### 13. Comfort measure of the motion of required markers

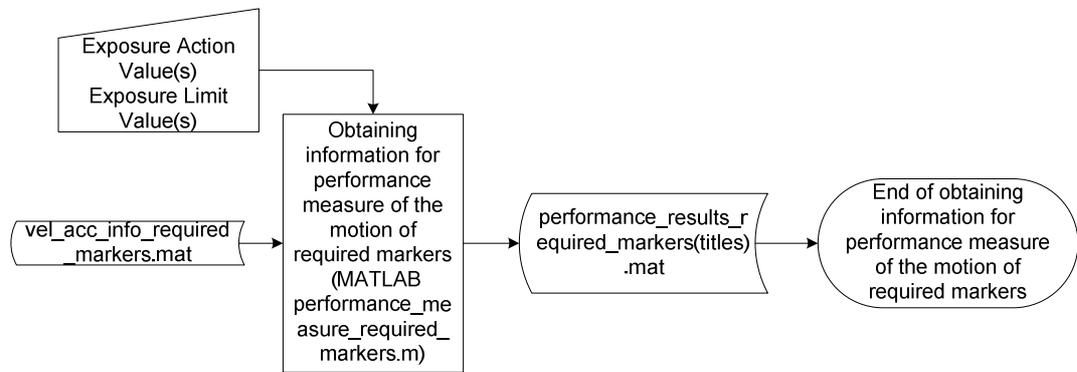


Figure B.14 Comfort measure of the motion of required markers

## APPEDIX C

### TECHNICAL DRAWINGS OF THE CALIBRATION MECHANISM

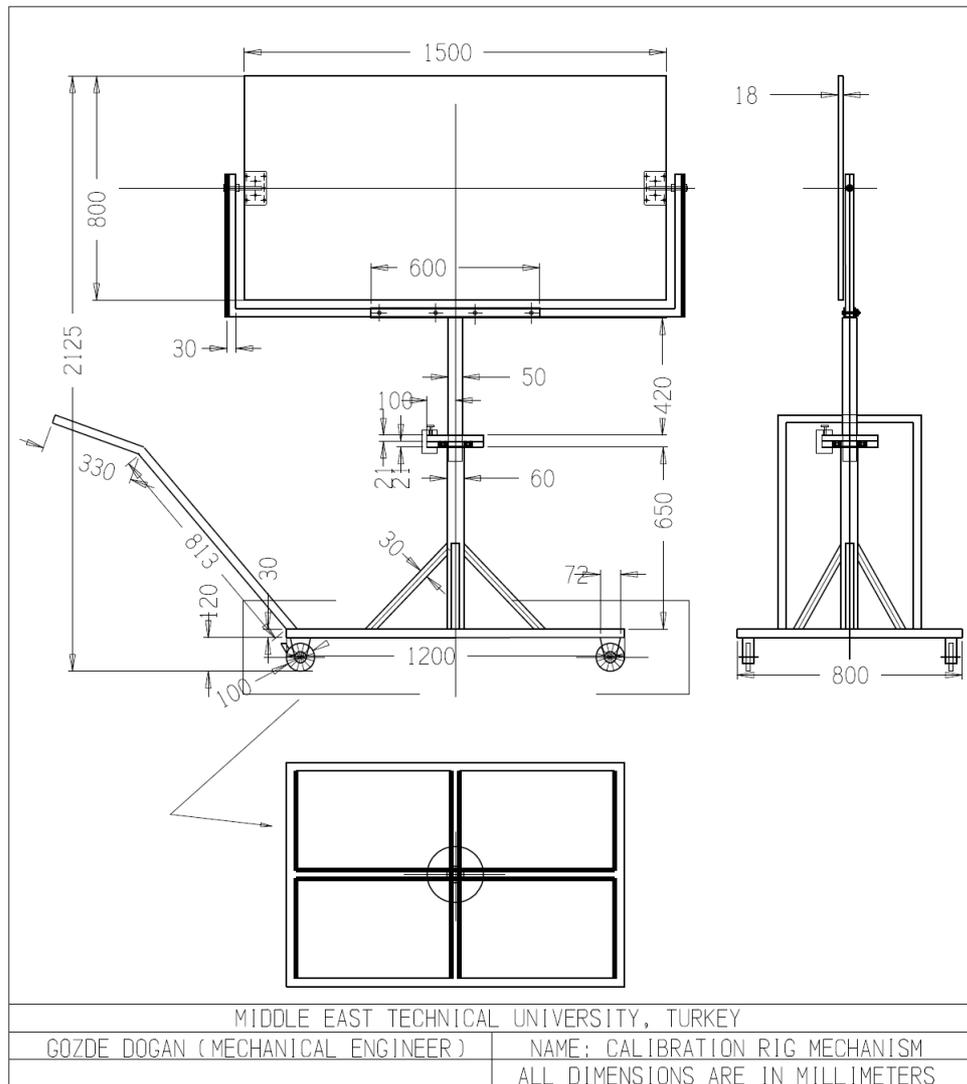


Figure C.1 Technical drawing of the whole calibration mechanism

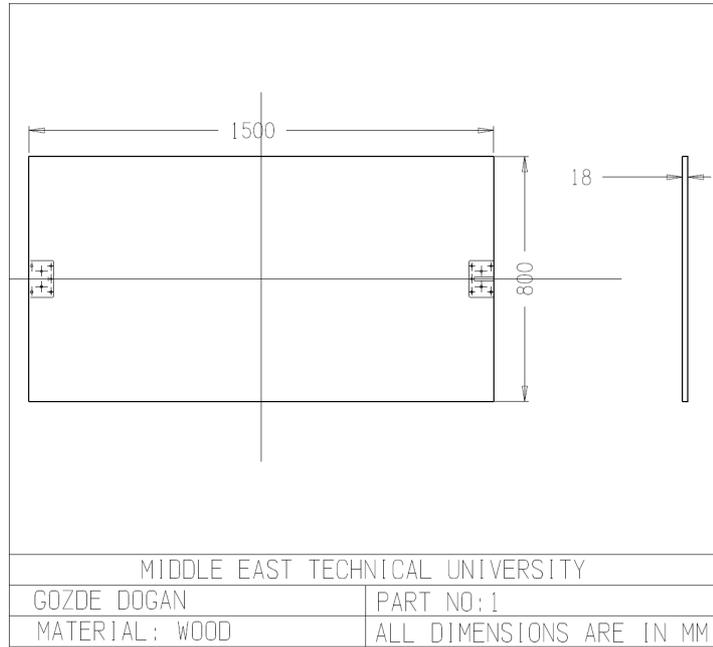


Figure C.2 Wooden part of the mechanism (Part 1)

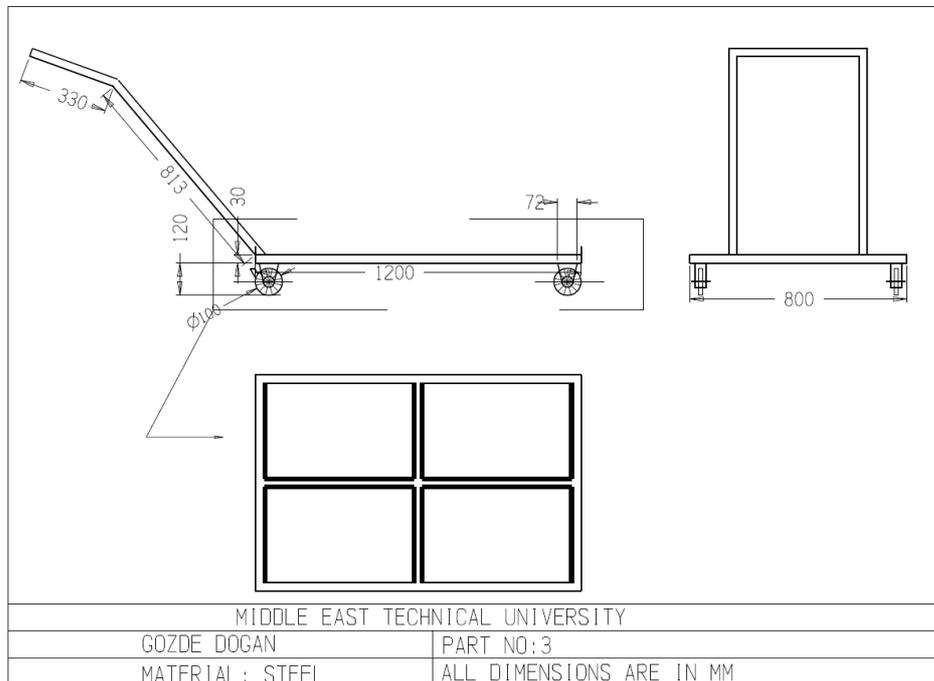


Figure C.3 Translating part of the mechanism (Part 2)

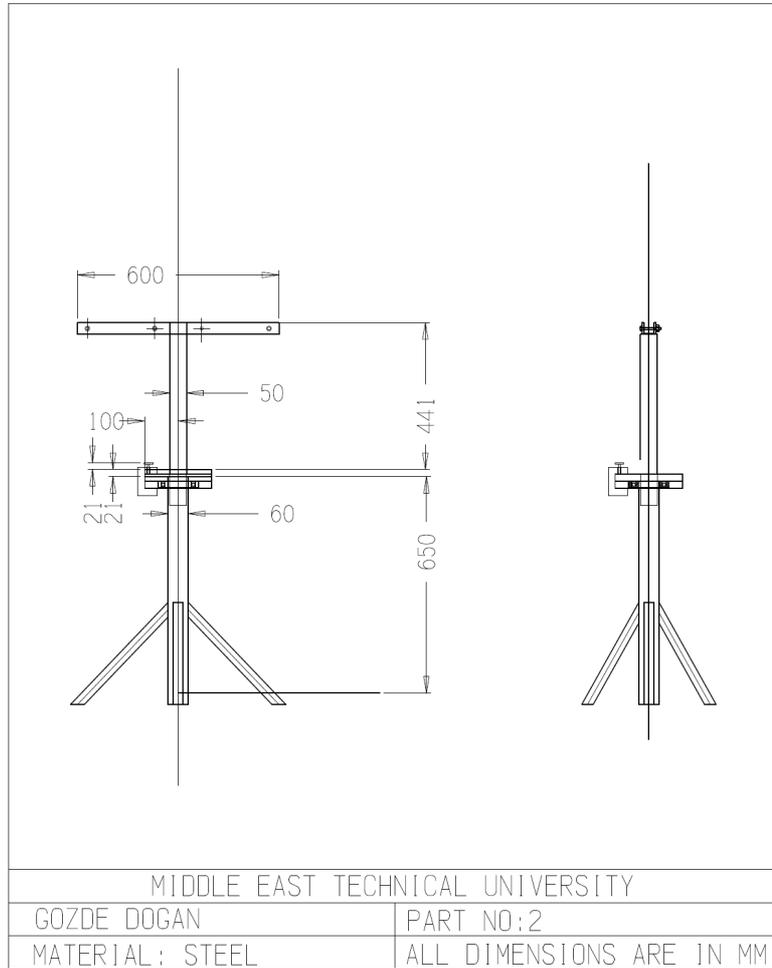


Figure C.4 Rotating part of the mechanism (Part 3)