

FEATURE EXTRACTION OF HONEYBEE FOREWINGS AND HINDLEGS
USING IMAGE PROCESSING AND ACTIVE CONTOURS

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

BY

AYŞEGÜL GÖNÜLŞEN

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR DEGREE OF
MASTER OF SCIENCE

IN

THE DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

JANUARY 2004

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. Mübeccel
DEMİREKLER
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Aykut KENCE
Co-Supervisor

Prof. Dr. Uğur HALICI
Supervisor

Examining Committee Members

Prof. Dr. Kemal LEBLEBİCİOĞLU (Chairman)

Prof. Dr. Uğur HALICI

Prof. Dr. Aykut KENCE

Assoc. Prof. Dr. Gözde Bozdağı AKAR

Dr. İlkay ULUSOY

ABSTRACT

FEATURE EXTRACTION OF HONEYBEE FOREWINGS AND HINDLEGS USING IMAGE PROCESSING AND ACTIVE CONTOURS

GÖNÜLŞEN, Ayşegül

MSc., Department of Electrical and Electronic Engineering

Supervisor: Prof. Dr. Uğur HALICI

Co-Supervisor: Prof. Dr. Aykut KENCE

January 2004, 235 pages

Honeybees have a rich genetic diversity in Anatolia. This is reflected in the presence of numerous subspecies of honeybee in Turkey. In METU, Department of Biology, honeybee populations of different regions in Turkey are investigated in order to characterize population variation in these regions. A total of 23 length and angle features belonging to the honeybee hindlegs and forewings are measured in these studies using a microscope and a monitor. These measurements

are carried out by placing rulers on the monitor that shows the honeybee image and getting the length and angle features. However, performing measurements in this way is a time consuming process and is open to human-dependent errors.

In this thesis, a “semi-automated honeybee feature extraction system” is presented. The aim is to increase the efficiency by decreasing the time spent on handling these measurements and by increasing the accuracy of measured hindleg and forewing features.

The problem is studied from the acquisition of the microscope images, to the feature extraction of the honeybee features. In this scope, suitable methods are developed for segmentation of honeybee hindleg and forewing images. Within intermediate steps, blob analysis is utilized, and edges of the forewing and hindlegs are thinned using skeletonization. Templates that represent the forewing and hindleg edges are formed by either Bezier Curves or Polynomial Interpolation. In the feature extraction phase, Active Contour (Snake) algorithm is applied to the images in order to find the critical points using these templates.

Keywords: Honeybee, Hindleg, Forewing, Segmentation, Blob Analysis, Skeletonization, Active Contour, Snake

ÖZ

İMGE İŞLEME VE AKTİF ÇEVİRİT KULLANARAK BALARI KANAT VE BACAKLARININ ÖZİNİTELİKLERİNİN ÇIKARILMASI

GÖNÜLŞEN, Ayşegül

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Uğur HALICI

Ortak Tez Yöneticisi: Prof. Dr. Aykut KENCE

Ocak 2004, 235 sayfa

Anadoludaki bal arıları zengin bir genetik çeşitliliğe sahiptir. Anadolu'da çok sayıda bal arısı alt türünün bulunması da bunu gösterir. ODTÜ Biyoloji Bölümünde, Türkiye'nin değişik bölgelerinde bulunan bal arılarının populasyon çeşitliliğini belirlemek amacı ile çalışmalar yapılmıştır. Bu çalışmalarda, mikroskop ve monitör kullanılarak bal arısı bacak ve kanadına ait toplam 23 tane

uzunluk ve açı özniteliği ölçülmüştür. Ölçümler, bal arısı görüntüsünü gösteren monitör üzerine yerleştirilen cetvel ile uzunluk ve açı karakterlerinin ölçerek gerçekleştirilmiştir. Ancak, ölçümlerin bu yolla yapılması zaman alıcı bir işlemdir ve insana bağımlı hatalara açıktır.

Bu tezde, bir “yarı-otomatik bal arısı öznitelik çıkarma sistemi” sunulmaktadır. Ölçümler sırasında harcanan zamanı azaltarak ve bacak ve kanat öznitelik ölçümlerinin doğruluğunu arttırarak verimliliği yükseltmek hedeflenmiştir.

Problem mikroskop görüntülerinin alınmasından öznitelik çıkarılmasına kadar bütün aşamaları içermektedir. Bu kapsamda bal arısı kanat ve bacak görüntülerinin bölütlenmesi için uygun yöntemler geliştirilmiştir. Ara safhalarda, birleşik parça analizi kullanılmış, kanat ve bacak kenarları iskelet çıkarma kullanılarak inceltirilmiştir. Bezier Eğrileri ya da Polinom Aradeğerleme yöntemi kullanılarak kanat ve bacak kenarlarını temsil eden şablonlar oluşturulmuştur. Öznitelik çıkarma aşamasında, kritik noktaların bulunması için, bu şablonlar kullanılarak görüntülere Aktif Çevrit (Yılan) algoritması uygulanmıştır.

Anahtar Kelimeler: Bal arısı, Bacak, Kanat, Bölütleme, Birleşik Parça Analizi İskelet çıkarma, Aktif Çevrit, Yılan

To My Family ...

ACKNOWLEDGMENTS

I would like to thank to my supervisor Prof. Dr. Uğur HALICI for her valuable supervision and support throughout the development and improvement of this thesis. This thesis would not have been completed without her guidance. I would like to thank to my co-supervisor Prof. Dr. Aykut KENCE and Prof. Dr. Kemal LEBLEBİCİOĞLU for their contributions in shaping the thesis.

I would like to thank to Assoc. Prof. Dr. Gözde Bozdağı AKAR and Erdem AKAGÜNDÜZ for their contribution on image capturing and to M. Ulaş ÇINAR as an expert for measuring honeybee features.

I have very special thanks to my husband Serkan for his valuable support, advices and comments.

Last, but not least, my sincere thanks go to my family for their incessant support, which has granted me the strength and courage to finish this thesis.

TABLE OF CONTENTS

ABSTRACT.....	iii
ÖZ.....	v
ACKNOWLEDGMENTS	viii
TABLE OF CONTENTS	ix
LIST OF TABLES	xiv
LIST OF FIGURES.....	xvii
LIST OF ALGORITHMS	xxiii
ABBREVIATIONS.....	xxiv
1 INTRODUCTION	25
2 BACKGROUND ON HONEYBEES	29
2.1 History and Evolution of Honeybees	29
2.2 Scientific Classification of Honey Bee (<i>Apis mellifera</i> L.).....	31
2.3 Biology of Honeybees.....	35
2.3.1 The Castes	35
2.4 Economic Value of Honeybees.....	36

2.4.1	Pollination	36
2.4.2	Honeybee Products.....	37
2.4.3	Beekeeping.....	37
2.5	Morphometric Study of Honey Bees.....	38
2.6	Morphometric Measurements	39
2.7	The materials and the methods.....	45
3	BACKGROUND ON IMAGE PROCESSING.....	50
3.1	Geometric Operations	50
3.1.1	Affine transformations	50
3.1.1.1	Forward and Backward Mapping.....	52
3.1.1.2	Horizantol and Vertical Reflection	53
3.2	Preprocessing and Segmentation.....	54
3.2.1	Spatial Filters	56
3.2.2	Low-Pass Spatial Filters.....	56
3.2.3	Rank Filters	57
3.2.4	Binarization	58
3.2.5	Morphological Operations	59
3.2.5.1	Fitting and Hitting.....	60
3.2.5.2	Erosion and Dilation	61
3.2.5.3	Opening and Closing.....	62
3.2.5.4	Thinning and Skeletonization	64

3.2.6	Labeling of Image Pixels	65
3.2.7	Blob Analysis	67
3.2.8	Edge Detection	68
3.3	Active Contour Model (Snake)	70
4	PREPROCESSING, SEGMENTATION AND FEATURE EXTRACTION OF HONEYBEE IMAGES	75
4.1	Preprocessing of Honeybee Images	76
4.1.1	User Interaction	76
4.1.2	Orientation of the Honeybee Images with respect to the Template Images.....	78
4.1.3	RGB to Gray Scale Conversion	81
4.1.4	Reducing Noise	83
4.1.5	Binarization	86
4.2	Blob Analysis	93
4.2.1	Selection of Hindleg Blob	94
4.2.2	Selection of Forewing Blob	96
4.3	Edge Detection of Honeybee Hindleg and Forewing Blobs	99
4.3.1	Edge Detection of Hindleg Blob	99
4.3.2	Edge Detection of Forewing Blob	101
4.4	The Features Extraction of Honeybee Hindleg and Forewing	102
4.4.1	Honeybee Forewing Feature Extraction	102

4.4.2	Honeybee Hindleg Feature Extraction	114
5	RESULTS	120
5.1	Honeybee Hindleg Feature Results	123
5.1.1	Femur Length (Fe)	123
5.1.2	Tibia Length (Ti).....	126
5.1.3	Metatarsus Length (ML)	128
5.1.4	Metatarsus Width (MW)	131
5.2	Honeybee Forewing Feature Results	133
5.2.1	Forewing Length (FL = L1+L2)	134
5.2.2	Forewing Width (FW).....	137
5.2.3	Cubital index a	139
5.2.4	Cubital index b	142
5.2.5	Distance c	145
5.2.6	Distance d.....	148
5.2.7	Angle EAB (A4)	150
5.2.8	Angle EBA (B4).....	153
5.2.9	Angle BDG (D7).....	156
5.2.10	Angle FGD (G18)	159
5.2.11	Angle OKF (K19)	162
5.2.12	Angle ROQ (O26)	165
5.2.13	Angle HEI (E9)	168

5.2.14	Angle PNJ (N23).....	171
5.2.15	Angle NJM (J16).....	174
5.2.16	Angle IJH (J10).....	177
5.2.17	Angle ILE (L13).....	179
5.3	Summary of the Difference Mean and Standard Deviation.....	182
5.4	Factors effecting the angle measurements	186
6	CONCLUSIONS	194
	REFERENCES.....	198
	APPENDICES	
	A. MATROX IMAGING LIBRARY	202
	B. THE USER’S GUIDE OF THE PROGRAM.....	222
	C. ALGORITHMS	233

LIST OF TABLES

2.1 - Scientific Classification of Honey Bee.....	31
2.2.- Honey Bee Subspecies.....	32
2.3 - Hindleg length parameters.....	39
2.4 - Forewing length parameters.....	40
2.5 - Forewing angle parameters.....	40
2.6 - Instruments used in image capturing.....	49
3.1 - Transformation coefficients for some simple affine transformations.....	52
4.1 - Method of formation of the forewing template edges.....	106
4.2 - Coefficients and neighborhood of snake for each contour.....	109
4.3 - Method of formation of the hindleg template edges.....	115
5.1 - Labeling of honeybee samples.....	121
5.2 - Measurement Results for Hindleg Fe (Femur Length).....	123
5.3 - Measurement Results for Hindleg Ti (Tibia Length).....	126
5.4 - Measurement Results for Hindleg ML (Metatarsus Length).....	129
5.5 - Measurement Results for Hindleg MW (Metatarsus Width).....	131
5.6 - Measurement Results for Forewing Length (FL).....	134

5.7 - Measurement Results for Forewing Width (FW)	137
5.8 - Measurement Results for Cubital index a.....	140
5.9 - Measurement Results for Cubital index b.....	143
5.10 - Measurement Results for Distance c	146
5.11 - Measurement Results for Distance d.....	148
5.12 - Measurement Results for Angle EAB (A4).....	151
5.13 - Measurement Results for Angle EBA (B4)	154
5.14 - Measurement Results for Angle BDG (D7)	157
5.15 - Measurement Results for Angle FGD (G18).....	160
5.16 - Measurement Results for Angle OKF (K19).....	163
5.17 - Measurement Results for Angle ROQ (O26)	166
5.18 - Measurement Results for Angle HEI (E9).....	169
5.19 - Measurement Results for Angle PNJ (N23)	172
5.20 - Measurement Results for Angle NJM (J16)	175
5.21 - Measurement Results for Angle IJH (J10)	177
5.22 - Measurement Results for Angle ILE (L13)	180
5.23 - Mean and Standard Deviation Summary of the Forewing Length Features	183
5.24 - Mean and Standard Deviation Summary of the Forewing Angle Features	184

5.25 - Mean and Standard Deviation Summary of the Hindleg Features	185
A.6.1 - Brief definitions of matrox functions used in the application.....	203

LIST OF FIGURES

1.1 - Sample images of honeybee Hindleg (a) and (b) Forewing sample images.	26
2.1 - Four length features belonging to the hindleg	42
2.2 - Eight length features belonging to the forewing.....	43
2.3 - Eight angle features belonging to the forewing	44
2.4 - Microscopic slides with honeybee hindleg and forewing.....	45
2.5 - TV capture card outlook.	47
2.6 - BNC to S-Video converter cable	48
2.7 - Setup used in capturing honeybee images	48
3.1- Binarization.....	59
3.2 - Binary image used to test fitting and hitting of structuring elements s_1 and s_2	61
3.3 - Morphological operations; erosion, dilation, opening, closing by 3×3 structuring element.....	63
3.4 - Thinning and skeletonization of a rectangle	65
3.5 - Labeling of image pixels: (a) original binary pattern (b) intermediate result after one pass; (c) corrected labels of neighboring regions after second pass.....	67

3.6 - 3 x 3 neighborhood that the search of the minimum energy occurs	71
4.1 - The reference points of the forewing image entered by the user.....	77
4.2 - The reference points of the hindleg image entered by the user	78
4.3 - (a) Forewing (b) Hindleg images with the reference points entered before the orientation	80
4.4 - Forewing image (a) Horizontal and vertical flip and (b) Rotation is applied	80
4.5 - Hindleg image (a) Horizontal and vertical flip and (b) Rotation is applied .	81
4.6 - 768x576, 24-bit gray scale bitmap honeybee images (a) Hindleg (b) Forewing of a honeybee	82
4.7 - Histogram of hindleg image a) before and (b) after applying low pass filter	84
4.8 - Histogram of forewing image a) before and (b) after applying low pass filter.....	85
4.9 - The resulting (a) hindleg (b) forewing image.....	86
4.10 - Regions of forewing image.....	90
4.11 - Histogram parameters used in finding the threshold of the hindleg.....	92
4.12 - (a) Hindleg and (b) Forewing images after binarization.....	93
4.13 - Binary hindleg image after (a) Small particles removed (b) Fill holes	95
4.14 - Selected hindleg blob after applying blob analysis	95
4.15 - Hindleg image after small blobs removed, all center of gravities and region defined by reference points	97
4.16 - Selected forewing blob after applying blob analysis.....	99
4.17 - Hindleg (a) edge detection (b) Skeletonization	100

4.18 - (a) Undesirable wing edges which consist of double edge (b) Desirable wing edges which consist of single edge	101
4.19 - Forewing edge formed with (a) Bezier Curve (b) Polynomial Interpolation	104
4.20 Straight edge formed with (a) Bezier Curve (b) Polynomial Interpolation	104
4.21 - Forewing template	105
4.22 - The result of the active contour algorithm applied to the wing image	111
4.23 - Example of two vertex points I (a) and (b) F and a nonvertex point G (c) from the edged forewing image	112
4.24 - The result of vertex point finding algorithm and correction of critical points by user	113
4.25 - Hindleg edge formed with (a) Bezier Curve (b) Polynomial Interpolation	116
4.26 - Hindleg template.....	117
4.27 - The result of the active contour algorithm applied to the hindleg image .	119
5.1 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Fe (Femur Length)	125
5.2 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Ti (Tibia Length).....	128
5.3 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Metatarsus Length (ML)	130
5.4 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Metatarsus Width (MW)	133

5.5 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Forewing Length (FL).....	136
5.6 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Forewing Width (FW).....	139
5.7 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Cubital index a	142
5.8 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Cubital index b.....	145
5.9 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Distance c.....	147
5.10 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Distance d.....	150
5.11 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle EAB (A4)	153
5.12 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle EBA (B4).....	156
5.13 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle BDG (D7).....	159
5.14 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle FGD (G18)	162
5.15 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle OKF (K19)	165

5.16 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle ROQ (O26).....	168
5.17 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle HEI (E9).....	171
5.18 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle PNJ (N23).....	174
5.19 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle NJM (J16).....	176
5.20 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle IJH (J10).....	179
5.21 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle ILE (L13).....	182
5.22 - A high quality forewing image.....	187
5.23 - High quality forewing image after applying edge detection and snake algorithms.....	187
5.24 - A low quality forewing image.....	188
5.25 - Low quality forewing image after applying edge detection and snake algorithms.....	188
5.26 - Manual LN distance measurement with L and N points are placed on actual places.....	190
5.27 - Manual LN distance measurement after applying binarization and thinning.....	191

5.28 - Manual LN distance measurement with L and N points are placed far from the actual places	192
B.1 - GUI of Honeybee Feature Extraction System.....	222
B.2 - Histogram window	225
B.3 - The hindleg template.....	227
B.4 - The forewing template	228

LIST OF ALGORITHMS

3.1 - Image rotation by backward mapping	53
3.2 - Pseudo code for snake (Greedy approach).	73
C.1 - Vertex Point Finding Algorithm	233

ABBREVIATIONS

BNC	: Bayonet Neill Concelman
BMP	: Bit Map
COG	: Center of Gravity
GUI	: Graphical User Interface
METU	: Middle East Technical University
MIL	: Matrox Imaging Library
PC	: Personal Computer
RGB	: Red Green Blue
VGA	: Video Graphics Array

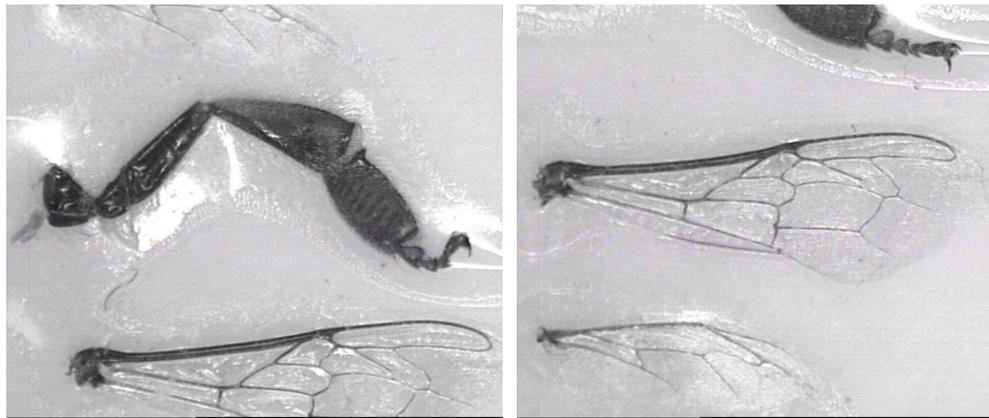
CHAPTER 1

INTRODUCTION

Honeybees in Anatolia have a rich genetic diversity; hence, there are numerous subspecies of honeybee in Turkey. Many studies have been conducted and some of them are still continuing in METU, Department of Biology to investigate the honeybee populations of different regions of Turkey. The aim of such researches is to characterize population variation of honeybees in these regions. The honeybee samples necessary for the studies are collected from different provinces, and dissected forewings and hindlegs of the honeybees are fixed on microscopic slides. Using these forewings and hindlegs, morphometric measurements are taken from the collected honeybee samples by using 8 length features and 11 angle features from forewing and 4 length features from hindleg.

In Biology Department, a system made of a microscope, a monitor, a camera control unit and a set of rulers are used to measure the features. The slides of the dissected honeybee forewings and hindlegs are placed under the microscope so that the camera control unit can transmit the image to the monitor. After this process, the morphometric measurements are carried out by placing the rulers on this monitor and measuring the length and angle information. However, performing measurements in this way is a time consuming process and is open to

human-dependent errors. In this study, the aim is to increase the efficiency of morphometric analyses by decreasing the time spent on handling these measurements and by increasing the accuracy of measured hindleg and forewing features with the help of a semi-automated system. The initial phase of this study requires capturing the honeybee hindleg and forewing images in digitized form. A PC with a TV card and a BNC to S-Video convertor cable is sufficient for picture digitization. The BNC to S-Video convertor cable is used for transmitting the video signals from the microscope to the PC. The detailed information on the setup for grabbing the images is given in Chapter 2. The honeybee images are captured as 24 bit RGB, 768x576 pixels and BMP format. Sample honeybee hindlegs and forewings are recorded in separate images as shown in Figure 1.1.



(a)

(b)

Figure 1.1 - Sample images of honeybee Hindleg (a) and (b) Forewing sample images

Once the images are recorded within the computer, the next step is to process them for feature extraction. First, image enhancement techniques are applied to reduce the noise of the images. Next, for the segmentation part, it is observed that applying a constant threshold to the images for binarization is not suitable since the lightening of the microscope may vary significantly. For that reason, an automatic and flexible thresholding algorithm is applied to the hindleg and forewing images.

The forewing images are separated into 3 regions. For each of these three regions, an automatic thresholding algorithm is applied to reach a specified percentage of white pixels vs. black pixels. For thresholding of the hindleg, there is no need to define regions on the image. An automatic thresholding algorithm is applied to the whole hindleg image. After the binarization process, the erroneous blobs (i.e. connected regions of pixels) around the actual forewing and hindleg blobs are eliminated from the images. Following that, edge detection and thinning (skeltonization) algorithms are applied to the hindleg blob to extract the edges. Once the binarized form of the forewing is obtained, only thinning (skeltonization) is applied to the forewing blob to extract the edges. After these processes single-pixel edges are obtained, so that the Active Contour (Snake) algorithm can be applied to the hindleg and forewing edges. Active Contour (Snake) algorithm is a method of moving the templates on the edges of the images. Applying snake algorithm is an important process which helps finding the critical points needed for measuring the length and angle features of the honeybees. All these processes and more are explained in Chapter 4.

For implementation part of the honeybee feature extraction system, Microsoft Visual C++ 6.0 is used as the software development environment with a professional image-processing library, Matrox Imaging Library (MIL). MIL is

used for the basic image processing techniques such as file I/O, binarization and morphological operations. The reason for using a professional library is to overcome the timing restrictions and reduce the development cycle of the system. More information about the MIL can be found in Appendix A.

The organization of the thesis is as follows. In Chapter 2, the history and evolution of honeybees, morphometric study of honeybees, the length and angle features of honeybees and the setup used to capture images are explained. In Chapter 3, the algorithms used for honeybee feature extraction system are discussed. In Chapter 4, application of the algorithms to the forewing and hindleg images and proposed methods for honeybee feature extraction is explained in detail. In Chapter 5, the automatic system measurements and the expert measurements are compared with the manual system measurements and the results are given. In Chapter 6, the conclusion is made and the future work is stated. Appendix A gives a brief explanation on the functions of Matrox Imaging Library and Appendix B is the user's guide of the software for the proposed system.

CHAPTER 2

BACKGROUND ON HONEYBEES

2.1 History and Evolution of Honeybees

Apidae family (family of honeybees) (Table 2.1) originated more than 70 million years ago and according to fossil records Apis genus (Table 2.1) appeared about 40 million years ago (upper Eocene in the early Cenozoic) [12]. As a result of natural selection, up-to-date forms of honey bees are obtained. Around 10,000 years ago, humans discovered how to domesticate first animals and then plants. With domestication, the need to hunt and gather was gone; the need to live in small groups and move with the rhythm of the seasons disappeared. The time to try to domesticate honey bee arrived [13]. Rock paintings from 8000 to 15,000 years old present a fascinating history of the process.

Apiculture has an old tradition in Anatolia. Excavations made at Çatal Höyük, demonstrate some rock paintings showing honey bees which collect nectar (Crane ,1976). In the Hittite code of tablets found at Boğazköy, dated about 1300 B.C., there are several paragraphs on beekeeping and they refer to an even older Codex. The value of a given weight unit of honey is mentioned as equal to the same unit of butter, and the price of a beehive was the same as that of a sheep [12].

Anatolia played an important role in evolution of many species because of its various climatic conditions, of its changing geologic structure from region to region and of its constitution a natural bridge between Africa, Europe and Asia. Honey bee (*Apis mellifera* L.) is one of these species.

There exist three basic hypothesis about the origin of honeybees and their dispersion over the world. First hypothesis states that, honeybees speciated in Southeast Asia or India (Rothenbuhler, 1968). The second hypothesis is that, honeybees speciated in Africa and then spread over Europe through Middle East [14]. The last hypothesis is that, honeybees speciated in the south of Caspian Sea, they then dispersed to Europe via Anatolia and to Africa via Arabian Peninsula [12]. In all of these hypothesis Anatolia is the key region for the dispersion of honeybees [1].

2.2 Scientific Classification of Honey Bee (*Apis mellifera* L.)

Honeybees belong to the genus *Apis* in the family Apidae, order Hymenoptera as indicated in Table 2.1. The four *Apis* species known are *Apis dorsata* (giant honey bee), *Apis florea* (dwarf honey bee), *Apis cerana* (Indian bee) and *Apis mellifera* (common honey bee). In the tropics, two species, *Apis dorsata* and *Apis florea* are evidently “old species” and live in the open area, suspending their honeycomb from tree branches. The third species of tropical bee, *Apis cerana* lives inside tree cavities and resembles to *Apis mellifera*. It was thought that *Apis mellifera* was developed from the more primitive *Apis cerana* [12].

Table 2.1 - Scientific Classification of Honey Bee

Kingdom	Animalia
Phylum	Arthropoda
Class	Insecta
Order	Hymenoptera
Family	Apidae
Genus	<i>Apis</i>

Table 2.1 (Continued)

Species	Dorsata
	Florea
	Cerana
	Mellifera

Table 2.2.- Honey Bee Subspecies

Name of subspecies	Living Area
<i>Apis mellifera adamii</i>	Cyprus
<i>Apis mellifera adansonii</i>	Africa
<i>Apis mellifera anatoliaca</i>	Anatolia
<i>Apis mellifera capensis</i>	Africa
<i>Apis mellifera carnica</i>	Europe
<i>Apis mellifera caucasia</i>	Anatolia, Caucasus
<i>Apis mellifera cocropia</i>	Europe
<i>Apis mellifera cypria</i>	Europe
<i>Apis mellifera iberica</i>	Europe
<i>Apis mellifera intermissa</i>	Africa
<i>Apis mellifera iran</i>	Middle East

Table 2.2 (Continued)

<i>Apis mellifera lamarkii</i>	Africa
<i>Apis mellifera ligustica</i>	Europe
<i>Apis mellifera litorea</i>	Africa
<i>Apis mellifera meda</i>	Middle East and Anatolia
<i>Apis mellifera mellifera</i>	Europe
<i>Apis mellifera sahariensis</i>	Africa
<i>Apis mellifera scutellata</i>	Africa
<i>Apis mellifera sicula</i>	Sicily, Italy
<i>Apis mellifera syriaca</i>	Middle East
<i>Apis mellifera unicolor</i>	Madagascar
<i>Apis mellifera yemetica</i>	Africa and Saudia Peninsula
<i>Apis mellifera monticola</i>	Africa

The natural distribution of honeybee, *Apis mellifera* L., ranges from Scandinavia to South Africa and also to the Himalayas. Geographic isolation and spatial differences in environmental factors may lead to the geographic differentiation within species. Within this diversity of habitats, the species has diverged into more than two dozen geographical races and the present state of taxonomic classification of honeybees is described in Table 2.1.

Since Turkey is located at the crossroads of Europe, Asia and the Middle East, a wide range of climates and habitats are found within its borders. Thus, numerous honeybee subspecies and ecotypes have been described from this region.

In 1941, Bodenheimer was the first to try a taxonomic classification of the honey bees of Anatolia, based on morphometric data [21]. At the end of his study, he separated the honey bees of Turkey into seven groups. Later, in 1953, Maa made the characterization of Anatolian honey bee (*Apis mellifera anatoliaca*) using their morphological properties [22]. He then published a formal taxonomic classification of Anatolian honey bees based on 3 museum specimens.

30 years after the studies of Maa, in 1983, Brother Adam, who classified living colonies and bees by appearance and behavior, reached the same conclusions as Bodenheimer. According to him, there exist mainly four races of honey bee in the West, Southeast, Northeast and Centre of Anatolia and the honey bees at the other regions are the hybrids of these four races. Brother Adam studied the behavior and performance of the Anatolian bees and the hybrids during his travels for over 30 years . He especially made attention to the adaptation of Anatolian honey bee towards extreme climatic conditions and to their energetic food-collecting activity.

Ruttner suggested that there are 3 different honeybee races in Turkey [12]: *Apis mellifera anatoliaca* in Central Anatolia, *Apis mellifera caucasica* in the Northeast

of Anatolia and *Apis mellifera meda* in the Southeast of Anatolia. Among these types, *Apis mellifera anatoliaca* shows wide distribution in Turkey. In addition to them, *Apis mellifera ligustica* in Trachia and *Apis mellifera syriaca* in Southeast of Turkey are reported [15].

In recent years, migratory beekeeping became very widespread in Turkey because of the availability of Turkish flora to produce different types of honey. Beekeepers spend the winter in temperate climatic zones and return in summer to Central and Eastern Anatolia. During these migrations, different honey bee races can hybridize with isolated honey bee populations. As a result of migratory beekeeping, low level of genetic variability detected in the Central Anatolia [16] [17]. The extensive practice of migratory beekeeping is now mixing all 3 races in Turkey. The genetic properties and local adaptations of honey bees are diminishing day by day and the honey bee gene pool is becoming homogen because of these uncontrolled genetic hybridizations [1].

2.3 Biology of Honeybees

2.3.1 The Castes

The honeybee community consists of three structurally different forms; the queen, the worker and the drone. In each hive, there are thousands of worker bees, about a hundred of drones, but a single queen. These castes are associated with different

functions in the colony. The queen is the only sexually productive female in the colony. She can lay up to 2000 eggs a day and she has an average lifespan of 1 to 3 years. Drones are male honeybees and their only function is to mate with the queen. Workers are all females but with undeveloped reproductive systems. A healthy colony may contain about 80,000 worker bees. Depending on age, they perform all tasks necessary for the support of the hive, such as; helping the nursery, feeding the larva, building work, storing the food, guarding the hive and collecting food [13].

2.4 Economic Value of Honeybees

2.4.1 Pollination

Flowers have their bright colors and nice smells to attract honeybees. Honeybees depend on flowers for food, while flowers depend on bees to help them make seeds. Taking pollen from the anther of one flower to the stigma of another flower is called “pollination”. Honey Bee is important in modern agriculture and in nature, providing pollination for many valuable crops and wild plants. Without honeybees to pollinate them, most plant species would perish.

Honeybees become the primary source of pollination for approximately one-fourth of all crops produced in the United States. The value of the crops that rely on such pollination has been estimated as high as 10 billion dollars annually in

this country. Many species of wild pollinators have disappeared from the land as their habitats have been destroyed or altered by humans. The honeybee is pollinator of many of the plants; its ecological and economical value in this regard is tremendous [18].

2.4.2 Honeybee Products

Honeybees collect nectar and pollen from flowers. Nectar is the bees' carbohydrate source, the food that provides the energy they need. Honeybees turn nectar into honey by adding the enzyme invertase. Pollen is the bees' protein source used for the most part to feed the growing young and to nurture the queen. The pollen also contains lipids, vitamins and minerals. Honeybees are the sole source of honey and pollen for human consumption.

Honeybees also produce beeswax, royal jelly and propolis. The last one is a gummy substance made from tree sap that has antibacterial properties. Honeybee venom is extracted for the production of antivenom therapy and is being investigated as a treatment for several serious diseases of the muscles, connective tissue and immune system, including multiple sclerosis and arthritis [18].

2.4.3 Beekeeping

Rich flora and suitable climate conditions of Turkey constitute an available environment for beekeeping. There are more than 10,000 families who are

performing apiculture as their main source of income, and approximately 150,000 families deal with this agricultural occupation as their additional source of income [19]. Since the working power is increasing in agricultural farms by technological developments, beekeeping is preferred for people who remain unemployed. Thus, the social aspect of the event and its economic contributions of it to the family budget are very important in our country [1].

2.5 Morphometric Study of Honey Bees

The honeybees of the world are widespread over their native range of Europe, Africa and Asia, and have been successfully transplanted to other regions and continents [12]. They have been well studied over most of their range, and have been assigned to many subspecies (races or geographical variants) based on morphometric analyses. However, understanding of honeybee morphological variation is incomplete [20]. By using morphological features, Bodenheimer [21], Maa [22], Adam [22], Karacaoğlu [24], Settar [25], and Kence and Darendelioğlu [16] tried to identify different honeybee species and to determine the phenotypic variation found in their populations.

Morphometric measurements are performed by using 8 length features and 11 angle features from forewing and 4 length features from hindleg. By this way, genetic diversity in honeybees from the regions of Turkey can be investigated [1].

2.6 Morphometric Measurements

In morphometry, there are 4 length features to be measured in each hindleg. These parameters are stated in Table 2.3 and shown in Figure 2.1. The 8 length features belonging to forewings are given in Table 2.4 and shown in Figure 2.2. The next step of morphometric study is the angle measurements of the honeybee samples that are shown in Figure 2.3. The 11 angle parameters should be found stated as in Table 2.5. The forewing features are indicated by Goetze in 1964 and by Alpatov in 1928; all hindleg features are determined by Alpatov in 1929; and 11 angles are presented by DuPraw in 1964 [12].

Table 2.3 - Hindleg length parameters

Hindleg Length Parameters	
1	Femur Length, FeL
2	Tibia Length, TiL
3	Metatarsus Length, ML
4	Metatarsus Width, MW

Table 2.4 - Forewing length parameters

Forewing Length Parameters	
1	Forewing Length, $FL = L1+L2$
2	Forewing Width, FW
3	Length 1, L1
4	Length 2, L2
5	Cubital index a
6	Cubital index b
7	Distance c
8	Distance d

Table 2.5 - Forewing angle parameters

Forewing Angle Parameters	
1	Angle EAB (A4)
2	Angle EBA (B4)
3	Angle BDG (D7)
4	Angle FGD (G18)

Table 2.5 (Continued)

5	Angle OKF (K19)
6	Angle ROQ (O26)
7	Angle HEI (E9)
8	Angle PNJ (N23)
9	Angle NJM (J16)
10	Angle IJH (J10)
11	Angle ILE (L13)

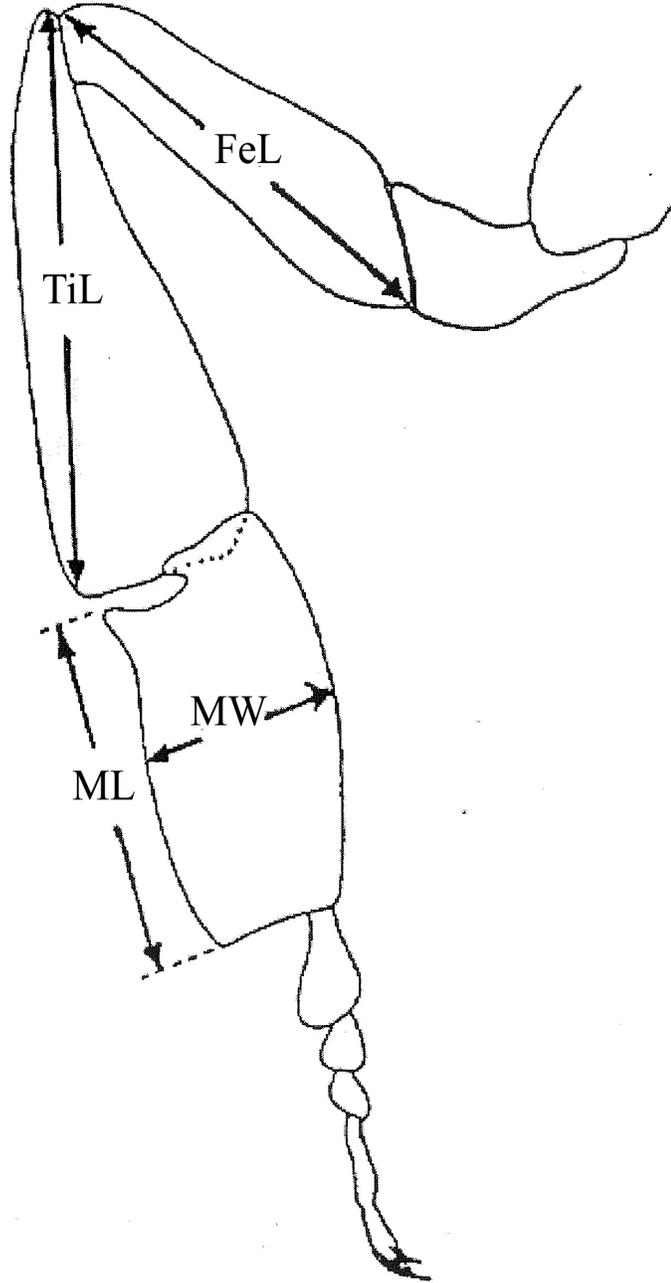
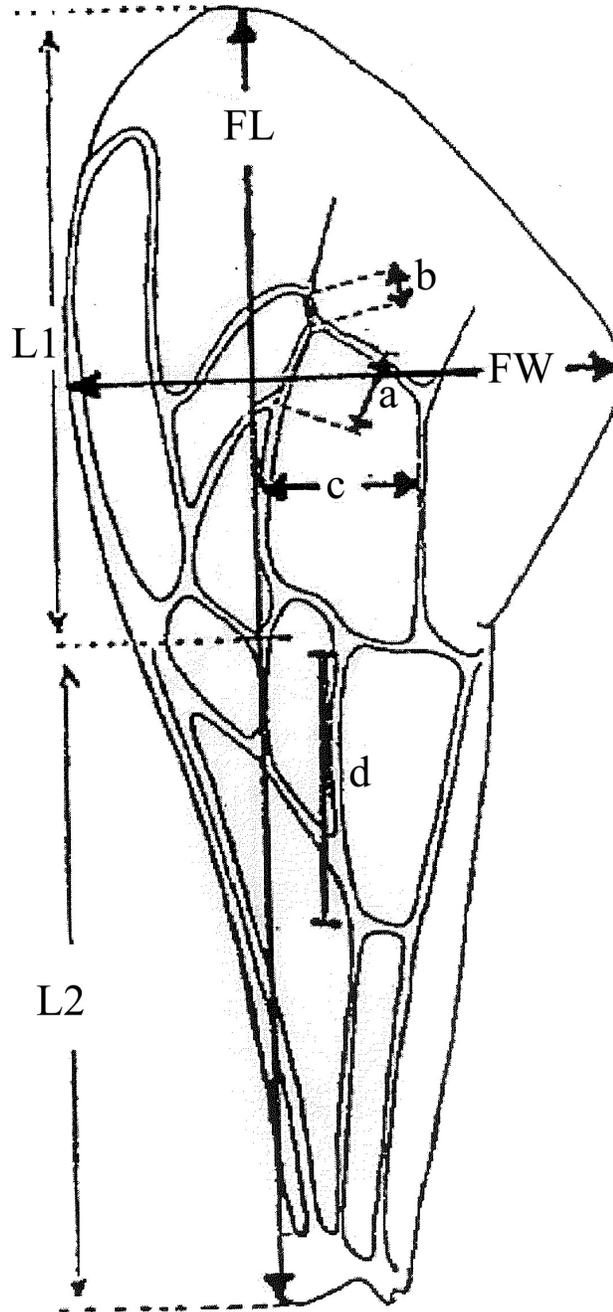


Figure 2.1 - Four length features belonging to the hindleg



.Figure 2.2 - Eight length features belonging to the forewing

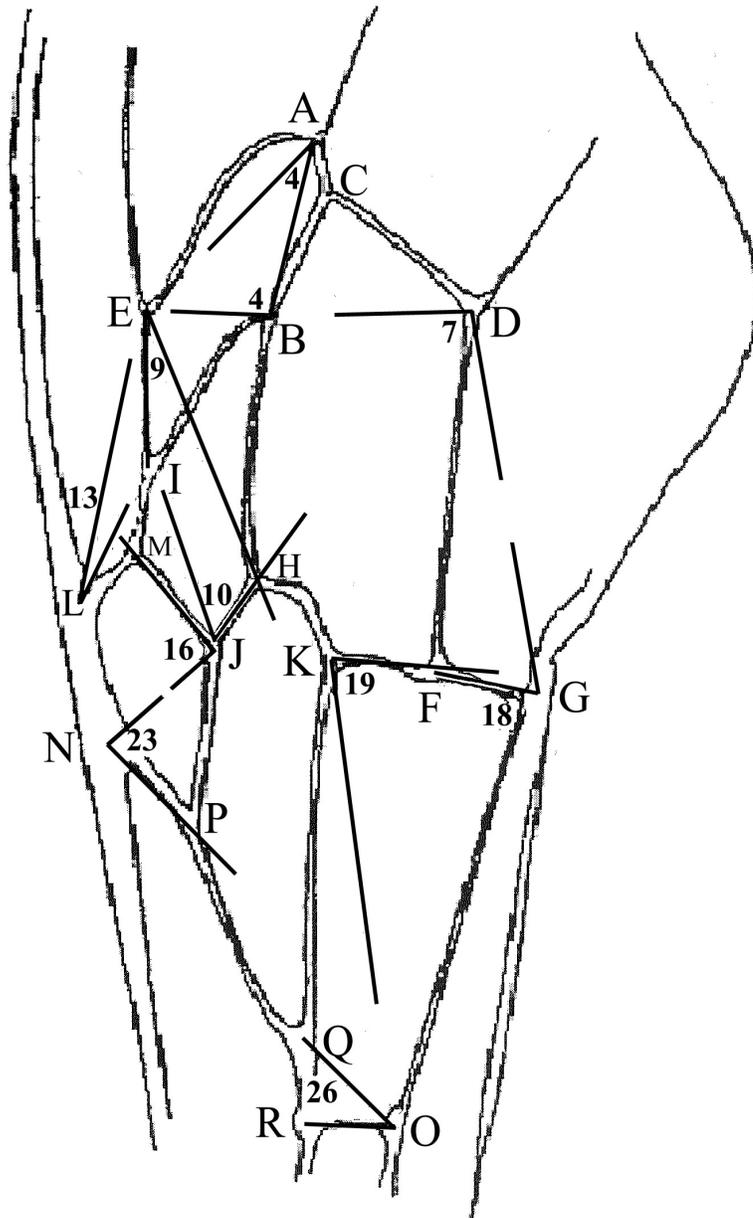


Figure 2.3 - Eight angle features belonging to the forewing

2.7 The materials and the methods

For investigating the honeybee populations of regions of Turkey in order to determine population variation in these regions, the honeybee samples are collected from the different provinces. The honeybee workers collected from each colony are put in separate plastic jars. Jars are labelled and information about each beekeeper is recorded. Until the samples are brought to the METU Biology Department laboratory, they are allowed to live with honey cakes that jars are containing. They are put to $-20\text{ }^{\circ}\text{C}$ and then, they are dissected into their wings, legs and thoraces. Dissected forewings and hindlegs of the honeybees are fixed on microscopic slides with transparent tape. A sample of these slides is given in Figure 2.4



Figure 2.4 - Microscopic slides with honeybee hindleg and forewing

The slides are put under the Olympus dissection microscope and JVC Camera Control Unit transmits the image to monitor. In Biology Department, morphometric measurements are carried out on this monitor with $1\text{mm}=3.08\text{ cm}$ (i.e. to obtain the real length values in mm, the measured length values in cm should be divided by 3.08) and 1.2X magnification and using only the left parts of the wings and legs using a set of rulers. The feature measurements are handled by putting the rulers on the hindleg or forewing image in the JVC monitor and getting the length and angle values of the features. However, doing the measurements in this way is time consuming and open to human errors. In this study, the aim is to increase the efficiency of morphometric analyses by decreasing the time spent on handling this measurements and increasing the accuracy of measured hindleg and forewing features. For that reason the hindleg and forewing images are captured by using the TV Card as seen in Figure 2.5.

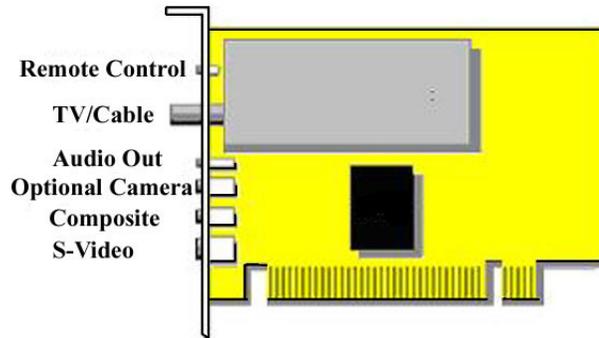


Figure 2.5 - TV capture card outlook.

A BNC to S-Video converter cable (Figure 2.6) is used for transmitting the video signals from the microscope to PC. The BNC side of the cable is connected to the video output of the monitor and S-Video side is connected to the S-Video input of the TV Card. After installing the Tview Card and TViewer Software on the PC, the images are captured with video standard PAL-B, D, G, H, I or PAL-NC, 24 bit RGB, image width 768, height 576 pixels and BMP image format. The setup used in capturing honeybee hindleg and forewing images are given in Figure 2.7. All the instruments used in capturing the honeybee images are listed in Table 2.6.

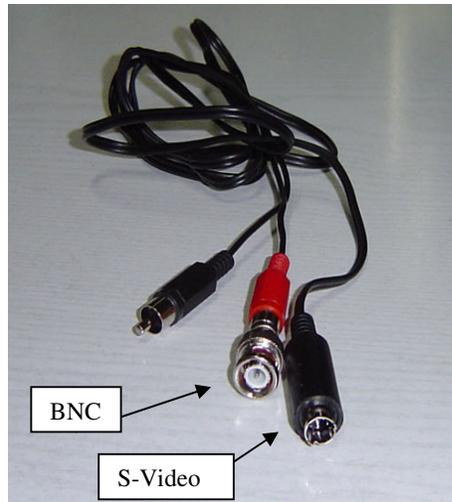


Figure 2.6 - BNC to S-Video converter cable



Figure 2.7 - Setup used in capturing honeybee images

Table 2.6 - Instruments used in image capturing

No	Instruments	Company
1	Binocular microscope	Olympus
2	Camera control unit	JVC
3	Monitor	JVC
4	TV Capture Card	Tview
5	IBM Compatible PC	System Requirements for using Tview card: Pentium 100 MHz or Higher One Available PCI Slot Minimum 8MB RAM (16MB Recommended) Microsoft Windows 95/98 SVGA Graphics Card Capability of Displaying at Least Hi Colors
6	BNC to S-Video converter cable	-
7	Microscopic slides with honeybee hindleg and forewing	-

CHAPTER 3

BACKGROUND ON IMAGE PROCESSING

This chapter is aimed to give the required background knowledge about preprocessing, object segmentation, and feature extraction steps.

3.1 Geometric Operations

Every image has its own set of geometric properties and it is sometimes necessary to transform image geometry, moving pixels around to change the relationships between image features. We might do this to remove distortions inherent in the imaging process, or to introduce a deliberate distortion that matches one image with another.

3.1.1 Affine transformations

A geometric transformation moves a pixel at coordinates (x, y) to a new position, (x', y') , given by a pair of transformation equations,

$$x' = T_x(x, y), \quad (3.1)$$

$$y' = T_y(x, y). \quad (3.2)$$

T_x and T_y are typically expressed as first order polynomials in x and y . They are linear in x and y , giving us an *affine transformation*,

$$x' = a_0x + a_1y + a_2, \quad (3.3)$$

$$y' = b_0x + b_1y + b_2. \quad (3.4)$$

This can be expressed in matrix form as,

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_0 & a_1 & a_2 \\ b_0 & b_1 & b_2 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3.5)$$

We can represent this matrix equation as 2 x 2 matrix multiplication as follows,

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_0 & a_1 \\ b_0 & b_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} .$$

Translation, scaling and rotation are all special cases of Equations 3.3 and 3.4. For instance a translation of 3 pixels down and 5 pixels to right is

$$x' = x + 5, \quad (3.6)$$

$$y' = y + 3. \quad (3.7)$$

The corresponding affine transformation matrix is

$$\begin{bmatrix} 1 & 0 & 5 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \end{bmatrix} .$$

Table 3.1 gives how the elements of the transformation matrix are computed for selected special cases of affine transformation.

Table 3.1 - Transformation coefficients for some simple affine transformations.

Transformation	a₀	a₁	a₂	b₀	b₁	b₂
Translation by $\Delta x, \Delta y$	1	0	Δx	0	1	Δy
Scaling by a factor s	s	0	0	0	s	0
Clockwise rotation through angle θ	$\cos \theta$	$-\sin \theta$	0	$\sin \theta$	$\cos \theta$	0

Any other combination of the transformations listed above is also an affine transformation.

3.1.1.1 Forward and Backward Mapping

In applying any kind of affine transformation, there are two approaches. One is described as *forward mapping*, involves iterating over each pixel of the input image, computing new coordinates for it using Equation 3.5 and copying its value to the new location. This approach is wasteful, as it potentially calculates many coordinates that do not lie within the bounds of the output image. Furthermore, each output pixel may be addressed several times or worse still, not at all. For example, a rotated image contains numerous ‘holes’ where no value can be computed for a pixel. To guarantee that a value is generated for every pixel in the output image, we must consider each output in turn and use the inverse transformation to determine the position in the input image from which a value must be sampled. This approach is known as the backward mapping. Every pixel

is calculated in this approach so that there will be no holes in the output image. Algorithm 3.1 shows the procedure for rotation by an angle θ using backward mapping. In this algorithm, the two problems namely that the calculated coordinates are real numbers, and that they might lie outside the bounds of the image are handled.

Algorithm 3.1 - Image rotation by backward mapping

```

Create an output image, g, of dimensions M x N
 $a_0 = \cos \theta$ 
 $a_1 = \sin \theta$ 
 $b_0 = -a_1$ 
 $b_1 = -a_0$ 
for all pixel coordinates  $x', y'$  in g do
     $x = \text{round}(a_0 x' + a_1 y')$ 
     $y = \text{round}(b_0 x' + b_1 y')$ 
    if (x, y) is inside f then
         $g(x', y') = f(x, y)$ 
    else
         $g(x', y') = 0$ 
    end if
end for

```

3.1.1.2 Horizontal and Vertical Reflection

Reflection is also a special case of affine transformation. It is mainly used as an aid to image visualization, but may be used as a preprocessing operator in much

the same way as rotation. The reflection operator geometrically transforms an image such that image elements, *i.e.* pixel values, located at position (x_1, y_1) in an original image are reflected about a user-specified image *axis* or image *point* into a new position (x_2, y_2) in a corresponding output image. Some commonly used transformations are the following:

- Reflection about a vertical axis of abscissa x_0 in the input image:

$$x_2 = -x_1 + 2x_0$$

$$y_2 = y_1$$

- Reflection about a horizontal axis of ordinate y_0 :

$$x_2 = x_1$$

$$y_2 = -y_1 + 2y_0$$

If (x_0, y_0) is not in the center of the input image, part of the image will be reflected out of the visible range of the image. Hence, most implementations use the reflection about origin in an image not to lose any part of it.

3.2 Preprocessing and Segmentation

Prior to manipulating and extracting information from an image, many applications require the best possible digital representation of the image to be obtained. Because the quality of the image affects the performance of the application. Several factors affect the quality of an image. These include:

1. **Random noise:** There are two main types of random noise:
 - a. **Gaussian noise:** When this type of noise is present, the exact value of any given pixel is different for each grabbed image; this type of noise adds to or subtracts from the actual pixel value.
 - b. **Salt-and-pepper noise** (also known as impulse or shot noise): This type of noise introduces pixels of arbitrary values (usually high-frequency values) that are generally noticeable because they are completely unrelated to the neighboring pixels.

Random noise can be caused, for example, by the camera or digitizer because electronic devices tend to generate a certain amount of noise. If the images were transmitted, the distance between the sending and the receiving devices also magnifies the random noise problem because of interference.

2. **Systematic noise:** Unlike random noise, this type of noise can be predicted, appearing as a group of pixels that should not be part of the actual image. This can be caused, for example, by the camera or digitizer or by uneven lighting. If the image was magnified, microscopic dust particles, on either the object or a camera lens, can appear to be part of the image.
3. **Distortions:** Distortions appear as geometric transforms of the actual image. These can be caused, for example, by the position of the camera relative to the object (not perpendicular), the curvature in the optical lenses, or a non-unity aspect ratio of an acquisition device [3].

There is Gaussian random noise in the hindleg and forewing images that are captured by the setup presented in Chapter 2 because the devices camera, monitor, TV card and BNC to S-Video cable can produce noise.

Dissected forewings and hindlegs of the honeybees are fixed on microscopic slides with transparent tape. These transparent tapes on the slides affect the quality of the image as they can be segmented with the wing and hindleg part of the image undesirably.

In this subsection, the preprocessing of the images as low pass filtering, binarizing, opening, closing, edge detection up to feature extraction phase will be discussed.

3.2.1 Spatial Filters

One of the basic and powerful steps in preprocessing is applying a proper filter to reduce the noise in the image. Spatial filtering provides an effective method to reduce noise. Spatial filtering operations determine each pixel's value based on its neighborhood values. They allow images to be separated into high-frequency and low-frequency components. There are two main types of spatial filters that can remove noise: low-pass filters and rank filters.

3.2.2 Low-Pass Spatial Filters

Low-pass spatial filters are effective in reducing Gaussian random noise (and high-frequency systematic noise), provided that the noise frequency is not too close to the spatial frequency of significant image data. These filters replace each pixel with a weighted sum of each pixel's neighborhood. Note, these filters have a side-effect of smoothing or blurring the image and removing edge information [3].

Any convolution kernel whose coefficients are all positive will act as a low pass filter. In the simplest case, below a normalized, 3 x 3 kernel is shown.

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.8)$$

It is clear that what this kernel does; pixel values from the neighborhood are summed without being weighted, and the sum is divided by the number of pixels in the neighborhood. Applying larger kernels produce more pronounced smoothing. A high degree of smoothing can also be achieved through repeated application of a small kernel to an image [4].

3.2.3 Rank Filters

Convolution is not the only way of carrying out spatial filtering. Non-linear techniques also exist. A number of these are known collectively as ‘order-statistic’ filters or rank filters. The idea behind rank filtering is simple. We compile a list of the gray levels in the neighborhood of a given pixel, sort this into ascending order and then select a value from a particular position in the list to use as the new value for the pixel. The new values must be stored in another image; we cannot perform the operation in place. The most common filter is the *median filter*, in which we select the middle-ranked value from a neighborhood as our output value. For a 3 x 3 neighborhood, the middle value is fifth in the list of sorted gray levels; for an $n \times n$ neighborhood n being odd, the middle value is at position $(n^2 / 2 + 1)$ [4].

Rank-filter operations are more suitable for removing salt-and-pepper type noise since they replace each pixel with a pixel in its neighborhood rather than a weighted sum of its neighborhood. The weighted sum generally creates a blotchy effect around each noise pixel [3].

The hindleg and forewing images do not have salt-and-pepper type noise but Gaussian random noise, for that reason, we only use Low-pass spatial filter in our system, rank-filter operation is unnecessary.

3.2.4 Binarization

Images which, for example, contain light objects on a dark background or dark objects on a light background can be segmented by means of a simple Binarization operation. This is the most effective way to extract the objects from the background. Binarization, which is also called as shareholding, reduces the images to two grayscale values: 0 and the maximum value in the image as 255.

A binarization operation is performed by comparing each pixel value in the image against one or two specified threshold values and mapping the pixels in that threshold range to 0 (or 255), and the pixels out of that range to 255 (or 0). The range can be between 0 - threshold_value_1 or between two threshold values threshold_value_1 and threshold_value_2. In the case that there are objects on a uniform background whose gray level values are all higher (all lower) than the gray level value(s) of the background, binarization with a threshold value is effective to extract the objects from the background.

In the case where there are objects having different gray level values on uniform background, to extract an object, whose gray level value is known, binarization with two threshold values is preferred. The threshold values are chosen as to include the gray level value of the object required to be extracted.

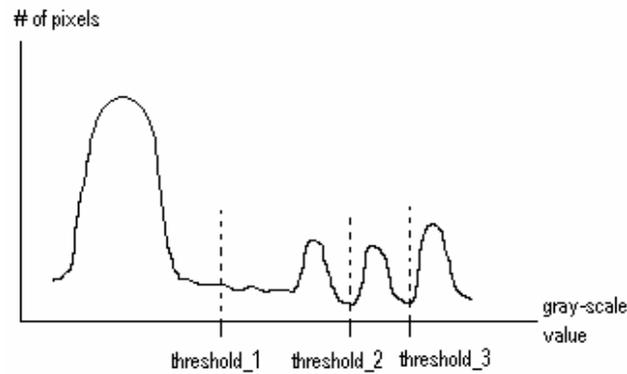


Figure 3.1- Binarization

For example, in the above figure it can be predicted that there is at least three objects having different gray-scale values and a background. To extract all the objects from the background, binarization with threshold_1 could be enough. If the objects having the lowest area are required to be extracted, the binarization should be applied between threshold_2 and threshold_3.

3.2.5 Morphological Operations

The binary images produced by simple segmentation techniques such as binarization may contain numerous imperfections caused by noise, texture or inaccurate specification of a threshold. Morphological operations can remove these imperfections.

Morphological image processing operates by passing a *structure element* over the image. Structuring element can be of any size, and it can contain any complement

of 1's and 0's. At each pixel position, a specified logical operation is performed between the structuring element and the underlying binary image. The binary result of that logical operation is stored in the output image at that pixel position. The effect created depends on the size and content of the structuring element and on the nature of the operation.

3.2.5.1 Fitting and Hitting

In a binary image, the structuring element is said to be *fit* the image if, for each of the structuring element's pixels that is set to 1, the corresponding image pixel is also 1. Similarly, a structuring element is said to intersect, or *hit*, an image if, for at least one of the structuring element's pixels that is set to 1, the corresponding image pixel is also 1. For fit and hit operations, we ignore image pixels for which the corresponding structuring element pixel is 0.

For example, suppose we have two 3 x 3 structuring elements

$$s_1 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}, \quad s_2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

These structuring elements are positioned over 3 x 3 neighborhoods labeled A, B and C in the image of Figure 3.2. Both s_1 and s_2 fit the image at A. However, only s_2 fits the image at B, and neither s_1 nor s_2 fit at C. We can see that both s_1 and s_2 hit the image in neighborhood of A and B. However, at C, only s_1 hits the image.

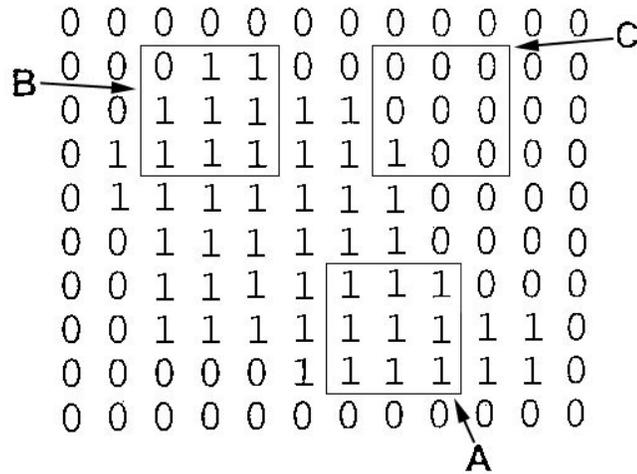


Figure 3.2 - Binary image used to test fitting and hitting of structuring elements s_1 and s_2

3.2.5.2 Erosion and Dilation

Erosion and dilation operations are the basic morphological operations.

Erosion is the process of eliminating all the boundary points from an object, removing unwanted pixels and the small particles from the image. Pixels are eroded from both the inner and outer boundaries of the regions, so erosion will enlarge the holes enclosed by a single region as well as making the gap between different regions larger. The erosion of an image f by a structuring element s is denoted by $f \otimes s$. To compute the erosion, we position s such that its origin is at image pixel coordinates (x, y) and apply the rule

$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ fits } f, \\ 0 & \text{otherwise,} \end{cases}$$

Dilation is the process of adding layers to objects or particles, enlarging them in area. Pixels are added to both the inner and outer boundaries of the regions, so dilation will shrink the holes enclosed by a single region as well as making the gap between different regions smaller. The dilation of an image f by a structuring element s is denoted by $f \oplus s$. To compute the dilation, we position s such that its origin is at image pixel coordinates (x, y) and apply the rule

$$g(x, y) = \begin{cases} 1 & \text{if } s \text{ hits } f, \\ 0 & \text{otherwise,} \end{cases}$$

Morphological operations can also be used for edge detection. It might seem that the simple difference between a dilated and eroded signal could define an edge, but this method is very noise sensitive.

3.2.5.3 Opening and Closing

The process of erosion followed by dilation is called *opening*. It has the effect of eliminating small and thin objects, breaking objects at thin points, and generally smoothing the boundaries of larger objects without significantly changing their area. Opening is defined by

$$f \circ s = (f \otimes s) \oplus s$$

The process of dilation followed by erosion is called *closing*. It has the effect of filling small and thin holes in objects, connecting nearby objects, and generally smoothing the boundaries of objects without significantly changing their area. Closing is defined by

$$f \bullet s = (f \oplus s) \otimes s$$

Often, when noisy images are segmented by thresholding (binarization), the resulting boundaries are quite ragged, the objects have false holes and the background is peppered with small noise objects. Successive openings and closings can improve the situation markedly. Sometimes several iterations of erosions followed by the same number of dilations, produces the desired effect.

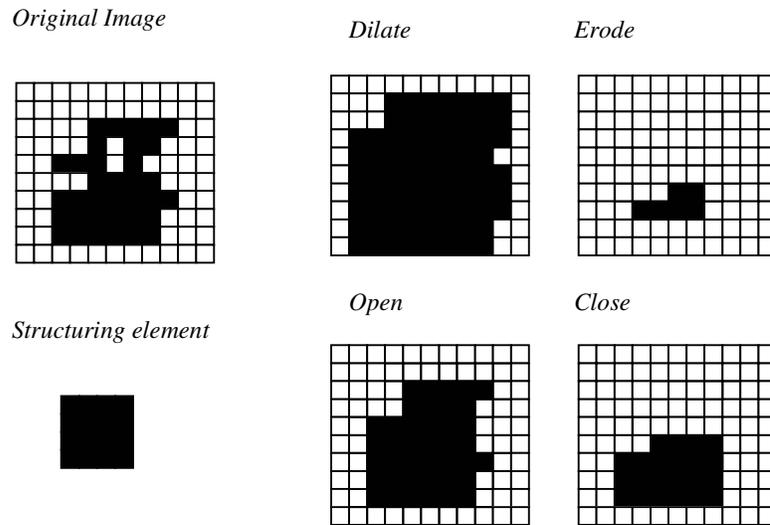


Figure 3.3 - Morphological operations; erosion, dilation, opening, closing by 3x3 structuring element

3.2.5.4 Thinning and Skeletonization

Thinning is an erosion-based process that does not break connected objects. The first step is normal erosion, but it is conditional; that is, pixels are marked as candidates for removal, but are not actually eliminated. In the second pass, those candidates that can be removed without destroying connectivity are eliminated, while those that cannot are retained.

An operation related to thinning is skeletonization. Skeletonization of an image is obtained by a medial axis transform, where the medial axis is the locus of points such that any medial point is equidistant to at least two points on the boundary. Mathematically, the medial axis can be found by finding the largest circle that can completely fit into the object and still kisses at least two boundary points. The center of the circle partly defines one of the points along the medial axis. Then find the next smaller circle that just fits inside the object, and the center points of all such kissing circles define the entire medial axis.

The primary difference between two thinning algorithm is that the medial axis skeleton extends to the boundary at corners, while the skeleton obtained by thinning does not. We can see this difference from the example of thinning and skeletonization of a rectangle in Figure 3.4.

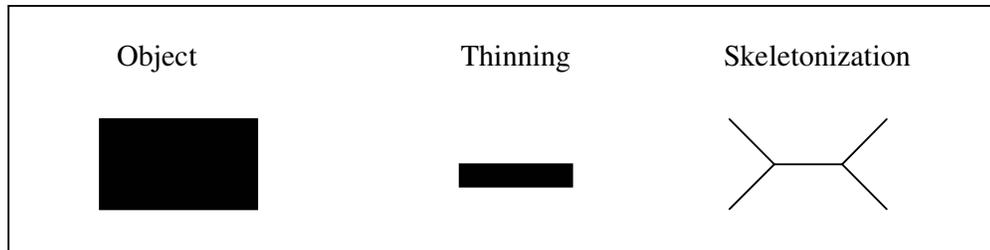


Figure 3.4 - Thinning and skeletonization of a rectangle

3.2.6 Labeling of Image Pixels

If a segmented image contains P objects which are to be analyzed separately, the image pixels corresponding to these different objects must first be supplied with P different labels. The aim of this is to assign the image pixels in each object area a uniform identification number i starting from 1 to P .

Several objects can be labeled simultaneously in the following manner. If it is assumed that objects are represented by 1 and the background by 0, we start looking from the upper left-hand corner of the image for object pixels in the first row of the image. The first object pixel found will have the label 1. If the next pixel to the right is similarly an object point, this also gets the label 1 because it belongs to the same object, and so on. The adjacent pixels in this row which do not belong to an object are labeled as background with 0. The next related sequence of object pixels in this row is assigned the label 2, and so on. The process continues in this way, labeling further rows of pixels but taking into account the labels of the pixels in preceding rows. That is, if an object pixel in the

row under investigation has a neighbor an object pixel that is already labeled, then this pixel is assigned the same label. Object pixels without neighbors that have already been labeled are assigned the next available unused label. If an unlabelled object pixel has two neighbors with different labels in the previous row, it is assigned the label of smaller value and a note is made in a table that the two labels define one and the same region. If necessary, this tabular information can be used in a subsequent processing stage to unify the labels of related object regions [5]. Figure 3.5 shows an example of labeling a simple binary pattern with two objects, both before (Figure 3.5 (b)) and after (Figure 3.5 (c)) the post processing step. The product of the labeling process depends on the definition of adjacency. In the four-neighborhood case the four pixels with the coordinates $(m + 1, n)$, $(m - 1, n)$, $(m, n + 1)$ and $(m, n - 1)$ are defined as the neighbors of the pixel with coordinates (m, n) . In the eight-neighborhood case the pixels with coordinates $(m + 1, n + 1)$, $(m + 1, n - 1)$, $(m - 1, n + 1)$ and $(m - 1, n - 1)$ are also treated as neighboring pixels. The eight-neighborhood case was assumed in the example in Figure 3.5.

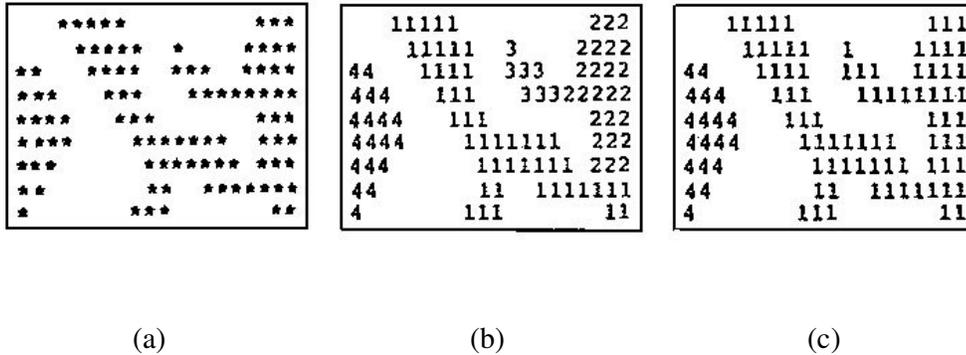


Figure 3.5 - Labeling of image pixels: (a) original binary pattern (b) intermediate result after one pass; (c) corrected labels of neighboring regions after second pass

3.2.7 Blob Analysis

The connected regions of pixels in an image are commonly known as *blobs*. Blobs are areas of touching pixels that are in the same logical pixel state. This pixel state is called the foreground state, while the alternate state is called the background state. Typically, the background has the value zero and the foreground is 1.

By labeling the connected regions, we find and identify each blob in an image and after that, the selected features of the blobs (i.e. area of a blob) can be easily calculated.

In many applications, we are interested only in blobs whose features satisfy certain criteria. Since computation is time-consuming, blob analysis is often performed as an elimination process whereby only blobs of interest are considered in further analysis.

The steps involved in feature extraction are:

1. Analyze an image and exclude or delete blobs that don't meet determined criteria.
2. Analyze the remaining blobs to extract further features and determine their criteria.

We shall repeat these steps, as necessary, until we have all the blob measurement results needed.

3.2.8 Edge Detection

Edges can be defined as locations in an image where there is a sudden variation in the gray level or color pixels [4]. Since edges consist of mainly high frequencies, we can, in theory, detect edges by applying a highpass frequency filter in the Fourier domain or by convolving the image with an appropriate kernel in the spatial domain. In practice, edge detection is performed in the spatial domain, because it is computationally less expensive and often yields better results. Kernels (Edge Detectors) used in spatial domain can easily extract edges in other words the boundaries of an object in an image. The strength of the edges is determined by the difference between the two values adjacent to the edge.

Edge detectors are based on the estimation of gray level gradient at a pixel. The gradient can be approximated in the x and y directions by

$$g_x(x, y) \approx f(x + 1, y) - f(x - 1, y), \quad (3.9)$$

$$g_y(x, y) \approx f(x, y + 1) - f(x, y - 1). \quad (3.10)$$

We can express gradient calculation as a pair of convolution operations,

$$g_x(x, y) = h_x * f(x, y), \quad (3.11)$$

$$g_y(x, y) = h_y * f(x, y), \quad (3.12)$$

where the kernels are

$$\mathbf{h}_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad \mathbf{h}_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \quad (3.13)$$

These are known as the *Prewitt kernels*. A similar pair of kernels is the *Sobel kernels*,

$$\mathbf{h}_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad \mathbf{h}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (3.14)$$

The two gradients computed at each pixel by Equations 3.11 and 3.12 can be regarded as the x and y components of a *gradient vector*,

$$\mathbf{g} = \begin{bmatrix} g_x \\ g_y \end{bmatrix}.$$

This vector is oriented along the direction of change, normal to the direction in which the edge runs. Gradient magnitude and direction are given by

$$g = \sqrt{g_x^2 + g_y^2}, \quad (3.15)$$

$$\theta = \tan^{-1} \left(\frac{g_y}{g_x} \right) \quad (3.16)$$

where θ is measured relative to the x axis.

Gradient magnitude will be large whenever g_x or g_y are large, for instance, whenever there is a big change in gray level within the 3 x 3 neighborhood of a pixel. Thus, g measures the strength of an edge, irrespective of its orientation.

3.3 Active Contour Model (Snake)

Active contour is a topic of image edge detection and are computer-generated curves that move within images to find object boundaries. They are often used in computer vision and image analysis to detect and locate objects, and to describe their shape. The active contour model, or snake, is an energy-minimizing spline guided by external constraint forces and influenced by image forces that pull it toward features such as lines and edges [6]. Snakes do not solve the entire problem of finding contours in images; rather, they depend on other mechanisms such as interaction with a user, interaction with some higher-level image understanding process, or information from image data adjacent in time or space [6]. The most important advantage of snake is that it is active and information from a higher-level process can be used. Changes in high-level interpretation can exert forces on a snake as it continues its energy minimization. The connectivity of the contours and the presence of corners affect the energy functional and hence the detailed structure of the locally optimal contour. In order to achieve good edge detection, applications, using this model let the user interact with the algorithm. In many applications, a user can give the initial position of the snake and experiment with snake algorithm's parameters.

The development of active contour models, or snakes, results from the work of Kass, Witkin, and Terzopoulos [6]. Energy function of the active contour is a weighted combination of internal and external forces. The internal forces emanate from the shape of the snake, while the external forces come from the image and/or from higher-level image understanding process.

Amini et al. [7] proposed an algorithm for the active contour model using dynamic programming. The approach is numerically stable. It also allows

inclusion of hard constraints in addition to the soft constraints inherent in the formulation of the functional.

Donna et al. [8] pointed out the problems of two approaches and proposed the *Greedy* algorithm. The algorithm is faster than Amini's $O(nm^3)$ algorithm, being $O(nm)$ for a contour having n points which are allowed to move to any point in a neighborhood of size m at each iteration. The function used in the algorithm is

$$E_{\text{snake}} = \int_1^0 (\alpha(s)E_{\text{cont}} + \beta(s)E_{\text{curv}} + \gamma E_{\text{image}}) ds \quad (3.17)$$

This energy functional is composed of three terms, a continuity term, a curvature term and an image term. The parameters $\alpha(s)$, $\beta(s)$ and γ are constants.

Computing E_{cont} : Given n points, $v_i = (x_i, y_i)$, $i = 1 \dots n$. Integer arithmetic is modulo n (the number of points on the contour) because the contour is closed. A neighborhood of size 3×3 , δ_i , of the current point v_i is shown in Figure 3.6.

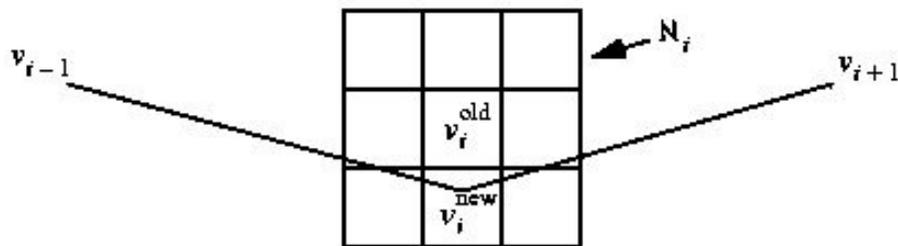


Figure 3.6 - 3×3 neighborhood that the search of the minimum energy occurs

Then,

$$\forall j \in \delta_i, E_{cont,j} = \bar{d} - |v_j - v_{i-1}| \quad (3.18)$$

where \bar{d} is the average distance between points, i.e.,

$$\bar{d} = \frac{\sum |v_i - v_{i-1}|}{n} \quad (3.19)$$

The continuity energy computed in Equation 3.19 is then normalized in the neighborhood δ_i as follows:

$$E_{cont,j} = \frac{E_{cont,j}}{\max_{j \in \delta_i} (E_{cont,j})} \quad (3.20)$$

Computing E_{curv} : The curvature energy is computed using the following formula:

$$\forall j \in \delta_i, E_{curv,j} = |v_{i+1} - 2v_j + v_{i-1}|^2 \quad (3.21)$$

This curvature energy is then normalized in the neighborhood δ_i as follows:

$$E_{curv,j} = \frac{E_{curv,j}}{\max_{j \in \delta_i} (E_{curv,j})} \quad (3.22)$$

Computing E_{image} : The image energy is computed from the image intensity gradient magnitudes, G_j , of all points in the neighborhood δ_i using the following algorithm:

$$\text{First, compute } G_{\max} = \max_{j \in \delta_i} (G_j) \text{ and } G_{\min} = \min_{j \in \delta_i} (G_j)$$

If $G_{\max} - G_{\min} < G_{thresh}$, then set $G_{\min} = G_{\max} - G_{thresh}$.

$$\forall j \in \delta_i, E_{image,j} = \frac{G_{\min} - G_j}{G_{\max} - G_{\min}} \quad (3.23)$$

Curvature: Curvature $c_i = \left[\frac{\Delta x_i}{\Delta s_i} - \frac{\Delta x_{i+1}}{\Delta s_{i+1}} \right]^2 + \left[\frac{\Delta y_i}{\Delta s_i} - \frac{\Delta y_{i+1}}{\Delta s_{i+1}} \right]^2$ with

backward differences $\Delta x_i = x_i - x_{i-1}$ and $\Delta y_i = y_i - y_{i-1}$ and arc length $\Delta s_i = \sqrt{\Delta x_i^2 + \Delta y_i^2}$.

Below is the Pseudocode for the snake algorithm of greedy approach.

Algorithm 3.2 – Pseudo code for snake (Greedy approach).

```

get initial contour // user specified
ptsMoved = n // number of points moved in the current iteration
while (ptsMoved > THRESHOLD3)
    // loop to move points on the current contour to new locations
    ptsMoved = 0
    for each point,  $v_i$ , on the contour
        for each point,  $v_j$ , in the neighborhood  $\delta_i$  of point  $v_i$ 
            compute its energy  $E_{\text{snake}} = \int_1^0 (\alpha(s)E_{\text{cont}} + \beta(s)E_{\text{curv}} + \gamma E_{\text{image}}) ds$ 
        end
        move point  $v_i$  to the point  $v_j$  in  $\delta_i$  with minimum energy.
        if  $v_i$  is moved to a new location, increment ptsMoved.
    end
    // loop to find corners where the curvature constraint is relaxed, i.e.,  $\beta$  is set to 0.
    for each point,  $v_i$ , on the contour
        compute curvature  $c_i$ 
        if (( $c_i > c_{i-1}$  and  $c_i > c_{i+1}$ ) //curvature is larger than neighbors
            and ( $c_i > \text{CURVATURE\_THRESHOLD}$ ) //curvature is larger than threshold
            and (gradient magnitude at  $v_i > \text{EDGE\_THRESHOLD}$ )) //strong edge

```

end *end* *then set $\beta_i = 0$ at point v_i*
end

CHAPTER 4

PREPROCESSING, SEGMENTATION AND FEATURE EXTRACTION OF HONEYBEE IMAGES

Preprocessing, segmentation and feature extraction are important steps for finding the features of the hindleg and forewing part of the honeybee images. In this chapter, application of these steps to the honeybee images, the results of each step and the difficulties encountered during the study will be discussed in detailed.

The sample images are captured by the TV card in the setup of Figure 2.1 and saved as 768x576, 24-bit RGB bitmap images. The hindleg and forewing of honeybees are taken in separate pictures for manipulating and extracting of the features from them. In the images the coordinates are represented as follows: Upper left corner = (0,0), Upper right corner = (0, IMAGE WIDTH), Lower left corner = (0, IMAGE HEIGHT) and Lower right corner = (IMAGE WIDTH, IMAGE HEIGHT).

4.1 Preprocessing of Honeybee Images

4.1.1 User Interaction

The proposed system for feature extraction of the honeybees requires interaction with the user. In the beginning of the program, the user is prompted to enter the critical points on the hindleg and forewing images. These reference points are used in different stages of the system for instance placing the honeybee images in the same position and orientation with the templates used in snake algorithm. These entered reference points speed up the system and make it more accurate. Moreover, in the forewing image the Forewing Length (F_L) and Forewing Width (F_W) parameters cannot be extracted with the image analysis since they are on the transparent part of the image and their gray level is same as the background gray level. Thus, entering the reference points is essential for this semi-automated system.

The reference points should be entered by the user respectively, which are shown in Figure 4.1 for forewing and in Figure 4.2 for hindleg images.

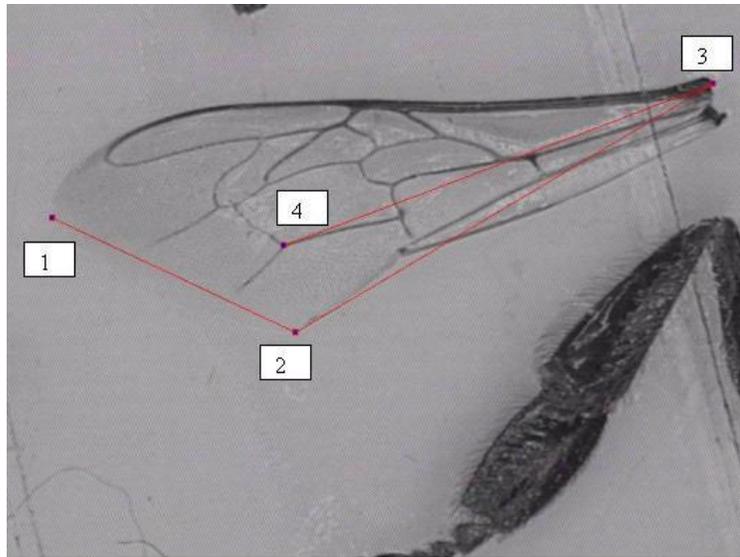


Figure 4.1 - The reference points of the forewing image entered by the user

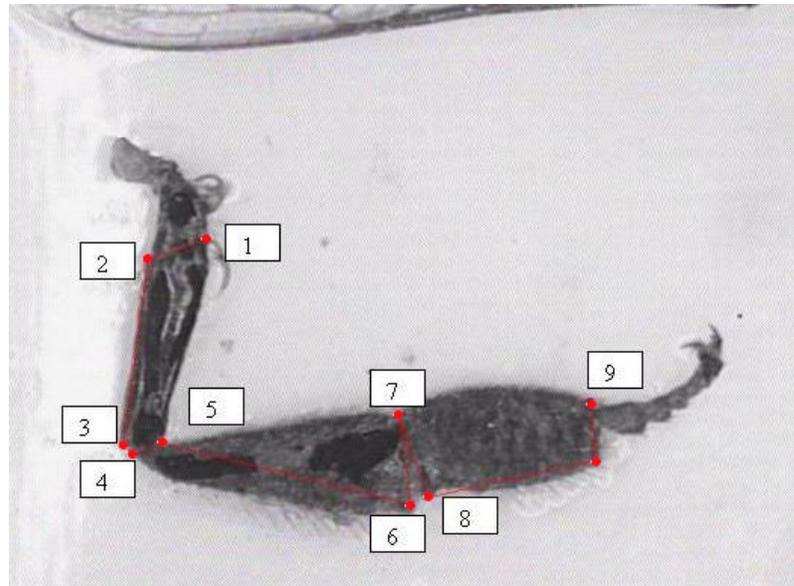


Figure 4.2 - The reference points of the hindleg image entered by the user

4.1.2 Orientation of the Honeybee Images with respect to the Template Images

The hindleg and forewing images should be put in the same orientation with respect to the template images before preprocessing of the images. In this process, the reference points entered (as explained in section 4.1.1) by the user on the hindleg and forewing images at the beginning of the application and the points on the template images that corresponds to the entered reference points are utilized. These points give the information about applying horizontal flip, vertical flip and

rotation to the honeybee images with respect to the template images. If the rotation angle is more than 5 degree, rotation can result with the loss of some part of the hindleg or forewing. To prevent this situation, the rotation must be applied to the image in a stepwise order with a small rotation angle as in the following:

1. Calculate the image rotation angle.
2. If the absolute rotation angle is less or equal to 5 degree, apply rotation using this value only once and go to step 7 else go to step 3.
3. Apply 5 degree rotation to the image.
4. Circularly shift the image to the side (the right or left) where the gap between the object of interest and the image border is greater than the gap of the other side. The shift value is determined as the (gap value – 5).
5. New value of image rotation angle is (image rotation angle – 5).
6. Go to step 2.
7. End of the algorithm.

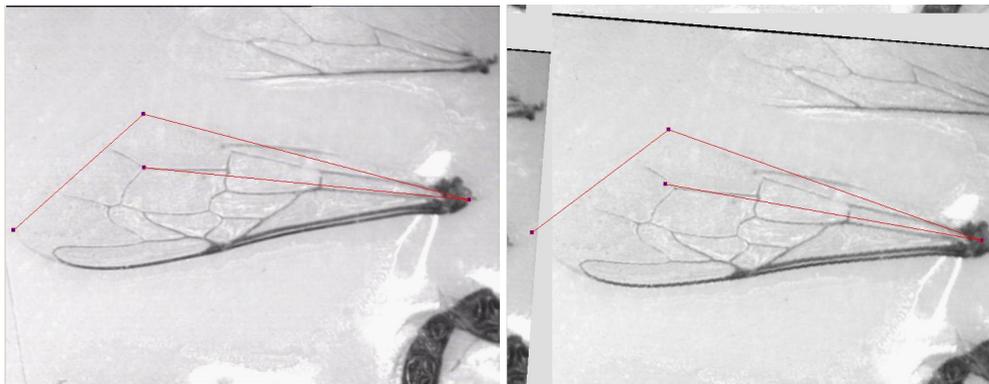
During these processes, the reference points entered by the user are also updated. The hindleg and forewing images with the reference points entered is shown in Figure 4.3 and images after applying horizontal, vertical flip and rotation is shown in Figure 4.4 and Figure 4.5.



(a)

(b)

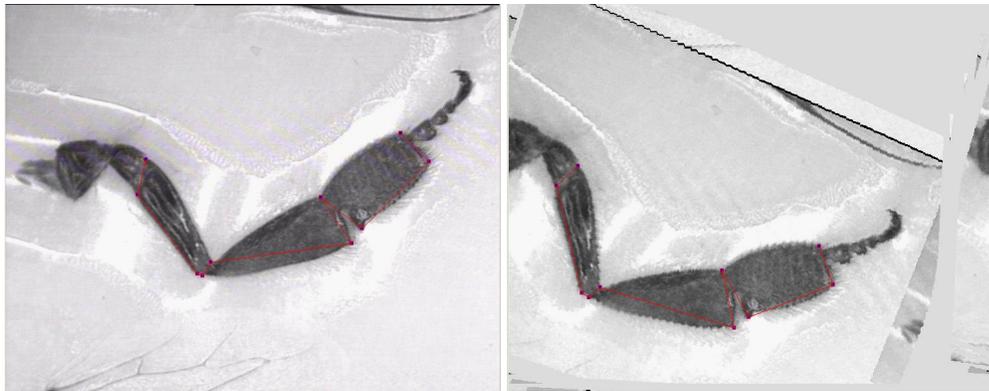
Figure 4.3 - (a) Forewing (b) Hindleg images with the reference points entered before the orientation



(a)

(b)

Figure 4.4 - Forewing image (a) Horizontal and vertical flip and (b) Rotation is applied



(a)

(b)

Figure 4.5 - Hindleg image (a) Horizontal and vertical flip and (b) Rotation is applied

The orientation of the images is a very important step because this process gives the initial position to the hindleg and forewing templates that is used in the snake algorithm. As it will be explained in the following sections, the initial position of the templates affects the whole performance of the snake algorithm. Moreover, the speed of whole system is increased by a great amount by this process.

4.1.3 RGB to Gray Scale Conversion

Although the images are taken as 24-bit true color images, they are converted to gray scale in the beginning of the process. The rest of the process works on the gray scale images since there is no need to extract a parameter related to the color information and working on a true color images requires three times more

memory than working on an 8-bit gray scale image. Figure 4.6 shows samples of 768x576, gray scale, bitmap hindleg and forewing images.



(a)

(b)

Figure 4.6 - 768x576, 24-bit gray scale bitmap honeybee images (a) Hindleg (b) Forewing of a honeybee

4.1.4 Reducing Noise

Prior to manipulating and extracting information, we must improve the quality of the hindleg and forewing images. The noise produced by the devices used in grabbing the images is reduced by applying low pass filtering. By using the convolution kernel in (4.1), low-pass spatial filtering is applied to the images.

$$\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} / 16 \quad (4.1)$$

This method reduces the finely structured erroneous intensity fluctuations seen in the histogram of the images. The histogram of hindleg image in Figure 4.6 (a) is shown in Figure 4.7 (a), which shows these noise peaks clearly. After applying low pass filter to the hindleg image, the noise peaks are removed in the histogram shown in Figure 4.7 (b).

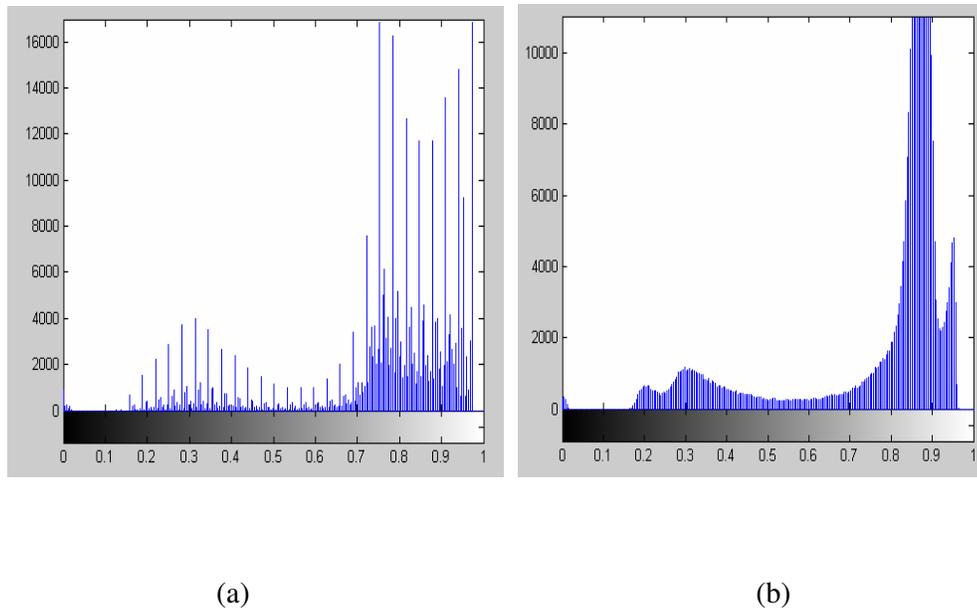
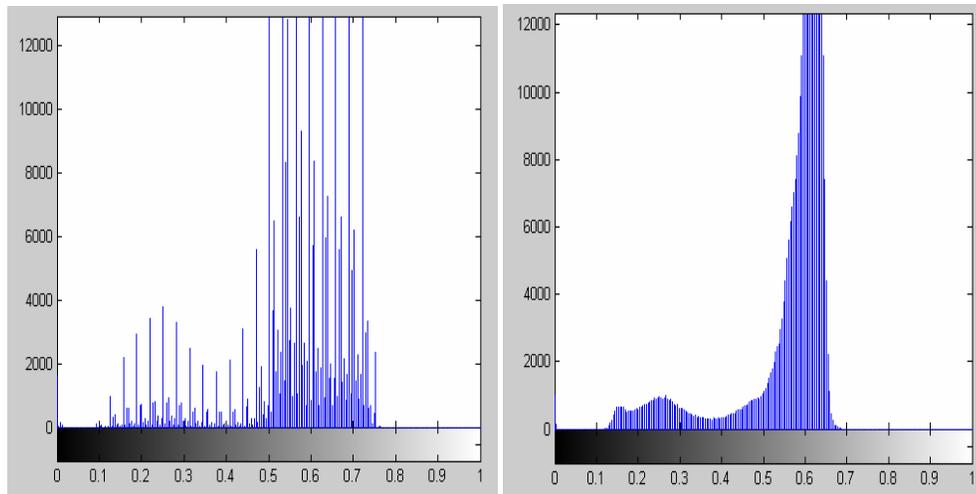


Figure 4.7 - Histogram of hindleg image a) before and (b) after applying low pass filter

The histogram of forewing image in Figure 4.6 (b) is shown in Figure 4.8(a), which shows noise fluctuations as in hindleg histogram. After applying low pass filter to the forewing image, the noise peaks are removed in the histogram shown in Figure 4.8 (b).

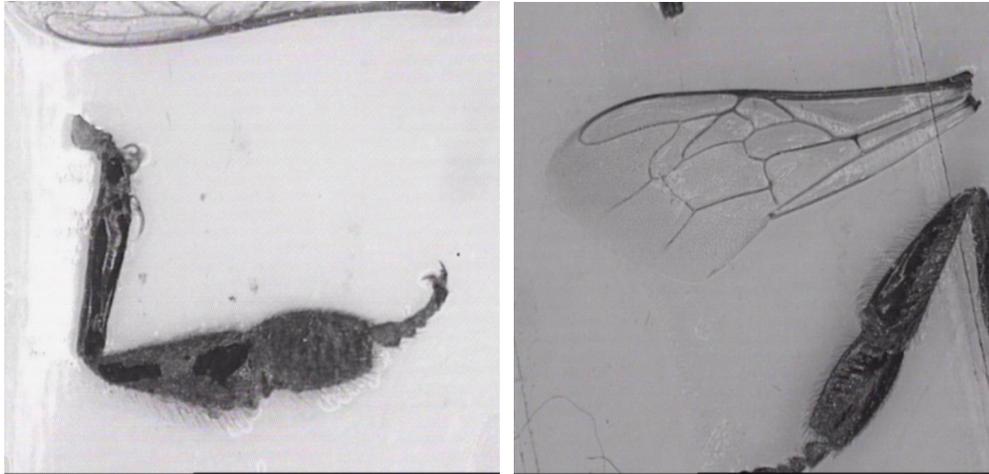


(a)

(b)

Figure 4.8 - Histogram of forewing image a) before and (b) after applying low pass filter

The resulting images after reducing the noise are shown in Figure 4.9. As low pass filtering replaces each pixel with a weighted sum of each pixel's neighborhood, there is a side effect of smoothing the images.



(a)

(b)

Figure 4.9 - The resulting (a) hindleg (b) forewing image after applying low pass filtering

4.1.5 Binarization

For the sample images, we have dark objects namely as hindleg and forewing on a light background. Binarization or thresholding the images with an appropriate value provides us to segment the hindleg and forewing part which are our “objects of interest”. The thresholded images are subjected to further analysis, which affects the whole performance of the system. Thus, segmentation represents an important early stage in image analysis and this should be done well.

If we examine the histogram of the honeybee images after low pass filtering (Figure 4.7 (b) and Figure 4.8 (b)), it is seen that the histogram consists of one big peak and one or more small peaks around the big one. This big peak corresponds to the light background and the smaller peaks on the left correspond to hindleg or forewing part of the image. As dissected forewings and hindlegs of the honeybees are fixed on microscopic slides with transparent tape and the images are taken from that slides, the part of the images that of the transparent tape exists is the brightest part of the whole image. One of these brightest parts can be seen in the left part of the hindleg image in Figure 4.9 (a). In the histogram of that image in Figure 4.7 (b), the peak on the right of big peak (i.e. background) represents this brightest part (i.e. transparent tape). As our aim is to extract the hindleg and wing part of the image, to take the left part of the big peak provides us to reach our goal. However, this is not simple as this. In our early experiments, we search the histogram of the images and decide that some percentage of the maximum peak value from the left in the histogram is our threshold value to separate the objects of interest. This early conclusion was stated in [9]. Especially for the wing images, applying binarization with one threshold to the whole image is not a solution, because wide part of the wing image is more transparent than the other parts of it. If one threshold is applied to the forewing image the transparent parts are not segmented sufficiently or if we increase the threshold value for the forewing image, the transparent parts are segmented but in this case the unwanted parts near the darker parts are segmented unnecessarily. This wrong thresholding affects the whole algorithm in finding the angle and length features of honeybees. To handle this problem, using also the reference points entered by the user, the wing image is separated into 3 critical regions in thresholding algorithm. These regions for the wing image have different thresholds calculated with the following algorithm:

1. Reload image in RGB format
2. Convert the image to gray scale
3. Apply low pass filtering to image
4. Find the histogram of the image
5. Find pixel intensity that has the maximum number of pixels in the histogram. This is the max value of the background pixel intensities ($G_{\text{MaxBackground}}$).
6. Find the number of pixels at intensity $G_{\text{MaxBackground}}$ in the histogram. Take the 1/4 of that number of pixels as $N_{1/4\text{Percent}}$ and from the $G_{\text{MaxBackground}}$ to 0, find the first pixel intensity that has number of pixels less than $N_{1/4\text{Percent}}$. This value is the initial empirical threshold (T_{initial}) for the wing image ($T_{\text{threshold}} = T_{\text{initial}}$).
7. Apply wing blob analysis to the image that will be explained in the further sections. Wing blob analysis simply places the forewing image in the same position and orientation with the forewing template, update the reference points accordingly, apply binarization using $T_{\text{threshold}}$, select wing blob (among the wing blob and unwanted blobs), apply edge detection and thinning algorithms.
8. Divide wing image into enough number of regions. For our application, 3 regions are appropriate and experimentally we determine the empirical white pixel percentage ($EP_{\text{WhitePixel}}$) for every region that is enough to form the wing edges in that region.
9. Calculate the white pixel (i.e. pixels with gray level 255) percentage (P_{WhitePix}) in the region. If this percentage is less than empirical white pixel percentage ($EP_{\text{WhitePixel}}$) than increase $T_{\text{threshold}}$ by 1, else if this percentage

is more than empirical white pixel percentage ($EP_{\text{WhitePixel}}$) than decrease $T_{\text{threshold}}$ by 1.

10. Repeat steps 1 through 12 until absolute value of $(P_{\text{WhitePix}} - EP_{\text{WhitePixel}})$ is less enough. Take the $T_{\text{threshold}}$ value as the threshold of that region ($T_{\text{ThresRegion}}$). This empirical delta value is also determined experimentally for every region.

11. Repeat steps 1 through 10 to find $T_{\text{ThresRegion}}$ value for every region.

The regions are selected with the help of the reference points of forewing image entered at the beginning of the application. The first and second region is selected using the midpoint between reference points 1 and 3. The first region is on the right of this midpoint in a specified region and the second region is on the left of this midpoint in a specified region. The third region is selected in the neighborhood of reference point 4, since the wing image in this region is almost transparent and this region should be segmented with a different threshold. The regions of forewing image are shown in Figure 4.10.

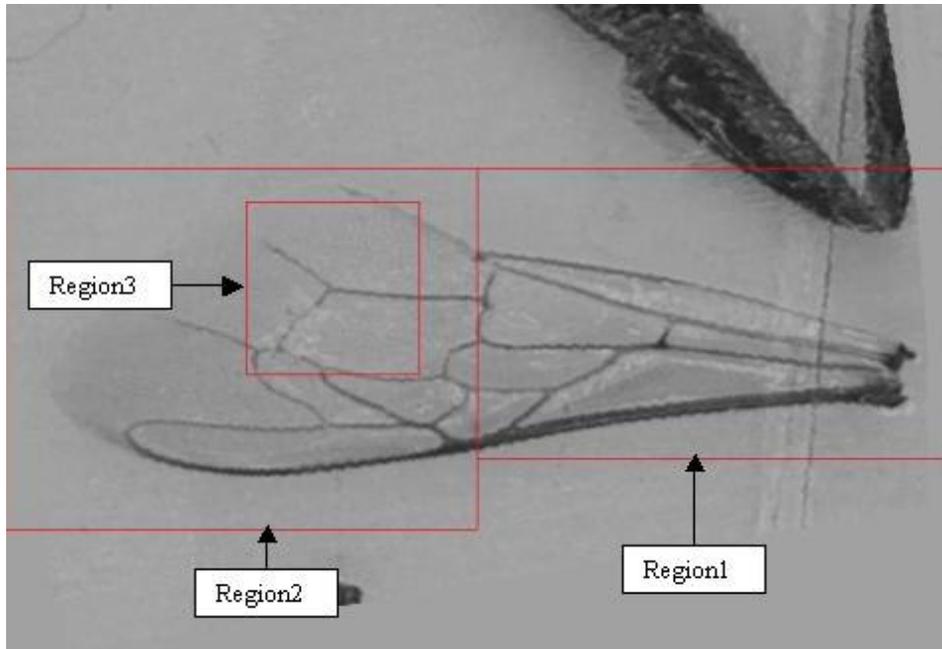


Figure 4.10 - Regions of forewing image

There is no need to separate the hindleg image into regions for determining the threshold values as in wing image, since the hindleg images have no transparent parts in it and oppositely they have gray levels easily separable in the histogram. Experimentally separating the smaller peak (i.e. hindleg) on the left of the biggest peak (i.e. background), as can be seen from the hindleg histogram in Figure 4.7 (b), results successfully.

The algorithm for the hindleg image is explained as follows:

1. Sum up the number of pixels in each gray level starting from gray level 0 through 255 using the histogram.
2. Stop summing if the number of pixels reached is 14 percent of total number of pixels of the hindleg image that is 768x576. The last pixel intensity reached in the histogram is the maximum pixel intensity limit (P_{Limit}) that will be used in searching the threshold value.
3. Find the pixel intensity that has the maximum pixel number between the pixel intensity 0 through P_{limit} in the histogram. This value is the mean pixel intensity ($P_{MeanHindleg}$) of the hindleg part.
4. Find the pixel intensity that has the minimum number of pixels between $P_{MeanHindleg}$ and P_{Limit} in the histogram. The found pixel intensity is the appropriate threshold ($P_{Threshold}$) for the hindleg image.

For the above algorithm, the parameters used in finding the threshold value for the hindleg image (Figure 4.9 (a)) are shown in Figure 4.11.

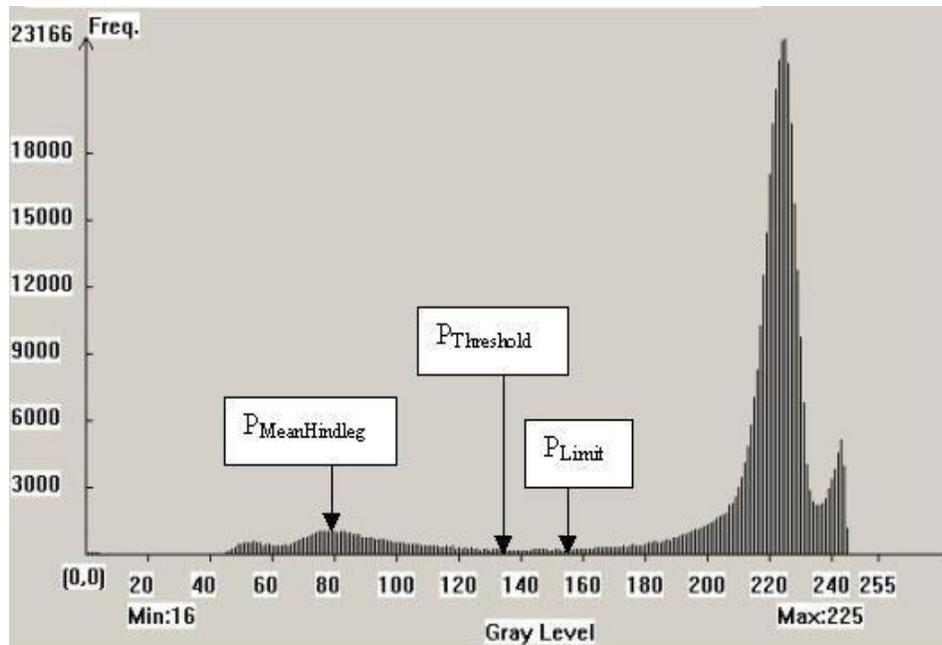


Figure 4.11 - Histogram parameters used in finding the threshold of the hindleg

Applying these algorithms to the hindleg and forewing images, the automatic thresholding is achieved. The resulting images are in Figure 4.12.



(a) (b)
Figure 4.12 - (a) Hindleg and (b) Forewing images after binarization

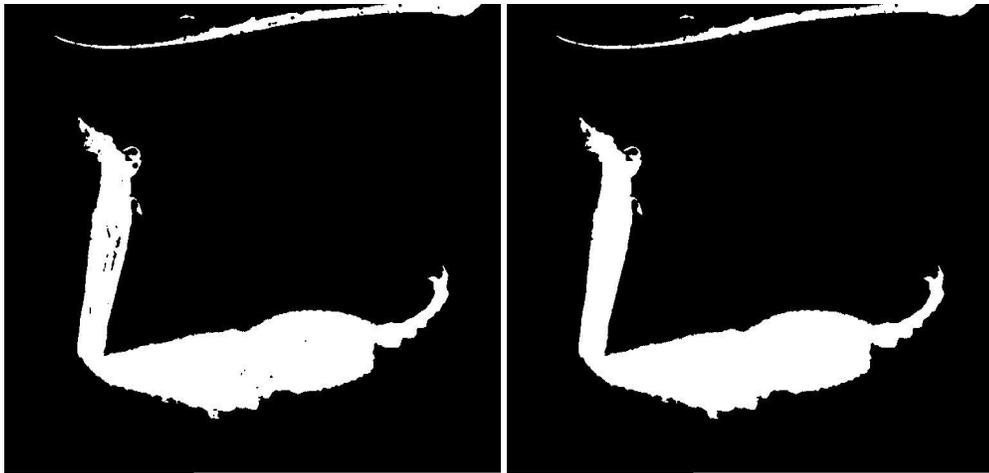
4.2 Blob Analysis

Blobs are connected regions of pixels within an image. After binarization of the images, many unwanted blobs are present in the images. Some part of the forewing image in the hindleg image and some part of the hindleg image in the forewing image can be segmented with thresholding. There are frequently small particles around the forewing and hindleg part of the image and small blobs inside the wing part of the image. Transparent tapes of slides as in the right side of wing image in Figure 4.12 (b) can also be observed on the image. These are all unnecessary connected pixels that should be eliminated by the blob analysis.

4.2.1 Selection of Hindleg Blob

The unnecessary blobs are eliminated and only the hindleg part of the image is selected in the following algorithm.

1. Scan the whole image by using eight-neighborhood adjacency, identify and label each blob (Labeling).
2. Calculate the area of each blob.
3. Eliminate each blob whose area is less than 50 pixels by assigning to these blob pixels the background color "0" value. This removes the small particles in the hindleg image as can be seen in Figure 4.13 (a).
4. Fill holes in the remaining blobs by assigning the blob value "255" to the whole pixels. In this way the holes on the hindleg part of the image is filled with the object pixel color that is shown in Figure 4.13 (b).
5. Scan the whole image by using eight-neighborhood adjacency, identify and label each blob again (Labeling).
6. Calculate the area of each blob.
7. Some part of the wing blob is partially seen in the hindleg image (Figure 4.13 (b)). As the hindleg is very dark with respect to the wing, the blob with the maximum area is the hindleg blob. So, find the blob with maximum number of pixels (i.e. area).
8. Eliminate each blob other than the blob with maximum area by giving these blobs the background gray level.
9. There is only the hindleg blob in the image as a result. The resulting image is shown in Figure 4.14.



(a)

(b)

Figure 4.13 - Binary hindleg image after (a) Small particles removed (b) Fill holes



Figure 4.14 - Selected hindleg blob after applying blob analysis

4.2.2 Selection of Forewing Blob

In the following algorithm the removing of the unnecessary blobs and selecting only the forewing part of the image is explained.

1. Scan the whole image by using eight-neighborhood adjacency, identify and label each blob (Labeling).
2. Calculate the area of each blob.
3. Exclude each blob whose area is less than 50 pixels by assigning to these blob pixels the background color "0" value. This provides removing the small particles in the forewing image.

The wing part has lighter gray level pixels than the hindleg part even some part of the wing edges is transparent. After binarization because of the braking off some veins in the wing part, the wing blob breaks into main blob and one or more small blobs. For that reason, it is not possible to propose an area-based method (i.e. the blob with maximum area) as in the hindleg image before eliminating the unwanted blobs with the area bigger than the wing blob. For instance, in Figure 4.12 (b), the most of the hindleg blob can be seen in the wing image and the wrong blob (i.e. hindleg) will be chosen if the blob with the maximum area is selected in this image.

4. Calculate the center of gravity (COG) of each blob. After the blobs are excluded with areas smaller than 50 and the center of gravity of the remaining blobs are found and marked with "+" sign, the resulting image is seen in Figure 4.15.

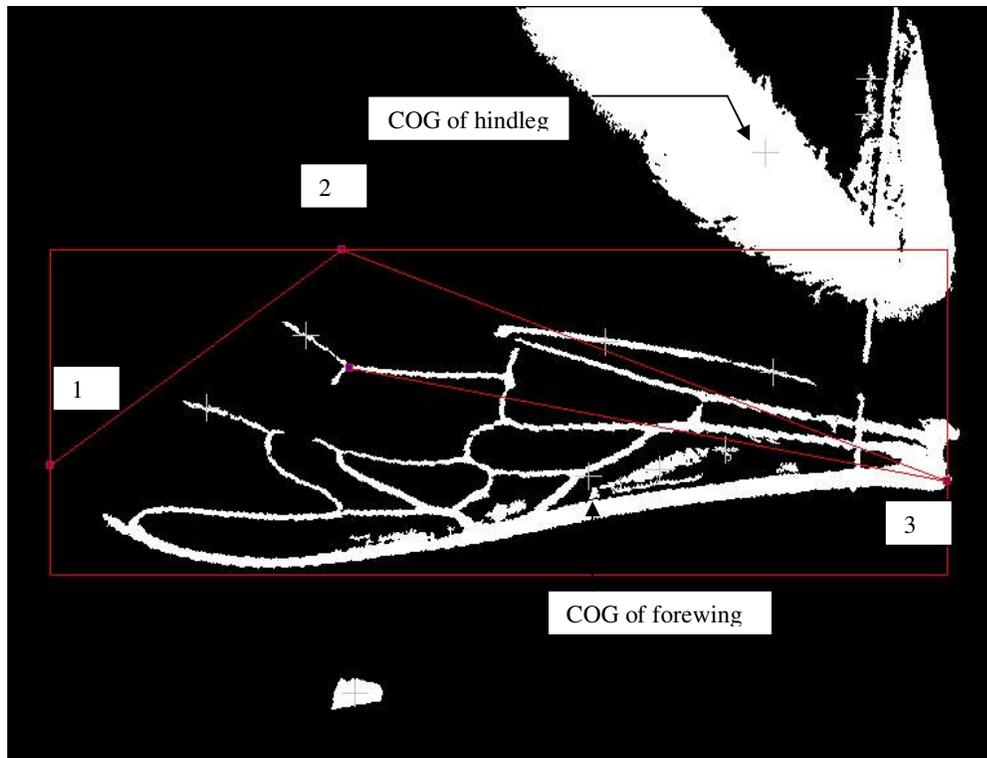


Figure 4.15 - Hindleg image after small blobs removed, all center of gravities and region defined by reference points

5. The first 3 reference points that are entered by the user in the beginning of the application are used in this step for the selection of the wing image. Determine a rectangular region that includes the wing part of the image. The vertices of this region starting from the top left corner and continue in the clockwise are: (x of Ref 1, y of Ref 2), (x of Ref 3, y of Ref 2), (x of Ref 3, y of Ref 1 + 83), (x of Ref 1, y of Ref 1 + 83). The reference points

used in this step and the region defined by these points are shown in Figure 4.15.

6. Use the wing region defined in step 5 and exclude the blobs whose center of gravity coordinates are out of that region by assigning pixels of the eliminated blobs the background color, which is "0". By this way, the blobs out of the wing region (the most important one is hindleg blob) are removed from the wing image.
7. Calculate the areas of the remaining blobs. Choose the blob with the maximum area as the wing blob.
8. So far, we have eliminated all small particles whose area is less than 50 and all the blobs out of the wing region, and found the wing blob. However, there are still blobs inside the wing region that should be eliminated. They cannot be eliminated until this point because the gray level of that blobs are low enough to be segmented with the wing blob and their areas are greater than 50. These blobs are between the center of gravity of the wing and the third reference point, which are shown in Figure 4.15. To eliminate these unwanted blobs, find the center of gravity of the wing blob. Exclude the blobs between the center of gravity of the wing and the third reference point by assigning pixels of the eliminated blobs the background color, which is "0". The vertices of this region starting from the top left corner and continue in the clockwise are defined as: (x of COG wing, y of Ref2 - 50), (x of Ref 2, y of Ref2 - 50), (x of Ref 2, IMAGE_HEIGHT), (x of COG wing, y IMAGE_HEIGHT), where IMAGE_HEIGHT is 768.
9. After applying all these steps, only the forewing blob is obtained in the image. The resulting image is shown in Figure 4.16.

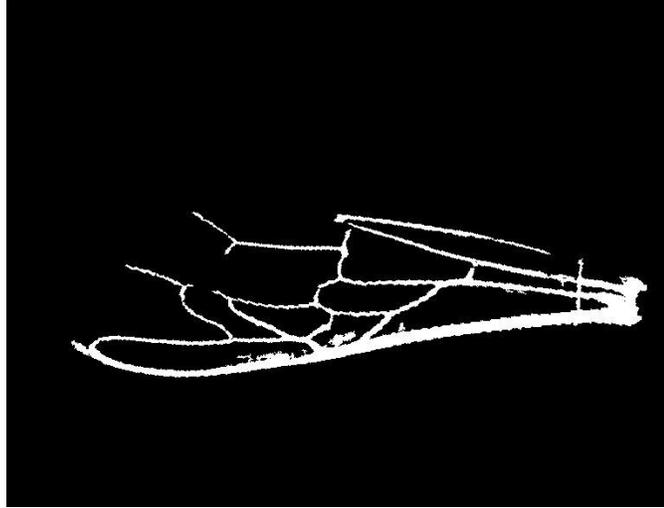


Figure 4.16 - Selected forewing blob after applying blob analysis

4.3 Edge Detection of Honeybee Hindleg and Forewing Blobs

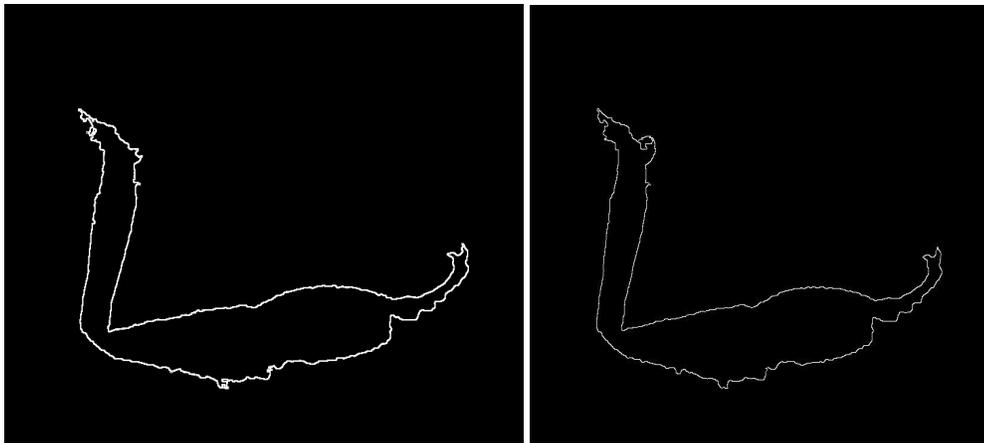
Binarization of the images with appropriate threshold values and selecting only hindleg and forewing blobs among the unnecessary blobs are explained so far. In this section last phase of segmentation, namely edge detection will be discussed.

4.3.1 Edge Detection of Hindleg Blob

The features that should be found for hindleg of honeybees are on the boundary of that object. So, edge detection algorithm is applied to extract the edges of hindleg blob. The convolution kernel in (4.2) is applied to the binary image in Figure 4.14.

$$\left[\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} + \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & -2 \\ -1 & 0 & 1 \end{bmatrix} \right] / 2 \quad (4.2)$$

The first term in this kernel represents the vertical edge and the second term represents horizontal edge detection. The resulting image is shown in Figure 4.17 (a). Thinning with skeletonization is applied after that to reduce the edges of the hindleg to its skeleton in other words to 1 pixel width. The resulting image is in Figure 4.17 (b).



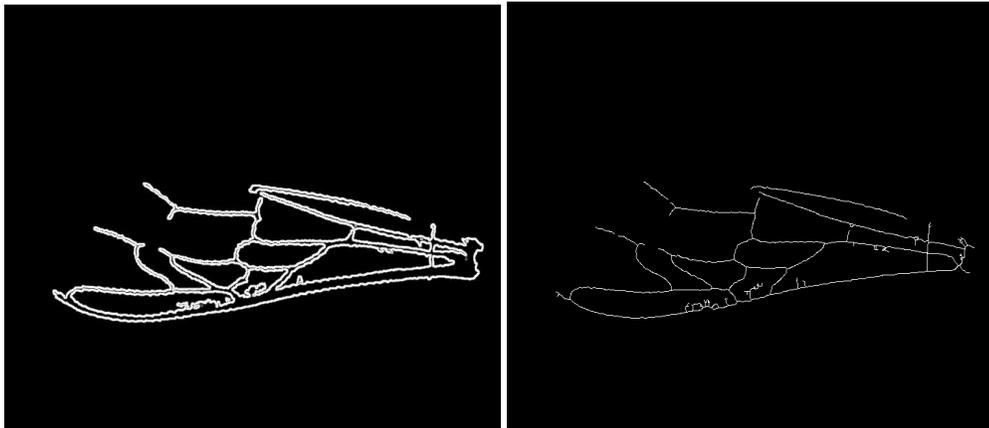
(a)

(b)

Figure 4.17 - Hindleg (a) edge detection (b) Skeletonization

4.3.2 Edge Detection of Forewing Blob

The kernel in (4.2) is not applied to the wing image (Figure 4.16). If this kernel is applied to the binary wing blob, two edges are formed because of the thick veins in wings which are not required. This undesirable result is in Figure 4.18 (a). Instead of using this kernel, the edges of the wing blob are extracted by applying simply the skeletonization which forms the edges in 1 pixel width. The desirable resulting image is in Figure 4.18 (b).



(a)

(b)

Figure 4.18 - (a) Undesirable wing edges which consist of double edge (b)

Desirable wing edges which consist of single edge

The edges of the forewing and hindleg are extracted by these algorithms. The images are now ready for the feature extraction, which will be discussed in the next section.

4.4 The Features Extraction of Honeybee Hindleg and Forewing

The features of the honeybees to be extracted were discussed in Chapter 2. Remember that there are four length features belonging to the hindleg as can be seen from the Figure 2.1 and the name of features are given in Table 2.3. Eight length features belonging to the forewing are shown in Figure 2.2 as the name of features are given in Table 2.4. Eight angle features belonging to the forewing are shown in Figure 2.3 and the features' names are given in Table 2.5.

All angle measurements and all length measurements except Forewing Length (FL), Forewing Width (FW) and Metatarsus Width (MW) are taken by utilizing the intersection points of the edges.

The Active Contour is used to find the boundaries of the objects by moving within images. It is an energy-minimizing spline also depends on interaction with a user. We use the Active Contour or Snake algorithm for these suitable properties.

The steps in using this algorithm are explained in the following sections.

4.4.1 Honeybee Forewing Feature Extraction

A template that represents the wing veins (i.e. edges) should be formed to use it in the Active Contour algorithm as energy-minimizing splines. The wing template that we use consists of 6 active contours which are created by taking sample points on the edges of a high quality sample wing image. The edges in each contour are formed by the active contour curves through which these sample points and additional points pass. These sample points and additional points are

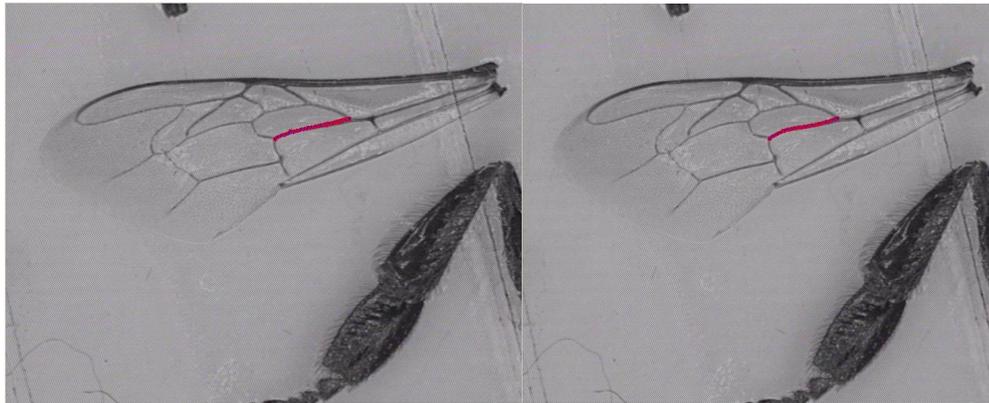
formed by one of the proper techniques, namely Bezier Curves or the Polynomial Interpolation [10] [11]. The sample points are given to one of these appropriate curve description methods and the required number of new points is generated as an output. A Bezier Curve is associated with the vertices of a polygon which uniquely define the curve shape. Only the first and last vertices of the polygon actually lie on the curve, however, the other vertices define the shape of the curve. In Polynomial Interpolation method, the curves pass through all the specified data points; i.e. it is a curve fitting technique. If 6 sample points are taken on the forewing vein and 50 points are required to be generated on the vein with each method, the shape of the curves are resulted as in Figure 4.19. For the wing edge that is shown in Figure 4.19, the Bezier curve method should be used for a fine result. The both algorithms result well in the straight curves as in Figure 4.20, for that edge any one of them can be used to form the active contour.



(a)

(b)

Figure 4.19 - Forewing edge formed with (a) Bezier Curve (b) Polynomial Interpolation



(a)

(b)

Figure 4.20 Straight edge formed with (a) Bezier Curve (b) Polynomial Interpolation

The wing template formed by Bezier Curves and Polynomial Interpolation methods is shown in Figure 4.21. This template is exploited in applying the active contour algorithm to each of 6 contours in the wing image. The six contours are labeled as 1 through 6 and the corners of the snakes (i.e. intersection points of the edges) are represented with the letters from A to O as in Figure 4.21

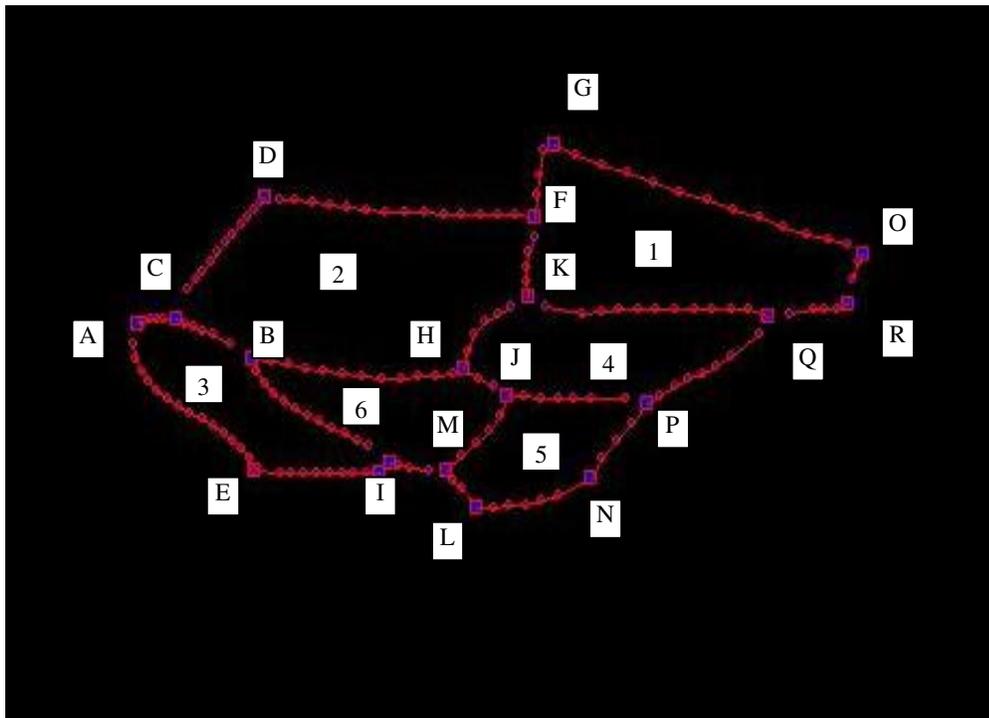


Figure 4.21 - Forewing template

The selection of Bezier Curves or Polynomial Interpolation methods used to form forewing template edges are given in Table 4.1.

Table 4.1 - Method of formation of the forewing template edges

Forewing template edges	Bezier Curve	Polynomial Interpolation
AC		✓
AE	✓	
CB		✓
CD	✓	
EI		✓
BI	✓	
BH	✓	
DF	✓	
IM		✓
ML		✓
MJ	✓	
HJ	✓	
HK	✓	
FK	✓	
FG		✓
LN		✓
NP	✓	
JP	✓	
KQ	✓	
PQ		✓
GO	✓	
QR		✓
RO	✓	

Some edges are common for the 6 active contours that make up the wing template. For instance, GO edge in contour 1 is an uncommon edge, but FK edge is a common edge for contour 1 and contour 2. If snake algorithm is applied to all contours from 1 to 6 only once, then snake is applied to common edges more than once (i.e. twice for FK edge).

In active contour algorithm, a point on the active contour of the wing template moves to a point with the minimum energy in a specified neighborhood. The energy of that point is calculated with the function below which is explained in detail in Chapter 3:

$$E_{\text{snake}} = \int_1^0 (\alpha(s)E_{\text{cont}} + \beta(s)E_{\text{curv}} + \gamma E_{\text{image}}) ds \quad (4.3)$$

The giving initial position to the active contour template and tuning the active contour parameters are very important steps that affect the whole performance of the algorithm. In many applications, the user can give the initial position of the snake and experiment with snake algorithm's parameters. In our system, the initial position of the wing template is determined by the first 4 reference points that the user enters at the beginning of the application. Three parameters affect the performance of the snake that are $\alpha(s)$ which controls the continuity between points, $\beta(s)$ represents the curvature of the snake, γ control the energy of the image itself.

The most important part about understanding and implementing this algorithm is to get an intuitive sense about how the energy terms work together as well as independently. In the active contour model, all three terms coexist and respond to each other, causing the snake to converge to its contour.

The first term of the energy equation maintains equidistance between points in the snake. It is formulated by taking the difference between the average segment length and the current snake segment length. The minimum energy for continuity results when the current segment length is closest to the average.

The second term of the energy equation represents the curvature of the snake. It keeps contour smooth except at corners. From an intuitive standpoint, the curvature term should try to make the snake form a convex circular shape when there are no other forces applied. Unfortunately though, the introduction of continuity causes the impetus for circularity to result in rapid shrinking. This is both good and bad. It's good because in most cases, the snake needs to converge on a shape by shrinking. However, in the cases where a snake needs to expand, there needs to be significant local image energy to pull on it.

The third and most important term is the energy due to the image itself. This energy results from the magnitude of the gradient. When this term is thought of in a negative sense, the image energy is minimal when the intensity differential is greatest. This means that the snakes have a great tendency to move toward the edges.

Well-balanced coefficients are the key points in getting the proper response from the snake. For that reason, the optimal coefficients $\alpha(s)$, $\beta(s)$ and γ for each active contour of the wing template are found by testing the parameters experimentally. The selection of snake neighborhood is also an important step, because a point in the snake cannot move on to an edge point if the edge point is not in the neighborhood of the snake point. The experimentally found coefficients $\alpha(s)$, $\beta(s)$, γ and the neighborhood (δ) of snake for each contour is given in Table 4.2. The snake convergence is fast in our implementation as the

snake is applied to the binary image in Figure 4.18 (b) instead of a noisy gray scale image.

Table 4.2 - Coefficients and neighborhood of snake for each contour.

Contour	$\alpha(s)$	$\beta(s)$	γ	δ
1	6	6	8	9x9
2	3	3	5	9x9
3	4	6	8	9x9
4	6	8	8	11x11
5	6	8	8	9x9
6	6	8	8	11x11

The steps of applying the active contour algorithm to the wing image are given below:

1. Form the edges of the wing template by one of the proper curve description methods (Table 4.1); Bezier Curve or Polynomial Interpolation. This template is stored and used for every forewing.
2. Give the initial position to forewing template by shifting the template from the D point to the reference point 4, entered by the user. Point D of the template should coincide with the reference point 4.

3. Add the points in the active contour n to the snake points list of that contour. (i.e. if n is 1, the active contour is composed of the edges GO, OR, RQ, QK, KF and FG. The points on these edges are added to the snake points list respectively.)
4. Set $\alpha(s)$, $\beta(s)$, γ and the neighborhood parameters (δ) of the snake that are found experimentally before (Table 4.2).
5. Apply the snake algorithm using the contour n (i.e. n^{th} contour of wing template) to the binary image (Figure 4.18 (b)). While applying the algorithm, do not change the snake points that correspond to the reference points entered by the user (i.e. Template point D which corresponds to reference point 4).
6. Apply the snake algorithm only once on the contour number n .
7. Apply the snake algorithm to the other contours in the wing template.

The implementation of the active contour results as in Figure 4.22.

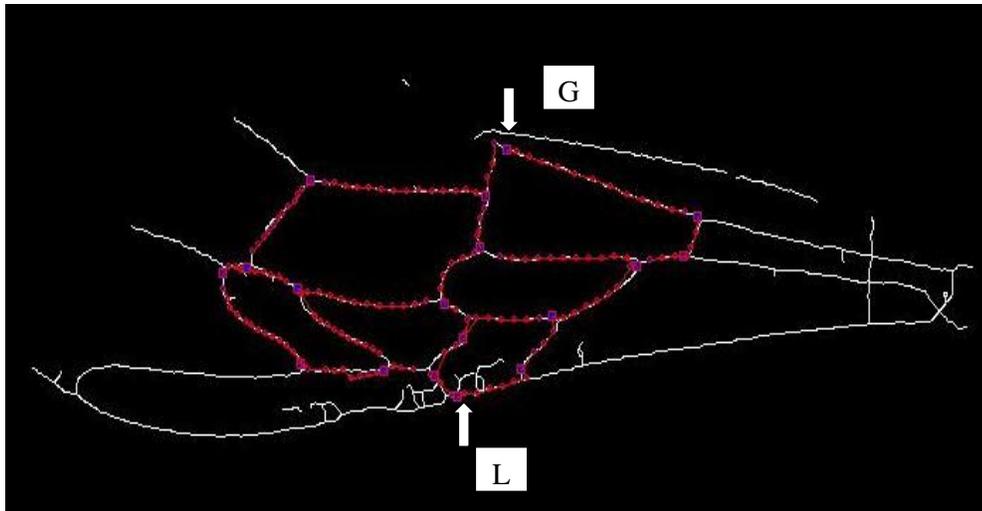


Figure 4.22 - The result of the active contour algorithm applied to the wing image

With applying the snake algorithm, the forewing template contours converge on the edges of the forewing. However, some critical points (i.e. corners of the active contours) used in the length and angle measurements can move to the wrong places on the forewing image. In order to correct these wing template critical points, an algorithm that is finding the vertex points in a specified neighborhood of these critical points, is applied to the forewing images. The critical points of wing template are then moved to the found vertex points. A vertex point is formed by the intersection of 3 edges of the forewing. The example of two vertex points is shown in Figure 4.23 (a) and (b). There is an example of a nonvertex point in Figure 4.23 (c). If 3 edges in Figure 4.23 (c) were intersected at a point, a vertex

point would be formed and it would be found by the vertex point finding algorithm. In fact, a vertex point at G in the foregoing images is seldomly formed, because the 3 veins that form the 3 edges are also seldomly intersected at a point in the foregoing.

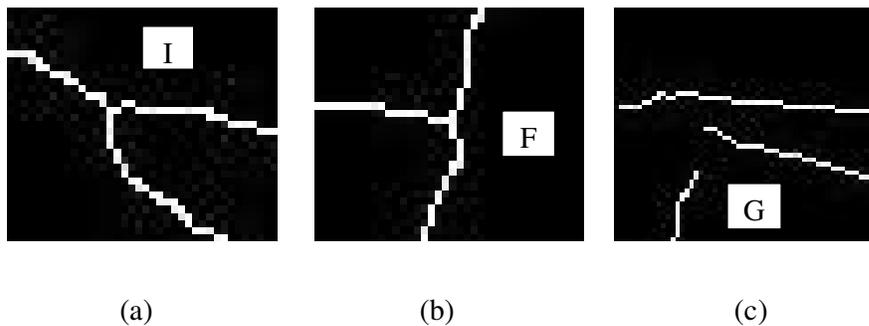


Figure 4.23 - Example of two vertex points I (a) and (b) F and a nonvertex point G (c) from the edged forewing image

The algorithm of finding vertex points is given in Appendix C.

Some parts of the edges may not be formed or some undesired extra edges may be formed because of the segmentation. Due to these factors, a critical point of the wing template is not moved to any point in case the vertex point is not found. In case an undesired vertex point is found, the critical point is moved to this wrong vertex point. For that reason, the selection of the neighborhood of the critical point

that the search of the vertex point is handled should be selected properly. A final correction of the critical points of the wing template that are not on the correct vertex points can be done by the facility provided by the system. One of these inevitable wrong points is critical point G mentioned as a nonvertex point and critical point L mentioned as undesired vertex point because of the extra edges (Figure 4.22). The user can move a critical point in the wing template to the correct vertex point by dragging and dropping that point. The resulting forewing image after applying “vertex point finding algorithm” and correcting G and L points is shown in Figure 4.24.

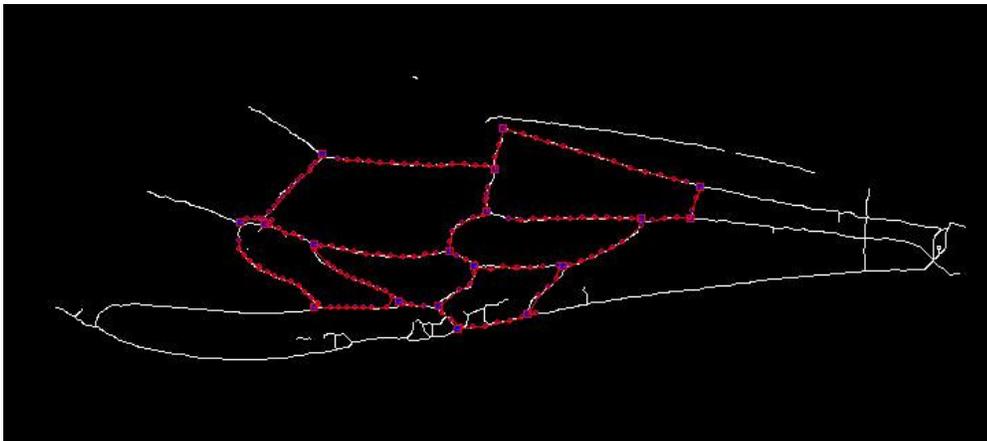


Figure 4.24 - The result of vertex point finding algorithm and correction of critical points by user

After applying the snake algorithm to whole image, the corners of the active contours will help us to find the length and angle measurements of the wing. The length between two points, say A and B, is calculated using the equation below:

$$Length_{AB} = \sqrt{(A_x - B_x)^2 + (A_y - B_y)^2} \quad (4.4)$$

The angles between the vectors \vec{u} and \vec{v} are calculated as the equation below:

$$\theta = \cos^{-1} \left(\frac{\vec{u} \bullet \vec{v}}{|\vec{u}| |\vec{v}|} \right) \quad (4.5)$$

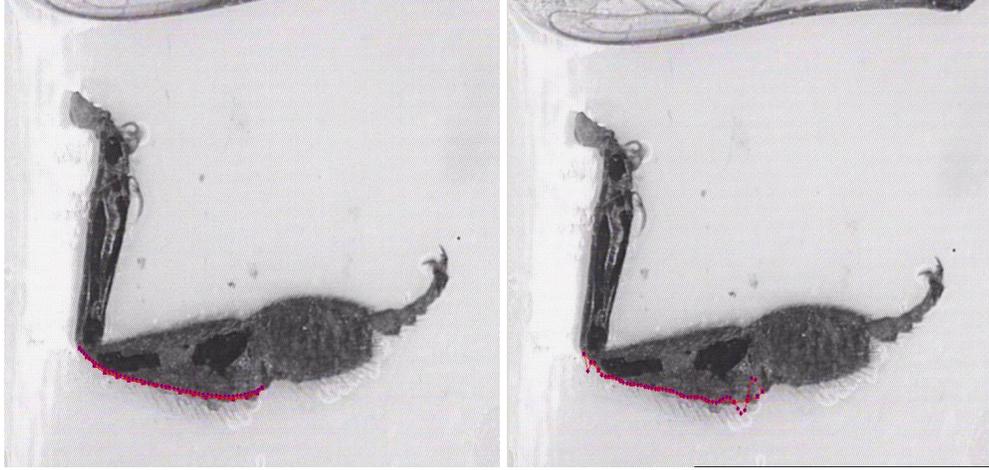
where \bullet is the inner product.

4.4.2 Honeybee Hindleg Feature Extraction

The hindleg template that represents the edges of the hindleg should be formed for feature extraction. The hindleg template consists of 2 contours which are created by taking sample points on the edges of a high quality sample hindleg image. The edges in each contour are formed by the proper choice of Bezier Curves or the Polynomial Interpolation [10] [11] methods as in the wing template. The selection of edge formation method for each edge of the hindleg template is given in Table 4.3. For the hindleg edge that is shown in Figure 4.25, the Bezier curve method should be used for a successful result.

Table 4.3 - Method of formation of the hindleg template edges

Hindleg template edges	Bezier Curve	Polynomial Interpolation
AB	✓	
BC	✓	
CE		✓
EA	✓	
EG	✓	
GJ	✓	
JI	✓	
IH		✓
HF		✓
FD	✓	
DC		✓



(a)

(b)

Figure 4.25 - Hindleg edge formed with (a) Bezier Curve (b) Polynomial Interpolation

The hindleg template formed by Bezier Curves or Polynomial Interpolation methods is shown in Figure 4.26. Only one edge is common in this template.

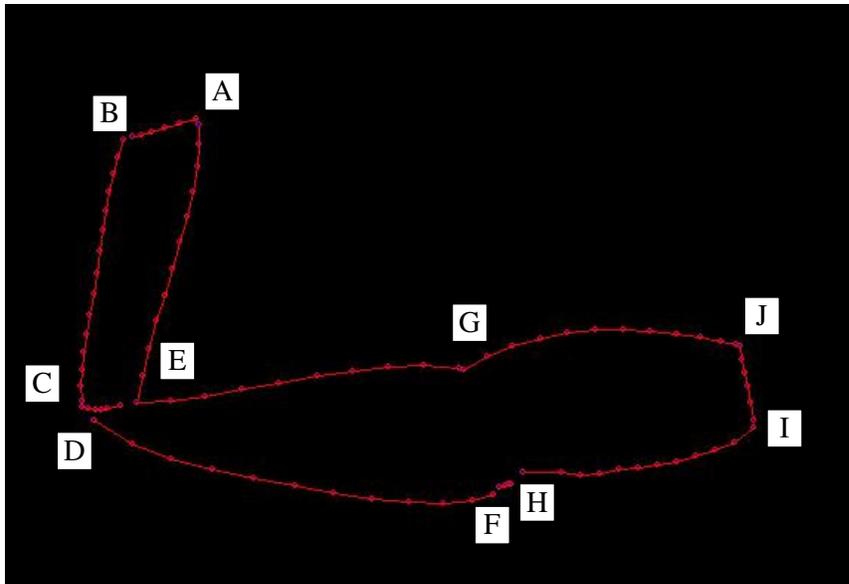


Figure 4.26 - Hindleg template

The steps of applying the active contour algorithm to the hindleg image are given below:

1. Form the edges of the hindleg template by one of the proper curve description method (Table 4.3); Bezier Curve or Polynomial Interpolation. This template is stored and used for every hindleg image.
2. Resize the edges of the hindleg template with respect to the image in Figure 4.17 (b).

3. Only use γ and the neighborhood parameters of the snake that are found experimentally before.
4. Apply the snake algorithm using the contour of the template. While applying the algorithm, do not change the snake points that correspond to the reference points entered by the user.
5. Apply snake algorithm only once.
6. Apply the snake algorithm to the other contour in the hindleg template.

The implementation of the above procedure results as in Figure 4.27.

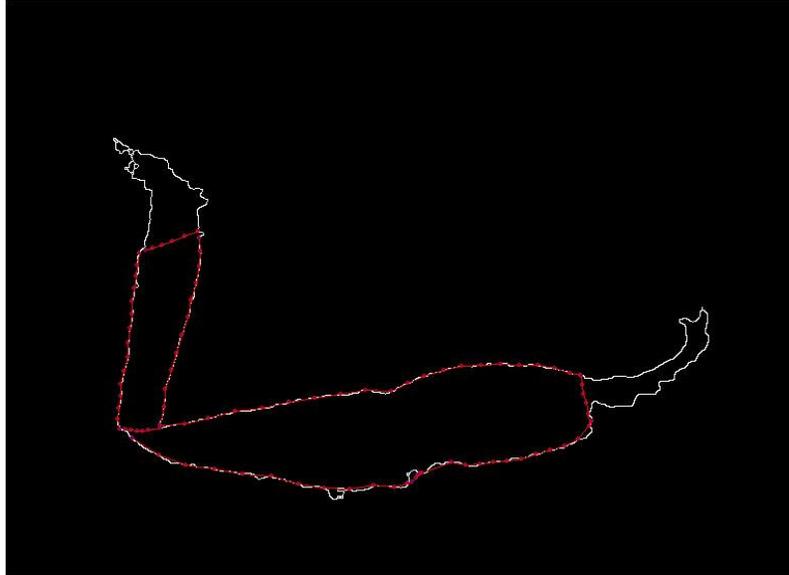


Figure 4.27 - The result of the active contour algorithm applied to the hindleg image

In the hindleg image, only the length parameters are to be measured. The corner points on the hindleg template provide the required points used in the calculations.

CHAPTER 5

RESULTS

In this chapter, the results of honeybee forewing and hindleg length and angle measurements with the proposed system will be discussed. The measured features of the honeybees are compared with the expert measurements.

The honeybee samples used in the measurements are collected from different providences. These providences are different villages of Ula, Marmaris, Milas, Bodrum and Muğla. The collected honeybees are labeled by assigning different numbers to them. Representation of each digit is given in Table 5.1.

Table 5.1 - Labeling of honeybee samples

Location	Village	Bee keeper	Beehive	Sample Number
Ula: 1	Karahörtlen: 1 Elmalı: 2	3 different beekeeper	5 different beehive	5 different samples
Muğla: 2	Yaraş: 1 İkizdere: 2	3 different beekeeper	5 different beehive	5 different samples
Marmaris: 3	Çamlı: 1 Bayır: 2 Aspiran: 3	3 different beekeeper	5 different beehive	5 different samples
Milas: 4	Derince: 1 Karakuyu: 2 Akçalı: 3 Bafa: 4	3 different beekeeper	5 different beehive	5 different samples
Bodrum: 5	Dereköy: 1 Gümüşlük: 2	3 different beekeeper	5 different beehive	5 different samples

A total of 50 hindleg and 50 forewing samples are used in the measurements. The method of measurements carried out by an expert in Biology Department is explained in “2.7 The materials and the methods”. As it was mentioned previously, the feature measurements are handled by placing the rulers on the hindleg or forewing images displayed on the JVC monitor. The measurements taken in this way are named as “expert measurements in cm” in the thesis.

The measurements of the features with the proposed system in which the lengths are in pixels are called as “automatic measurements in pixels”. The length feature measurements in pixels are converted to the expert measurements scaling to cm

for comparing purposes. Conversion is achieved by multiplying all the length features in pixels with a constant scaling factor. This scaling factor is found by taking Metatarsus Width (MW) of hindleg for 10 different honeybees using manual mode of the proposed system, comparing it with the expert measurements and averaging these 10 different scaling factors. The resulting scaling factor after these steps is found to be 4.2 for x and y coordinates. The length measurements multiplied by this factor are called as “scaled automatic measurements in cm”.

The measurements of honeybee features are also carried out using the manual mode of the system. These measurements in which the lengths are in pixels are called as “manual measurements in pixels”. These length feature measurements in pixels are scaled to cm by using the same scaling factor, which are called as “scaled manual measurements in cm”. The manual measurements are accepted as the reference measurements and the difference between Automatic and Manual measurements in cm and the difference between Expert and Manual measurements in cm are calculated.

The expert measurements (lengths in cm, angles in degrees), automatic measurements (lengths in pixels, angles in degrees), scaled automatic measurements (lengths in cm, angles in degrees), manual measurements (lengths in pixels, angles in degrees), scaled manual measurements (lengths in cm, angles in degrees), difference between expert measurements and scaled manual measurements, difference between scaled automatic measurements and scaled manual measurements and percentage difference with respect to the scaled manual measurements for each honeybee features will be given in the tables explained in the following sections. The mean and standard deviation of all measurements, standard deviation of differences and percentage differences for each honeybee features are also given in the tables. The frequency histogram of the differences

between Expert and Manual and frequency histogram of the differences between Automatic and Manual are shown after each table.

5.1 Honeybee Hindleg Feature Results

The results of the hindleg Femur Length (Fe), Tibia Length (Ti), Metatarsus Length (ML) and Metatarsus Width (MW) measurements are given in this section.

5.1.1 Femur Length (Fe)

The measurement results for honeybee feature “Femur Length” are given in Table 5.2 and the difference histograms are shown in Figure 5.1. Mean of differences between Expert and Manual (E/M) measurements for feature Fe is found to be 0.2 cm and related standard deviation is 0.2 cm. Mean and standard deviation of percentage differences (E/M) are -3.0 and 2.3 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.0 and 0.1 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0.3 and 1.4.

Table 5.2 - Measurement Results for Hindleg Fe (Femur Length)

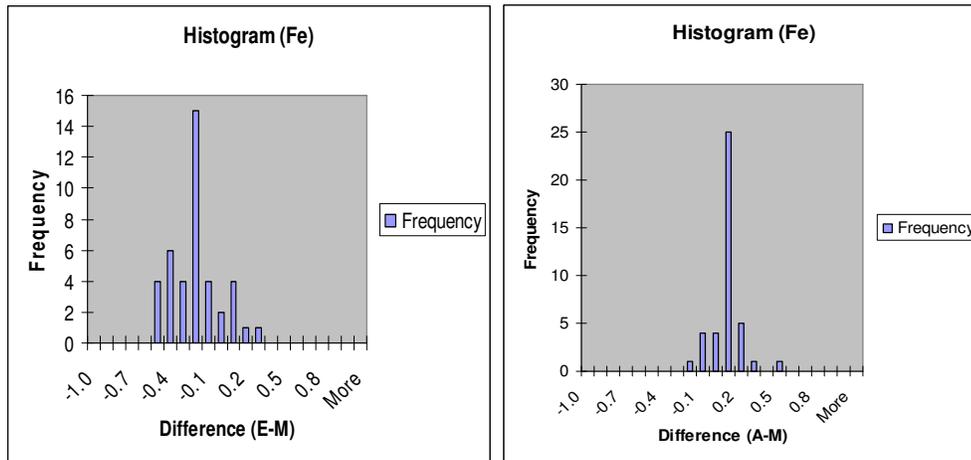
Fe (Femur Length)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	7.7	191.3	8.0	188.8	7.9	-0.2	-2.5	0.1	1.3
11215	7.6	189.3	7.9	185.4	7.7	-0.1	-1.3	0.2	2.6

Table 5.2 (Continued)

1223	8.1	198.3	8.3	197.6	8.2	-0.1	-1.2	0.1	1.2
11224	7.9	192.0	8.0	191.8	8.0	-0.1	-1.3	0.0	0.0
11231	7.6	193.7	8.1	190.2	7.9	-0.3	-3.8	0.2	2.5
11235	7.4	193.8	8.1	191.8	8.0	-0.6	-7.5	0.1	1.3
11244	7.6	186.5	7.8	187.5	7.8	-0.2	-2.6	0.0	0.0
11252	7.6	191.5	8.0	192.8	8.0	-0.4	-5.0	0.0	0.0
11253	7.8	188.5	7.8	188.6	7.8	0.0	0.0	0.0	0.0
12113	7.4	184.3	7.7	185.1	7.7	-0.3	-3.9	0.0	0.0
12115	7.4	187.3	7.8	187.2	7.8	-0.4	-5.1	0.0	0.0
12133	7.3	183.7	7.6	180.7	7.5	-0.2	-2.7	0.1	1.3
12313	7.4	189.4	7.9	189.5	7.9	-0.5	-6.3	0.0	0.0
12314	7.8	196.0	8.2	192.3	8.0	-0.2	-2.5	0.2	2.5
12354	7.8	198.0	8.2	197.8	8.2	-0.4	-4.9	0.0	0.0
12356	7.8	195.2	8.1	195.6	8.1	-0.3	-3.7	0.0	0.0
22211	7.6	186.4	7.8	185.6	7.7	-0.1	-1.3	0.1	1.3
22212	7.7	178.4	7.4	180.4	7.5	0.2	2.7	-0.1	-1.3
22242	8.0	193.0	8.0	191.1	8.0	0.0	0.0	0.0	0.0
22245	7.6	186.5	7.8	186.3	7.8	-0.2	-2.6	0.0	0.0
22255	7.6	188.9	7.9	190.5	7.9	-0.3	-3.8	0.0	0.0
22256	7.8	186.2	7.7	188.0	7.8	0.0	0.0	-0.1	-1.3
31111	7.7	186.8	7.8	189.7	7.9	-0.2	-2.5	-0.1	-1.3
31311	8.2	202.3	8.4	200.8	8.4	-0.2	-2.4	0.0	0.0
31313	7.8	188.1	7.8	188.7	7.9	-0.1	-1.3	-0.1	-1.3
33215	7.6	180.0	7.5	180.6	7.5	0.1	1.3	0.0	0.0
33217	7.6	191.1	8.0	191.6	8.0	-0.4	-5.0	0.0	0.0
33231	7.7	194.0	8.1	191.5	8.0	-0.3	-3.8	0.1	1.3
33233	7.9	195.4	8.1	195.4	8.1	-0.2	-2.5	0.0	0.0
33327	7.3	182.6	7.6	181.9	7.6	-0.3	-3.9	0.0	0.0
41123	8.0	210.5	8.8	201.4	8.4	-0.4	-4.8	0.4	4.8
41126	7.7	187.3	7.8	190.3	7.9	-0.2	-2.5	-0.1	-1.3
42323	7.9	189.0	7.9	188.2	7.8	0.1	1.3	0.1	1.3
42326	7.6	196.0	8.2	192.8	8.0	-0.4	-5.0	0.2	2.5
42341	7.9	196.8	8.2	200.2	8.3	-0.4	-4.8	-0.1	-1.2
42344	7.4	191.8	8.0	192.0	8.0	-0.6	-7.5	0.0	0.0
51113	7.8	194.0	8.1	198.6	8.3	-0.5	-6.0	-0.2	-2.4
51121	7.8	191.5	8.0	195.5	8.1	-0.3	-3.7	-0.1	-1.2
51123	7.8	192.6	8.0	194.6	8.1	-0.3	-3.7	-0.1	-1.2

Table 5.2 (Continued)

51132	7.6	191.3	8.0	192.1	8.0	-0.4	-5.0	0.0	0.0
51134	7.5	186.7	7.8	184.5	7.7	-0.2	-2.6	0.1	1.3
MEAN	7.7	190.9	7.9	190.6	7.9	-0.2	-3.0	0.0	0.3
STDEV	0.2	5.9	0.2	5.4	0.2	0.2	2.3	0.1	1.4



(a)

(b)

Figure 5.1 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Fe (Femur Length)

5.1.2 Tibia Length (Ti)

The measurement results for honeybee feature “Tibia Length” are given in Table 5.3 and the difference histograms are shown in Figure 5.2. Mean of differences between Expert and Manual (E/M) measurements for feature Ti is found to be 0.0 cm and related standard deviation is 0.2 cm. Mean and standard deviation of percentage differences (E/M) are 0.4 and 2.4 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.1 and 0.2 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0.6 and 1.6.

Table 5.3 - Measurement Results for Hindleg Ti (Tibia Length)

Ti (Tibia Length)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	9.7	234.4	9.8	230.4	9.6	0.1	1.0	0.2	2.1
11215	10.0	237.6	9.9	239.1	9.9	0.1	1.0	0.0	0.0
11223	10.8	256.3	10.7	255.9	10.7	0.1	0.9	0.0	0.0
11224	10.1	240.9	10.0	240.2	10.0	0.1	1.0	0.0	0.0
11231	10.3	249.3	10.4	249.5	10.4	-0.1	-1.0	0.0	0.0
11235	10.1	245.0	10.2	244.3	10.2	-0.1	-1.0	0.0	0.0
11244	9.2	228.9	9.5	225.9	9.4	-0.2	-2.1	0.1	1.1
11252	10.0	229.9	9.6	231.3	9.6	0.4	4.2	0.0	0.0
11253	10.0	243.2	10.1	239.4	10.0	0.0	0.0	0.1	1.0
12113	10.4	245.2	10.2	242.9	10.1	0.3	3.0	0.1	1.0
12115	9.8	237.7	9.9	236.0	9.8	0.0	0.0	0.1	1.0
12133	9.3	222.6	9.3	219.5	9.1	0.2	2.2	0.2	2.2
12313	9.8	230.9	9.6	227.2	9.5	0.3	3.2	0.1	1.1

Table 5.3 (Continued)

12314	10.3	239.2	10.0	239.0	9.9	0.4	4.0	0.1	1.0
12354	10.3	242.2	10.1	247.6	10.3	0.0	0.0	-0.2	-1.9
12356	10.1	246.1	10.2	241.9	10.1	0.0	0.0	0.1	1.0
22211	9.6	235.0	9.8	235.5	9.8	-0.2	-2.0	0.0	0.0
22212	9.9	228.9	9.5	229.0	9.5	0.4	4.2	0.0	0.0
22242	10.2	241.1	10.0	241.1	10.0	0.2	2.0	0.0	0.0
22245	10.1	236.0	9.8	238.7	9.9	0.2	2.0	-0.1	-1.0
22255	10.1	250.1	10.4	241.2	10.0	0.1	1.0	0.4	4.0
22256	9.9	238.6	9.9	234.0	9.7	0.2	2.1	0.2	2.1
31111	10.2	245.8	10.2	243.6	10.1	0.1	1.0	0.1	1.0
31311	10.5	256.3	10.7	255.0	10.6	-0.1	-0.9	0.1	0.9
31313	9.6	225.8	9.4	227.2	9.5	0.1	1.1	-0.1	-1.1
33215	9.8	230.9	9.6	229.8	9.6	0.2	2.1	0.0	0.0
33217	9.2	221.6	9.2	233.5	9.7	-0.5	-5.2	-0.5	-5.2
33231	9.8	239.1	10.0	236.8	9.9	-0.1	-1.0	0.1	1.0
33233	9.9	243.0	10.1	239.3	10.0	-0.1	-1.0	0.1	1.0
33327	9.2	226.7	9.4	220.6	9.2	0.0	0.0	0.2	2.2
41123	9.9	222.6	9.3	227.5	9.5	0.4	4.2	-0.2	-2.1
41126	9.7	233.3	9.7	231.3	9.6	0.1	1.0	0.1	1.0
42323	9.6	233.0	9.7	231.7	9.6	0.0	0.0	0.1	1.0
42326	9.0	230.9	9.6	226.1	9.4	-0.4	-4.3	0.2	2.1
42341	10.0	244.8	10.2	236.3	9.8	0.2	2.0	0.4	4.1
42344	9.8	244.8	10.2	244.3	10.2	-0.4	-3.9	0.0	0.0
51113	9.8	247.0	10.3	248.3	10.3	-0.5	-4.9	0.0	0.0
51121	10.1	245.0	10.2	238.6	9.9	0.2	2.0	0.3	3.0
51123	9.8	237.6	9.9	236.7	9.9	-0.1	-1.0	0.0	0.0
51132	10.0	243.2	10.1	243.8	10.1	-0.1	-1.0	0.0	0.0
51134	9.3	229.9	9.6	229.2	9.5	-0.2	-2.1	0.1	1.1
MEAN	9.9	238.0	9.9	236.8	9.9	0.0	0.3	0.1	0.6
STDEV	0.4	8.7	0.4	8.4	0.4	0.2	2.4	0.2	1.6

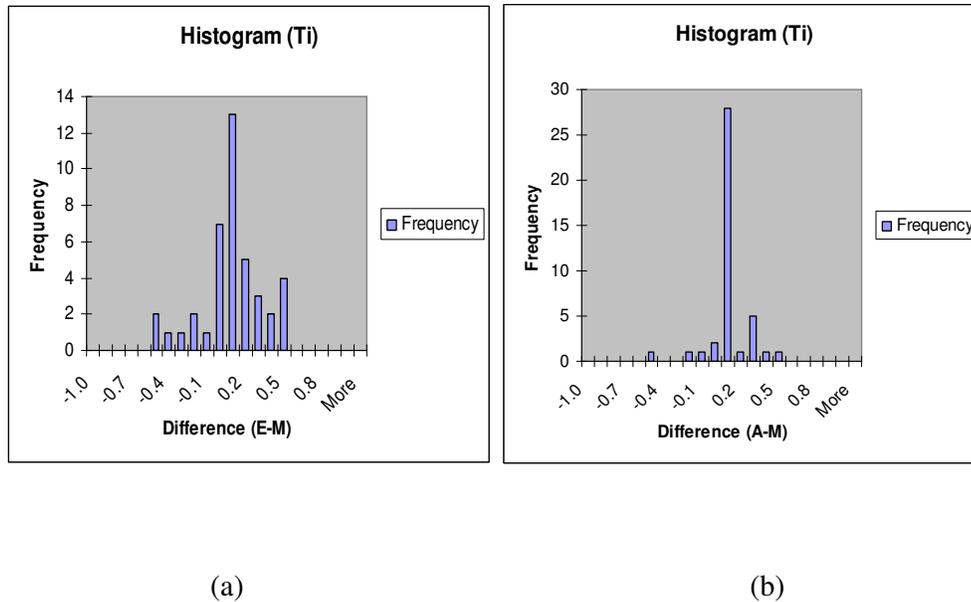


Figure 5.2 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Ti (Tibia Length)

5.1.3 Metatarsus Length (ML)

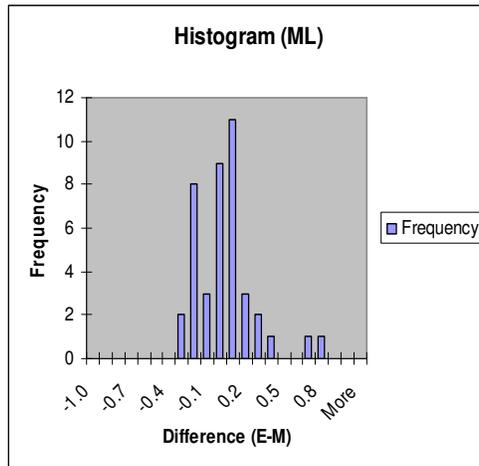
The measurement results for “Metatarsus Length” are given in Table 5.4 and the difference histograms are shown in Figure 5.3. Mean of differences between Expert and Manual (E/M) measurements is 0.0 cm and standard deviation is 0.2 cm. Mean and standard deviation of percentage differences (E/M) are -0.2 and 3.6 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.0 and 0.1 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are -0.8 and 2.1.

Table 5.4 - Measurement Results for Hindleg ML (Metatarsus Length)

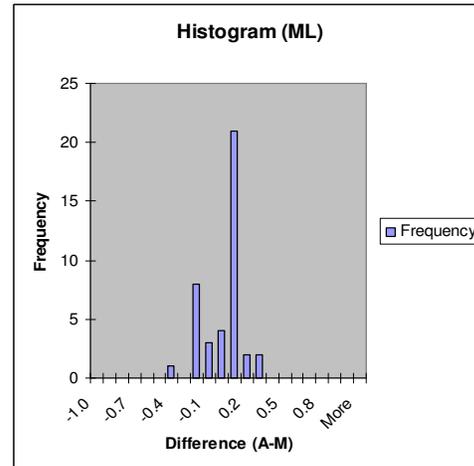
ML (Metatarsus Length)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	%Diff (E/M)	Diff (A/M) (cm)	%Diff (A/M)
11213	6.0	145.2	6.0	144.3	6.0	0.0	0.0	0.0	0.0
11215	6.3	148.8	6.2	152.1	6.3	0.0	0.0	-0.1	-1.6
11223	6.9	157.8	6.6	158.1	6.6	0.3	4.6	0.0	0.0
11224	6.3	152.3	6.3	153.4	6.4	-0.1	-1.6	-0.1	-1.6
11231	6.4	159.5	6.6	155.2	6.5	-0.1	-1.5	0.1	1.5
11235	6.2	152.1	6.3	151.0	6.3	-0.1	-1.6	0.0	0.0
11244	6.0	140.8	5.9	140.2	5.8	0.2	3.4	0.1	1.7
11252	6.2	151.2	6.3	153.4	6.4	-0.2	-3.1	-0.1	-1.6
11253	6.2	152.8	6.4	150.9	6.3	-0.1	-1.6	0.1	1.6
12113	6.3	155.3	6.5	156.4	6.5	-0.2	-3.1	0.0	0.0
12115	6.1	153.8	6.4	154.1	6.4	-0.3	-4.7	0.0	0.0
12133	5.9	138.1	5.7	143.4	6.0	-0.1	-1.7	-0.3	-5.0
12313	6.1	145.2	6.0	149.5	6.2	-0.1	-1.6	-0.2	-3.2
12314	6.2	152.7	6.4	154.8	6.4	-0.2	-3.1	0.0	0.0
12354	6.3	150.3	6.3	150.0	6.2	0.1	1.6	0.1	1.6
12356	6.2	148.7	6.2	152.4	6.3	-0.1	-1.6	-0.1	-1.6
22211	5.4	133.0	5.5	137.6	5.7	-0.3	-5.2	-0.2	-3.5
22212	6.1	141.0	5.9	140.6	5.9	0.2	3.4	0.0	0.0
22242	6.4	154.8	6.4	154.8	6.4	0.0	0.0	0.0	0.0
22245	6.4	147.8	6.1	151.2	6.3	0.1	1.6	-0.2	-3.2
22255	6.6	156.6	6.5	156.3	6.5	0.1	1.5	0.0	0.0
22256	6.4	150.3	6.3	149.0	6.2	0.2	3.2	0.1	1.6
31111	6.1	148.5	6.2	149.5	6.2	-0.1	-1.6	0.0	0.0
31311	6.5	161.9	6.7	161.2	6.7	-0.2	-3.0	0.0	0.0
31313	6.2	127.5	5.3	132.8	5.5	0.7	12.7	-0.2	-3.6
33215	6.0	145.0	6.0	145.4	6.1	-0.1	-1.7	-0.1	-1.6
33217	6.0	147.1	6.1	150.6	6.3	-0.3	-4.8	-0.2	-3.2
33231	6.1	145.1	6.0	144.8	6.0	0.1	1.7	0.0	0.0
33233	6.2	152.4	6.3	150.9	6.3	-0.1	-1.6	0.0	0.0
33327	5.9	143.2	6.0	143.5	6.0	-0.1	-1.7	0.0	0.0
41123	6.3	151.8	6.3	155.1	6.5	-0.2	-3.1	-0.2	-3.1
41126	6.2	144.8	6.0	150.6	6.3	-0.1	-1.6	-0.3	-4.8
42323	6.2	147.1	6.1	148.9	6.2	0.0	0.0	-0.1	-1.6
42326	6.2	131.5	5.5	135.7	5.6	0.6	10.6	-0.1	-1.8

Table 5.4 (Continued)

42341	6.1	143.2	6.0	154.0	6.4	-0.3	-4.7	-0.4	-6.3
42344	6.0	147.9	6.2	149.4	6.2	-0.2	-3.2	0.0	0.0
51113	6.2	151.6	6.3	147.4	6.1	0.1	1.6	0.2	3.3
51121	6.3	157.3	6.5	152.4	6.3	0.0	0.0	0.2	3.2
51123	6.3	150.8	6.3	149.6	6.2	0.1	1.6	0.1	1.6
51132	6.3	150.4	6.3	152.0	6.3	0.0	0.0	0.0	0.0
51134	6.0	143.0	6.0	143.6	6.0	0.0	0.0	0.0	0.0
MEAN	6.2	148.2	6.2	149.4	6.2	0.0	-0.2	0.0	-0.8
STDEV	0.2	7.3	0.3	6.1	0.3	0.2	3.6	0.1	2.1



(a)



(b)

Figure 5.3 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Metatarsus Length (ML)

5.1.4 Metatarsus Width (MW)

The measurement results for honeybee feature “Metatarsus Width” are given in Table 5.5 and the difference histograms are shown in Figure 5.4. Mean of differences between Expert and Manual (E/M) measurements for feature MW is found to be -0.2 cm and related standard deviation is 0.1 cm. Mean and standard deviation of percentage differences (E/M) are -6.1 and 3.1 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.1 and 0.2 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 2.2 and 4.1.

Table 5.5 - Measurement Results for Hindleg MW (Metatarsus Width)

MW (Metatarsus Width)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	3.5	92.6	3.9	90.7	3.8	-0.3	-7.9	0.1	2.6
11215	3.6	91.2	3.8	91.8	3.8	-0.2	-5.3	0.0	0.0
11223	3.7	92.3	3.8	90.6	3.8	-0.1	-2.6	0.0	0.0
11224	3.7	97.5	4.1	90.7	3.8	-0.1	-2.6	0.3	7.9
11231	3.6	89.1	3.7	92.3	3.8	-0.2	-5.3	-0.1	-2.6
11235	3.6	97.0	4.0	89.9	3.7	-0.1	-2.7	0.3	8.1
11244	3.7	105.6	4.4	104.6	4.4	-0.7	-15.9	0.0	0.0
11252	3.4	89.2	3.7	83.5	3.5	-0.1	-2.9	0.2	5.7
11253	3.7	90.9	3.8	91.7	3.8	-0.1	-2.6	0.0	0.0
12113	3.4	85.6	3.6	86.8	3.6	-0.2	-5.6	0.0	0.0
12115	3.5	89.1	3.7	89.8	3.7	-0.2	-5.4	0.0	0.0
12133	3.3	85.1	3.5	82.3	3.4	-0.1	-2.9	0.1	2.9
12313	3.5	89.6	3.7	90.1	3.7	-0.2	-5.4	0.0	0.0

Table 5.5 (Continued)

12314	3.5	90.1	3.7	88.7	3.7	-0.2	-5.4	0.0	0.0
12354	3.6	100.7	4.2	93.4	3.9	-0.3	-7.7	0.3	7.7
12356	3.5	89.5	3.7	87.9	3.7	-0.2	-5.4	0.0	0.0
22211	3.4	88.3	3.7	91.3	3.8	-0.4	-10.5	-0.1	-2.6
22212	3.5	92.3	3.8	87.0	3.6	-0.1	-2.8	0.2	5.6
22242	3.5	89.5	3.7	86.0	3.6	-0.1	-2.8	0.1	2.8
22245	3.5	89.0	3.7	90.5	3.8	-0.3	-7.9	-0.1	-2.6
22255	3.5	86.1	3.6	88.0	3.7	-0.2	-5.4	-0.1	-2.7
22256	3.6	91.6	3.8	91.4	3.8	-0.2	-5.3	0.0	0.0
31111	3.5	94.4	3.9	91.3	3.8	-0.3	-7.9	0.1	2.6
31311	3.6	98.8	4.1	92.1	3.8	-0.2	-5.3	0.3	7.9
31313	3.6	93.7	3.9	93.2	3.9	-0.3	-7.7	0.0	0.0
33215	3.4	87.7	3.6	87.7	3.7	-0.3	-8.1	-0.1	-2.7
33217	3.5	95.2	4.0	89.1	3.7	-0.2	-5.4	0.3	8.1
33231	3.3	93.4	3.9	87.4	3.6	-0.3	-8.3	0.3	8.3
33233	3.6	93.0	3.9	90.7	3.8	-0.2	-5.3	0.1	2.6
33327	3.3	97.4	4.1	85.6	3.6	-0.3	-8.3	0.5	13.9
41123	3.5	90.0	3.7	93.2	3.9	-0.4	-10.3	-0.2	-5.1
41126	3.5	90.4	3.8	88.4	3.7	-0.2	-5.4	0.1	2.7
42323	3.5	87.2	3.6	85.1	3.5	0.0	0.0	0.1	2.9
42326	3.4	90.5	3.8	89.0	3.7	-0.3	-8.1	0.1	2.7
42341	3.4	91.0	3.8	88.3	3.7	-0.3	-8.1	0.1	2.7
42344	3.6	91.1	3.8	89.1	3.7	-0.1	-2.7	0.1	2.7
51113	3.4	95.9	4.0	89.6	3.7	-0.3	-8.1	0.3	8.1
51121	3.5	91.4	3.8	89.7	3.7	-0.2	-5.4	0.1	2.7
51123	3.7	92.7	3.9	90.4	3.8	-0.1	-2.6	0.1	2.6
51132	3.3	88.8	3.7	90.6	3.8	-0.5	-13.2	-0.1	-2.6
51134	3.4	86.2	3.6	89.5	3.7	-0.3	-8.1	-0.1	-2.7
MEAN	3.5	91.7	3.8	89.7	3.7	-0.2	-6.1	0.1	2.2
STDEV	0.1	4.2	0.2	3.4	0.1	0.1	3.1	0.2	4.1

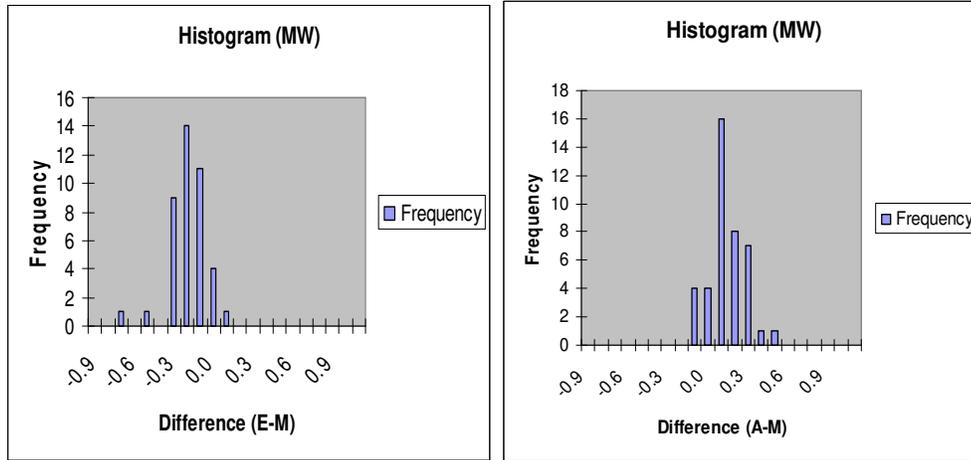


Figure 5.4 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Hindleg Metatarsus Width (MW)

5.2 Honeybee Forewing Feature Results

The results of the length and angle measurements for each the honeybee forewing features are given in this section. Only the Length 1 (L1) and Length 2 (L2) feature measurements are not given although they are carried out in the system because the expert had not taken these two feature measurements. Instead, result of Forewing Length which is the sum of these two features ($FL = F1+F2$) is given.

5.2.1 Forewing Length (FL = L1+L2)

The measurement results for honeybee feature “Forewing Length” are given in Table 5.6 and the difference histograms are shown in Figure 5.5. Mean of differences between Expert and Manual (E/M) measurements for feature FL is found to be 0.1 cm and related standard deviation is 0.6 cm. Mean and standard deviation of percentage differences (E/M) are 0.2 and 2.0 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.0 and 0.2 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0.0 and 0.5.

This feature is extracted from the reference points 1 and 3 which are entered by the user in the beginning of the program. The reference point 1 must be entered because this critical point is on the transparent part of the forewing. Since these points are entered by the user, the correctness of FL measurement is increased.

Table 5.6 - Measurement Results for Forewing Length (FL)

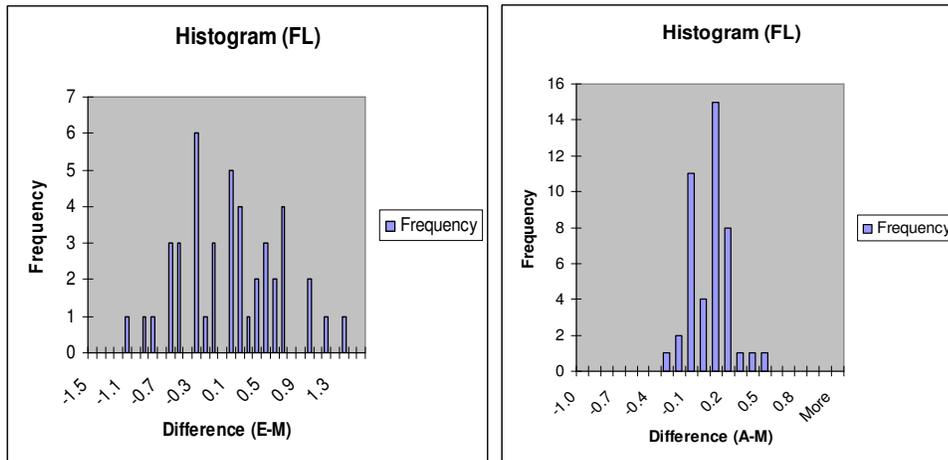
FL (Forewing Length)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	29.0	680.1	28.3	679.2	28.3	0.7	2.5	0.0	0.0
11215	28.9	683.1	28.4	684.4	28.5	0.4	1.4	-0.1	-0.4
11223	30.0	712.1	29.6	707.5	29.4	0.6	2.0	0.2	0.7
11224	29.2	691.1	28.8	688.2	28.6	0.6	2.1	0.2	0.7
11231	29.8	702.1	29.2	699.4	29.1	0.7	2.4	0.1	0.3
11235	29.3	685.1	28.5	682.6	28.4	0.9	3.2	0.1	0.4

Table 5.6 (Continued)

11241	29.7	680.1	28.3	682.2	28.4	1.3	4.6	-0.1	-0.4
11244	29.1	674.1	28.1	676.1	28.1	1.0	3.6	0.0	0.0
11252	29.0	666.1	27.7	666.8	27.8	1.2	4.3	-0.1	-0.4
11253	27.8	681.1	28.3	679.2	28.3	-0.5	-1.8	0.0	0.0
12113	29.0	687.1	28.6	684.7	28.5	0.5	1.8	0.1	0.4
12115	28.4	668.1	27.8	667.5	27.8	0.6	2.2	0.0	0.0
12133	26.6	657.1	27.3	656.9	27.3	-0.7	-2.6	0.0	0.0
12137	28.5	681.3	28.4	680.4	28.3	0.2	0.7	0.1	0.4
12313	28.7	677.1	28.2	679.4	28.3	0.4	1.4	-0.1	-0.4
12314	28.1	680.1	28.3	671.2	27.9	0.2	0.7	0.4	1.4
12354	27.7	689.1	28.7	692.2	28.8	-1.1	-3.8	-0.1	-0.3
12356	29.5	692.1	28.8	692.3	28.8	0.7	2.4	0.0	0.0
22212	28.6	686.1	28.6	687.1	28.6	0.0	0.0	0.0	0.0
22242	28.1	690.1	28.7	690.0	28.7	-0.6	-2.1	0.0	0.0
22245	28.2	689.0	28.7	692.2	28.8	-0.6	-2.1	-0.1	-0.3
22255	28.6	693.1	28.8	686.1	28.6	0.0	0.0	0.2	0.7
22256	29.2	695.1	28.9	690.4	28.7	0.5	1.7	0.2	0.7
31111	26.7	656.1	27.3	652.7	27.2	-0.5	-1.8	0.1	0.4
31113	28.5	680.1	28.3	689.9	28.7	-0.2	-0.7	-0.4	-1.4
31311	28.8	698.1	29.1	700.5	29.2	-0.4	-1.4	-0.1	-0.3
31313	28.7	687.1	28.6	690.1	28.7	0.0	0.0	-0.1	-0.3
33213	27.0	663.1	27.6	670.6	27.9	-0.9	-3.2	-0.3	-1.1
33215	26.9	659.1	27.4	661.1	27.5	-0.6	-2.2	-0.1	-0.4
33217	27.9	676.0	28.1	678.1	28.2	-0.3	-1.1	-0.1	-0.4
33231	28.0	671.1	27.9	673.2	28.0	0.0	0.0	-0.1	-0.4
33233	28.1	687.1	28.6	684.4	28.5	-0.4	-1.4	0.1	0.4
33237	27.3	660.1	27.5	661.7	27.5	-0.2	-0.7	0.0	0.0
41123	28.8	680.1	28.3	681.0	28.3	0.5	1.8	0.0	0.0
41126	27.7	669.1	27.8	672.2	28.0	-0.3	-1.1	-0.2	-0.7
42323	27.2	654.1	27.2	658.2	27.4	-0.2	-0.7	-0.2	-0.7
42324	27.9	672.1	28.0	675.8	28.1	-0.2	-0.7	-0.1	-0.4
42341	27.8	675.1	28.1	676.2	28.1	-0.3	-1.1	0.0	0.0
42344	28.5	676.1	28.1	683.0	28.4	0.1	0.4	-0.3	-1.1
51113	27.9	693.1	28.8	692.5	28.8	-0.9	-3.1	0.0	0.0
51121	27.8	677.1	28.2	675.7	28.1	-0.3	-1.1	0.1	0.4
51123	28.3	677.1	28.2	681.0	28.3	0.0	0.0	-0.1	-0.4

Table 5.6 (Continued)

51132	28.0	674.1	28.1	667.5	27.8	0.2	0.7	0.3	1.1
51134	27.9	667.1	27.8	665.7	27.7	0.2	0.7	0.1	0.4
MEAN	28.3	679.4	28.3	679.7	28.3	0.1	0.2	0.0	0.0
STDEV	0.8	12.7	0.5	12.1	0.5	0.6	2.0	0.2	0.5



(a)

(b)

Figure 5.5 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Forewing Length (FL)

5.2.2 Forewing Width (FW)

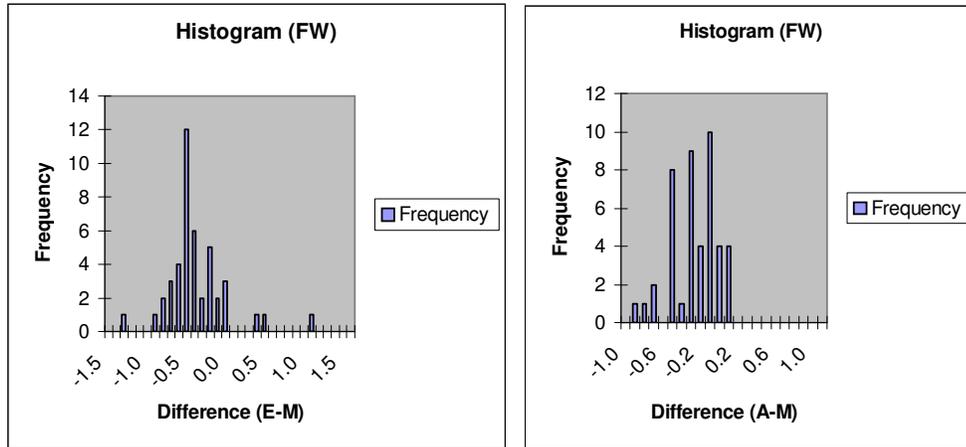
The measurement results for honeybee feature “Forewing Width” are given in Table 5.7 and the difference histograms are in Figure 5.6. Mean of differences between Expert and Manual (E/M) measurements for feature FW is found to be -0.4 cm and related standard deviation is 0.4 cm. Mean and standard deviation of percentage differences (E/M) are -4.3 and 4.1 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are -0.3 and 0.2 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are -3.3 and 2.4.

Table 5.7 - Measurement Results for Forewing Width (FW)

FW (Forewing Width)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	9.3	229.0	9.5	236.0	9.8	-0.5	-5.1	-0.3	-3.1
11215	9.2	220.0	9.2	230.0	9.6	-0.4	-4.2	-0.4	-4.2
11223	9.6	235.0	9.8	241.0	10.0	-0.4	-4.0	-0.2	-2.0
11224	9.3	225.0	9.4	230.0	9.6	-0.3	-3.1	-0.2	-2.1
11231	9.3	228.0	9.5	240.2	10.0	-0.7	-7.0	-0.5	-5.0
11235	9.2	224.0	9.3	232.1	9.7	-0.5	-5.2	-0.4	-4.1
11241	9.3	225.0	9.4	240.2	10.0	-0.7	-7.0	-0.6	-6.0
11244	9.5	223.0	9.3	233.4	9.7	-0.2	-2.1	-0.4	-4.1
11252	9.2	230.0	9.6	233.1	9.7	-0.5	-5.2	-0.1	-1.0
11253	9.3	225.0	9.4	231.2	9.6	-0.3	-3.1	-0.2	-2.1
12113	9.1	220.0	9.2	232.1	9.7	-0.6	-6.2	-0.5	-5.2
12115	9.1	218.0	9.1	230.1	9.6	-0.5	-5.2	-0.5	-5.2
12133	8.6	193.0	8.0	200.0	8.2	0.4	4.9	-0.2	-2.4

Table 5.7 (Continued)

12137	9.5	218.0	9.1	231.7	9.6	-0.1	-1.0	-0.5	-5.2
12313	8.8	207.0	8.6	227.1	9.5	-0.7	-7.4	-0.9	-9.5
12314	8.7	216.0	9.0	230.0	9.6	-0.9	-9.4	-0.6	-6.3
12354	9.3	213.0	8.9	233.0	9.7	-0.4	-4.1	-0.8	-8.2
12356	9.4	233.0	9.7	234.2	9.7	-0.3	-3.1	0.0	0.0
22212	8.9	212.0	8.8	217.3	9.0	-0.1	-1.1	-0.2	-2.2
22242	9.5	234.0	9.7	234.1	9.7	-0.2	-2.1	0.0	0.0
22245	9.4	187.0	7.8	200.0	8.3	1.1	13.3	-0.5	-6.0
22255	8.9	210.0	8.7	221.1	9.2	-0.3	-3.3	-0.5	-5.4
22256	9.1	229.0	9.5	227.9	9.5	-0.4	-4.2	0.0	0.0
31111	9.0	221.0	9.2	227.0	9.4	-0.4	-4.3	-0.2	-2.1
31113	9.1	230.0	9.6	236.0	9.8	-0.7	-7.1	-0.2	-2.0
31311	9.3	220.0	9.2	238.0	9.9	-0.6	-6.1	-0.7	-7.1
31313	9.3	211.0	8.8	234.0	9.7	-0.4	-4.1	-0.9	-9.3
33213	8.7	221.0	9.2	225.4	9.4	-0.7	-7.4	-0.2	-2.1
33215	8.2	222.0	9.2	229.1	9.5	-1.3	-13.7	-0.3	-3.2
33217	9.3	218.0	9.1	225.2	9.4	-0.1	-1.1	-0.3	-3.2
33231	8.7	220.0	9.2	223.8	9.3	-0.6	-6.5	-0.1	-1.1
33233	9.2	235.0	9.8	241.0	10.0	-0.8	-8.0	-0.2	-2.0
33237	8.7	222.0	9.2	227.5	9.5	-0.8	-8.4	-0.3	-3.2
41123	8.7	220.0	9.2	226.2	9.4	-0.7	-7.4	-0.2	-2.1
41126	8.9	224.0	9.3	226.1	9.4	-0.5	-5.3	-0.1	-1.1
42323	8.9	218.0	9.1	225.4	9.4	-0.5	-5.3	-0.3	-3.2
42324	9.3	208.0	8.7	210.0	9.0	0.3	3.3	-0.3	-3.3
42341	9.1	227.0	9.4	230.8	9.6	-0.5	-5.2	-0.2	-2.1
42344	9.3	231.0	9.6	231.5	9.6	-0.3	-3.1	0.0	0.0
51113	8.9	223.0	9.3	225.3	9.4	-0.5	-5.3	-0.1	-1.1
51121	9.1	226.0	9.4	228.1	9.5	-0.4	-4.2	-0.1	-1.1
51123	8.9	222.0	9.2	227.4	9.5	-0.6	-6.3	-0.3	-3.2
51132	8.7	215.0	8.9	219.0	9.1	-0.4	-4.4	-0.2	-2.2
51134	8.7	217.0	9.0	221.1	9.2	-0.5	-5.4	-0.2	-2.2
MEAN	9.1	220.6	9.2	228.3	9.5	-0.4	-4.3	-0.3	-3.3
STDEV	0.3	9.7	0.4	8.9	0.4	0.4	4.1	0.2	2.4



(a)

(b)

Figure 5.6 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Forewing Width (FW)

5.2.3 Cubital index a

The measurement results for honeybee feature “Cubital index a” are given in Table 5.8 and the difference histograms are shown in Figure 5.7. Mean of differences between Expert and Manual (E/M) measurements for feature “Cubital index a” is found to be 0.0 cm and related standard deviation is 0.2 cm. Mean and standard deviation of percentage differences (E/M) are -1.2 and 10.9 respectively. Mean and standard deviation of differences between Automatic and Manual

(A/M) are 0.0 and 0.2 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0.5 and 14.6.

Table 5.8 - Measurement Results for Cubital index a

a (Cubital Index)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	1.6	38.6	1.6	37.0	1.5	0.1	6.7	0.1	6.7
11215	1.3	32.2	1.3	33.6	1.4	-0.1	-7.1	-0.1	-7.1
11223	1.6	38.0	1.6	36.1	1.5	0.1	6.7	0.1	6.7
11224	1.8	43.7	1.8	41.8	1.7	0.1	5.9	0.1	5.9
11231	1.7	37.3	1.6	39.9	1.7	0.0	0.0	-0.1	-5.9
11235	1.5	36.8	1.5	37.7	1.6	-0.1	-6.3	-0.1	-6.3
11241	1.4	34.5	1.4	35.4	1.5	-0.1	-6.7	-0.1	-6.7
11244	1.5	35.5	1.5	33.6	1.4	0.1	7.1	0.1	7.1
11252	1.6	37.7	1.6	38.9	1.6	0.0	0.0	0.0	0.0
11253	1.9	37.0	1.5	44.9	1.9	0.0	0.0	-0.4	-21.1
12113	1.9	43.5	1.8	45.2	1.9	0.0	0.0	-0.1	-5.3
12115	1.8	40.7	1.7	44.6	1.9	-0.1	-5.3	-0.2	-10.5
12133	1.4	42.5	1.8	38.9	1.6	-0.2	-12.5	0.2	12.5
12137	1.4	34.0	1.4	35.0	1.5	-0.1	-6.7	-0.1	-6.7
12313	1.7	37.9	1.6	41.5	1.7	0.0	0.0	-0.1	-5.9
12314	1.8	39.0	1.6	39.0	1.6	0.2	12.5	0.0	0.0
12354	1.6	37.3	1.6	39.9	1.7	-0.1	-5.9	-0.1	-5.9
12356	1.5	28.6	1.2	37.3	1.6	-0.1	-6.3	-0.4	-25.0
22212	1.2	33.6	1.4	36.8	1.5	-0.3	-20.0	-0.1	-6.7
22242	1.7	43.7	1.8	47.0	2.0	-0.3	-15.0	-0.2	-10.0
22245	1.4	34.2	1.4	35.8	1.5	-0.1	-6.7	-0.1	-6.7
22255	1.8	47.2	2.0	44.4	1.8	0.0	0.0	0.2	11.1
22256	1.6	38.9	1.6	39.8	1.7	-0.1	-5.9	-0.1	-5.9
31111	1.7	38.9	1.6	42.2	1.8	-0.1	-5.6	-0.2	-11.1

Table 5.8 (Continued)

31113	1.8	34.0	1.4	46.5	1.9	-0.1	-5.3	-0.5	-26.3
31311	1.5	37.7	1.6	32.3	1.3	0.2	15.4	0.3	23.1
31313	1.3	37.2	1.5	30.7	1.3	0.0	0.0	0.2	15.4
33213	1.4	35.8	1.5	34.4	1.4	0.0	0.0	0.1	7.1
33215	1.4	36.7	1.5	38.0	1.6	-0.2	-12.5	-0.1	-6.3
33217	1.6	41.6	1.7	35.5	1.5	0.1	6.7	0.2	13.3
33231	1.5	43.0	1.8	39.2	1.6	-0.1	-6.3	0.2	12.5
33233	1.4	35.8	1.5	40.7	1.7	-0.3	-17.6	-0.2	-11.8
33237	1.5	38.0	1.6	34.7	1.4	0.1	7.1	0.2	14.3
41123	1.6	36.2	1.5	40.5	1.7	-0.1	-5.9	-0.2	-11.8
41126	1.7	45.2	1.9	46.2	1.9	-0.2	-10.5	0.0	0.0
42323	1.5	38.9	1.6	37.2	1.5	0.0	0.0	0.1	6.7
42324	1.4	30.5	1.3	36.7	1.5	-0.1	-6.7	-0.2	-13.3
42341	1.4	37.2	1.5	35.8	1.5	-0.1	-6.7	0.0	0.0
42344	1.3	35.8	1.5	32.2	1.3	0.0	0.0	0.2	15.4
51113	1.6	39.9	1.7	26.4	1.1	0.5	45.5	0.6	54.5
51121	1.6	38.1	1.6	42.1	1.8	-0.2	-11.1	-0.2	-11.1
51123	1.8	39.2	1.6	39.6	1.6	0.2	12.5	0.0	0.0
51132	1.9	48.8	2.0	46.6	1.9	0.0	0.0	0.1	5.3
51134	1.5	40.3	1.7	32.0	1.3	0.2	15.4	0.4	30.8
MEAN	1.6	38.2	1.6	38.5	1.6	0.0	-1.2	0.0	0.5
STDEV	0.2	4.1	0.2	4.7	0.2	0.2	10.9	0.2	14.6

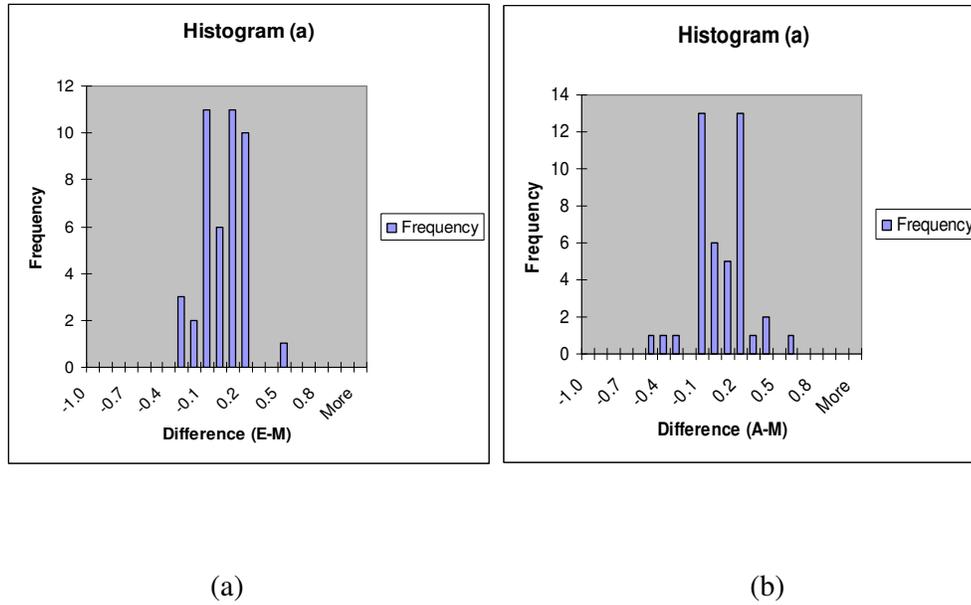


Figure 5.7 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Cubital index a

5.2.4 Cubital index b

The measurement results for honeybee feature “Cubital index b” are given in Table 5.9 and the difference histograms are shown in Figure 5.8. Mean of differences between Expert and Manual (E/M) measurements for feature “Cubital index b” is found to be -0.1 cm and related standard deviation is 0.1 cm. Mean and standard deviation of percentage differences (E/M) are -9.7 and 9.1 respectively. Mean and standard deviation of differences between Automatic and Manual

(A/M) are 0.0 and 0.1 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 3.5 and 14.1.

Table 5.9 - Measurement Results for Cubital index b

b (Cubital Index)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	0.7	20.2	0.8	17.3	0.7	0.0	0.0	0.1	14.3
11215	0.8	23.2	1.0	21.2	0.9	-0.1	-11.1	0.1	11.1
11223	0.7	24.3	1.0	17.5	0.7	0.0	0.0	0.3	42.9
11224	0.6	16.3	0.7	17.3	0.7	-0.1	-14.3	0.0	0.0
11231	0.6	17.0	0.7	16.5	0.7	-0.1	-14.3	0.0	0.0
11235	0.7	20.6	0.9	20.1	0.8	-0.1	-12.5	0.1	12.5
11241	0.8	20.0	0.8	20.6	0.9	-0.1	-11.1	-0.1	-11.1
11244	0.9	23.0	1.0	23.1	1.0	-0.1	-10.0	0.0	0.0
11252	0.6	18.4	0.8	17.7	0.7	-0.1	-14.3	0.1	14.3
11253	0.8	19.1	0.8	18.2	0.8	0.0	0.0	0.0	0.0
12113	0.7	23.0	1.0	17.1	0.7	0.0	0.0	0.3	42.9
12115	0.7	22.1	0.9	18.7	0.8	-0.1	-12.5	0.1	12.5
12133	0.7	18.2	0.8	23.0	1.0	-0.3	-30.0	-0.2	-20.0
12137	0.7	19.1	0.8	22.0	0.9	-0.2	-22.2	-0.1	-11.1
12313	0.8	18.2	0.8	19.6	0.8	0.0	0.0	0.0	0.0
12314	0.7	19.4	0.8	23.2	1.0	-0.3	-30.0	-0.2	-20.0
12354	0.9	20.0	0.8	23.3	1.0	-0.1	-10.0	-0.2	-20.0
12356	1.0	29.0	1.2	23.2	1.0	0.0	0.0	0.2	20.0
22212	0.9	23.1	1.0	22.2	0.9	0.0	0.0	0.1	11.1
22242	0.8	21.0	0.9	20.0	0.8	0.0	0.0	0.1	12.5
22245	0.8	23.0	1.0	24.5	1.0	-0.2	-20.0	0.0	0.0
22255	0.6	18.0	0.8	19.0	0.8	-0.2	-25.0	0.0	0.0
22256	0.7	20.4	0.8	21.2	0.9	-0.2	-22.2	-0.1	-11.1
31111	0.7	20.1	0.8	18.2	0.8	-0.1	-12.5	0.0	0.0

Table 5.9 (Continued)

31113	0.8	21.1	0.9	21.6	0.9	-0.1	-11.1	0.0	0.0
31311	0.8	18.7	0.8	18.2	0.8	0.0	0.0	0.0	0.0
31313	0.8	20.4	0.8	21.2	0.9	-0.1	-11.1	-0.1	-11.1
33213	0.8	22.2	0.9	23.1	1.0	-0.2	-20.0	-0.1	-10.0
33215	0.7	18.1	0.8	18.2	0.8	-0.1	-12.5	0.0	0.0
33217	0.8	20.6	0.9	24.7	1.0	-0.2	-20.0	-0.1	-10.0
33231	0.8	18.4	0.8	20.2	0.8	0.0	0.0	0.0	0.0
33233	0.8	20.2	0.8	20.2	0.8	0.0	0.0	0.0	0.0
33237	0.9	23.2	1.0	24.0	1.0	-0.1	-10.0	0.0	0.0
41123	0.7	19.1	0.8	17.0	0.7	0.0	0.0	0.1	14.3
41126	0.7	20.6	0.9	16.1	0.7	0.0	0.0	0.2	28.6
42323	0.6	20.9	0.9	17.0	0.7	-0.1	-14.3	0.2	28.6
42324	0.8	21.2	0.9	22.2	0.9	-0.1	-11.1	0.0	0.0
42341	0.7	19.4	0.8	17.0	0.7	0.0	0.0	0.1	14.3
42344	0.9	24.5	1.0	23.0	1.0	-0.1	-10.0	0.0	0.0
51113	0.8	20.0	0.8	20.0	0.8	0.0	0.0	0.0	0.0
51121	0.8	20.2	0.8	18.2	0.8	0.0	0.0	0.0	0.0
51123	0.7	22.0	0.9	22.4	0.9	-0.2	-22.2	0.0	0.0
51132	0.7	19.1	0.8	18.1	0.8	-0.1	-12.5	0.0	0.0
51134	0.8	20.2	0.8	19.4	0.8	0.0	0.0	0.0	0.0
MEAN	0.8	20.6	0.9	20.2	0.8	-0.1	-9.7	0.0	3.5
STDEV	0.1	2.3	0.1	2.5	0.1	0.1	9.1	0.1	14.1

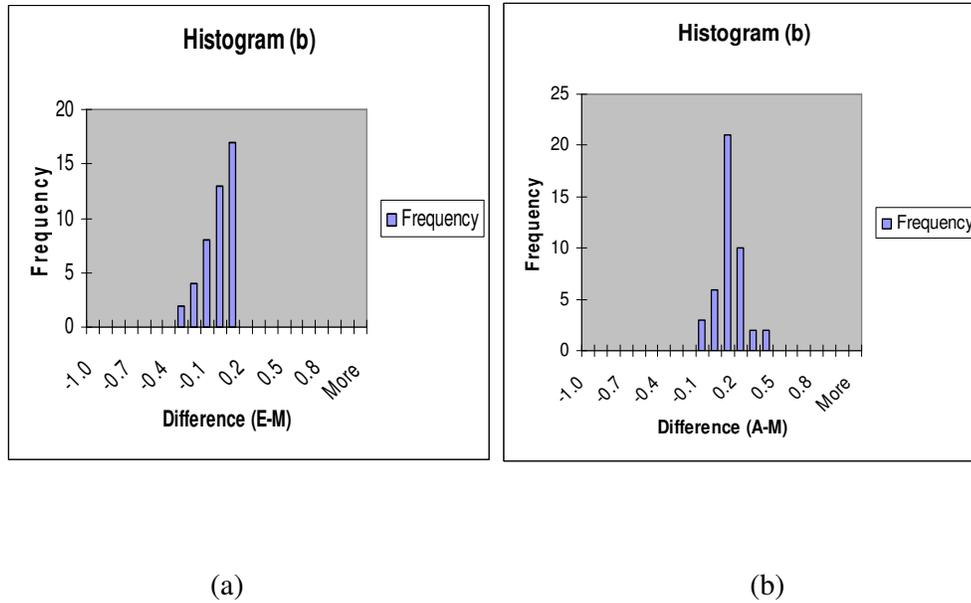


Figure 5.8 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Cubital index b

5.2.5 Distance c

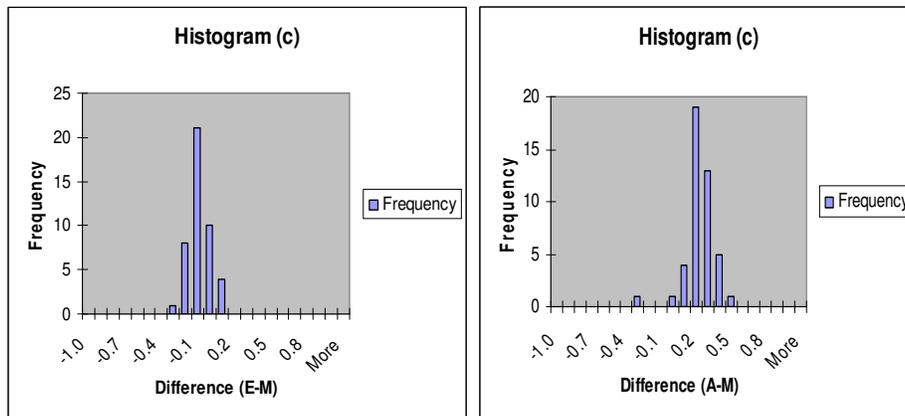
The measurement results for “Distance c” are given in Table 5.10 and the difference histograms are shown in Figure 5.9. Mean of differences between Expert and Manual (E/M) measurements for “Distance c” is -0.1 cm and standard deviation is 0.1 cm. Mean and standard deviation of percentage differences (E/M) are -5.2 and 2.9 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.2 and 0.1 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 6.2 and 5.1.

Table 5.10 - Measurement Results for Distance c

c (Cubital Index)									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	%Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	2.7	71.7	3.0	67.0	2.8	-0.1	-3.6	0.2	7.1
11215	2.6	71.6	3.0	67.1	2.8	-0.2	-7.1	0.2	7.1
11223	2.7	71.2	3.0	66.0	2.7	0.0	0.0	0.3	11.1
11224	2.7	58.8	2.4	67.0	2.8	-0.1	-3.6	-0.4	-14.3
11231	2.7	71.2	3.0	67.0	2.8	-0.1	-3.6	0.2	7.1
11235	2.6	71.5	3.0	67.0	2.8	-0.2	-7.1	0.2	7.1
11241	2.7	71.3	3.0	67.0	2.8	-0.1	-3.6	0.2	7.1
11244	2.7	70.8	2.9	67.1	2.8	-0.1	-3.6	0.1	3.6
11252	2.6	68.9	2.9	65.0	2.7	-0.1	-3.7	0.2	7.4
11253	2.7	72.8	3.0	69.0	2.9	-0.2	-6.9	0.1	3.4
12113	2.7	74.7	3.1	68.4	2.8	-0.1	-3.6	0.3	10.7
12115	2.6	68.6	2.9	66.0	2.7	-0.1	-3.7	0.2	7.4
12133	2.4	65.1	2.7	63.3	2.6	-0.2	-7.7	0.1	3.8
12137	2.7	75.3	3.1	66.3	2.8	-0.1	-3.6	0.3	10.7
12313	2.5	73.7	3.1	64.0	2.7	-0.2	-7.4	0.4	14.8
12314	2.5	67.1	2.8	63.1	2.6	-0.1	-3.8	0.2	7.7
12354	2.5	68.5	2.9	65.1	2.7	-0.2	-7.4	0.2	7.4
12356	2.6	71.2	3.0	68.6	2.9	-0.3	-10.3	0.1	3.4
22212	2.6	75.2	3.1	64.4	2.7	-0.1	-3.7	0.4	14.8
22242	2.7	69.3	2.9	68.0	2.8	-0.1	-3.6	0.1	3.6
22245	2.6	71.3	3.0	68.0	2.8	-0.2	-7.1	0.2	7.1
22255	2.6	67.7	2.8	68.5	2.8	-0.2	-7.1	0.0	0.0
22256	2.6	68.7	2.9	66.6	2.8	-0.2	-7.1	0.1	3.6
31111	2.6	72.0	3.0	65.1	2.7	-0.1	-3.7	0.3	11.1
31113	2.7	73.0	3.0	67.0	2.8	-0.1	-3.6	0.2	7.1
31311	2.7	74.3	3.1	68.1	2.8	-0.1	-3.6	0.3	10.7
31313	2.7	73.4	3.1	69.0	2.9	-0.2	-6.9	0.2	6.9
33213	2.4	69.1	2.9	68.0	2.8	-0.4	-14.3	0.1	3.6
33215	2.5	68.6	2.9	67.0	2.8	-0.3	-10.7	0.1	3.6
33217	2.6	68.6	2.9	65.0	2.7	-0.1	-3.7	0.2	7.4
33231	2.4	64.2	2.7	63.0	2.6	-0.2	-7.7	0.1	3.8
33233	2.6	70.9	3.0	68.5	2.8	-0.2	-7.1	0.2	7.1
33237	2.6	62.8	2.6	62.4	2.6	0.0	0.0	0.0	0.0
41123	2.5	66.7	2.8	66.1	2.7	-0.2	-7.4	0.1	3.7
41126	2.5	67.3	2.8	64.0	2.7	-0.2	-7.4	0.1	3.7

Table 5.10 (Continued)

42323	2.6	65.9	2.7	66.2	2.8	-0.2	-7.1	-0.1	-3.6
42324	2.6	68.9	2.9	64.9	2.7	-0.1	-3.7	0.2	7.4
42341	2.6	70.0	2.9	64.5	2.7	-0.1	-3.7	0.2	7.4
42344	2.8	73.4	3.1	69.4	2.9	-0.1	-3.4	0.2	6.9
51113	2.6	76.5	3.2	64.6	2.7	-0.1	-3.7	0.5	18.5
51121	2.6	69.3	2.9	68.1	2.8	-0.2	-7.1	0.1	3.6
51123	2.6	65.5	2.7	62.1	2.6	0.0	0.0	0.1	3.8
51132	2.5	66.1	2.8	63.0	2.6	-0.1	-3.8	0.2	7.7
51134	2.6	70.0	2.9	63.0	2.6	0.0	0.0	0.3	11.5
MEAN	2.6	69.8	2.9	66.1	2.8	-0.1	-5.2	0.2	6.2
STDEV	0.1	3.6	0.1	2.0	0.1	0.1	2.9	0.1	5.1



(a)

(b)

Figure 5.9 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Distance c

5.2.6 Distance d

The measurement results for honeybee feature “Distance d” are given in Table 5.11 and the difference histograms are shown in Figure 5.10. Mean of differences between Expert and Manual (E/M) measurements for feature “Distance d” is found to be -0.2 cm and related standard deviation is 0.1 cm. Mean and standard deviation of percentage differences (E/M) are -2.8 and 1.9 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0.0 and 0.1 cm. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0.5 and 1.4.

Table 5.11 - Measurement Results for Distance d

Distance d									
Honey bee	Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
11213	5.9	146.1	6.1	143.0	6.0	-0.1	-1.7	0.1	1.7
11215	6.1	149.0	6.2	147.0	6.1	0.0	0.0	0.1	1.6
11223	6.3	156.0	6.5	156.1	6.5	-0.2	-3.1	0.0	0.0
11224	6.1	150.1	6.2	147.1	6.1	0.0	0.0	0.1	1.6
11231	6.0	152.1	6.3	151.0	6.3	-0.3	-4.8	0.0	0.0
11235	6.1	149.1	6.2	149.1	6.2	-0.1	-1.6	0.0	0.0
11241	6.2	153.1	6.4	154.2	6.4	-0.2	-3.1	0.0	0.0
11244	6.1	149.0	6.2	150.1	6.2	-0.1	-1.6	0.0	0.0
11252	6.0	152.1	6.3	147.1	6.1	-0.1	-1.6	0.2	3.3
11253	6.1	152.0	6.3	150.0	6.2	-0.1	-1.6	0.1	1.6
12113	6.3	154.1	6.4	154.6	6.4	-0.1	-1.6	0.0	0.0
12115	6.0	146.1	6.1	146.1	6.1	-0.1	-1.6	0.0	0.0
12133	5.6	144.0	6.0	147.3	6.1	-0.5	-8.2	-0.1	-1.6

Table 5.11 (Continued)

12137	6.0	149.1	6.2	148.2	6.2	-0.2	-3.2	0.0	0.0
12313	6.1	152.0	6.3	152.0	6.3	-0.2	-3.2	0.0	0.0
12314	5.8	147.1	6.1	146.0	6.1	-0.3	-4.9	0.0	0.0
12354	6.2	152.1	6.3	152.0	6.3	-0.1	-1.6	0.0	0.0
12356	6.0	150.0	6.2	148.0	6.2	-0.2	-3.2	0.0	0.0
22212	5.9	148.0	6.2	148.5	6.2	-0.3	-4.8	0.0	0.0
22242	6.1	152.1	6.3	153.0	6.4	-0.3	-4.7	-0.1	-1.6
22245	6.0	144.0	6.0	148.3	6.2	-0.2	-3.2	-0.2	-3.2
22255	6.2	152.0	6.3	149.6	6.2	0.0	0.0	0.1	1.6
22256	6.2	150.0	6.2	147.1	6.1	0.1	1.6	0.1	1.6
31111	5.7	140.0	5.8	139.1	5.8	-0.1	-1.7	0.0	0.0
31113	5.8	143.1	6.0	142.1	5.9	-0.1	-1.7	0.1	1.7
31311	6.1	149.1	6.2	147.0	6.1	0.0	0.0	0.1	1.6
31313	6.2	152.1	6.3	151.0	6.3	-0.1	-1.6	0.0	0.0
33213	5.7	145.0	6.0	144.2	6.0	-0.3	-5.0	0.0	0.0
33215	5.8	148.0	6.2	146.0	6.1	-0.3	-4.9	0.1	1.6
33217	5.9	145.0	6.0	146.5	6.1	-0.2	-3.3	-0.1	-1.6
33231	5.8	144.0	6.0	147.0	6.1	-0.3	-4.9	-0.1	-1.6
33233	5.9	151.0	6.3	149.7	6.2	-0.3	-4.8	0.1	1.6
33237	5.6	141.1	5.9	140.9	5.9	-0.3	-5.1	0.0	0.0
41123	5.8	146.0	6.1	148.1	6.2	-0.4	-6.5	-0.1	-1.6
41126	6.0	148.1	6.2	147.0	6.1	-0.1	-1.6	0.1	1.6
42323	5.6	138.1	5.7	137.1	5.7	-0.1	-1.8	0.0	0.0
42324	5.8	142.0	5.9	141.4	5.9	-0.1	-1.7	0.0	0.0
42341	5.7	144.0	6.0	141.6	5.9	-0.2	-3.4	0.1	1.7
42344	5.9	149.1	6.2	147.8	6.1	-0.2	-3.3	0.1	1.6
51113	5.9	149.0	6.2	146.3	6.1	-0.2	-3.3	0.1	1.6
51121	6.1	152.1	6.3	152.0	6.3	-0.2	-3.2	0.0	0.0
51123	5.7	142.0	5.9	139.7	5.8	-0.1	-1.7	0.1	1.7
51132	5.7	144.1	6.0	140.0	5.8	-0.1	-1.7	0.2	3.4
51134	5.7	144.0	6.0	140.6	5.9	-0.2	-3.4	0.1	1.7
MEAN	5.9	147.9	6.2	147.1	6.1	-0.2	-2.8	0.0	0.5
STDEV	0.2	4.1	0.2	4.4	0.2	0.1	1.9	0.1	1.4

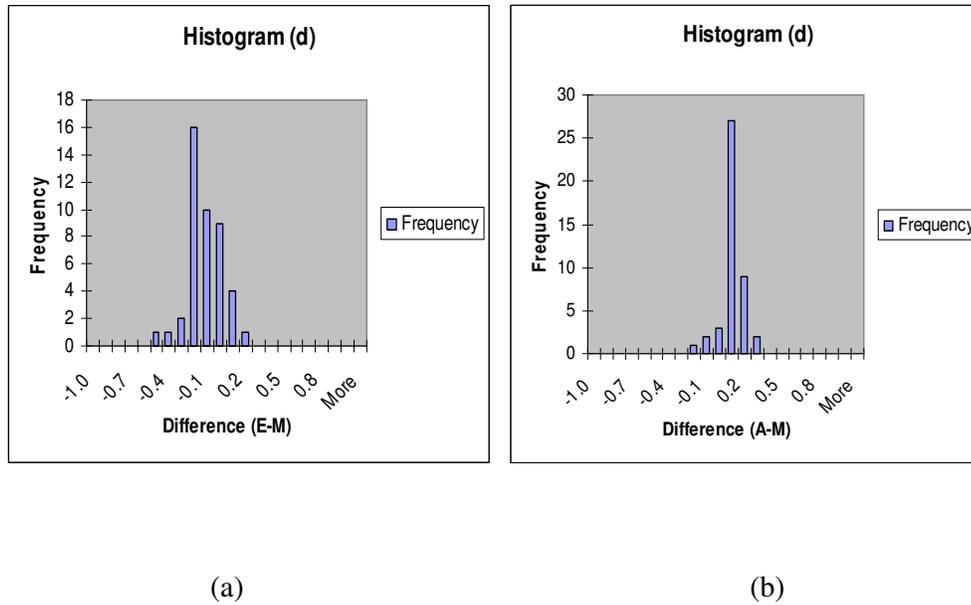


Figure 5.10 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Distance d

5.2.7 Angle EAB (A4)

The measurement results for honeybee feature Angle EAB are given in Table 5.12 and the difference histograms are shown in Figure 5.11. Mean of differences between Expert and Manual (E/M) measurements for feature Angle EAB is found to be 0 degrees and related standard deviation is 2 degrees. Mean and standard deviation of percentage differences (E/M) are 1 and 6 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

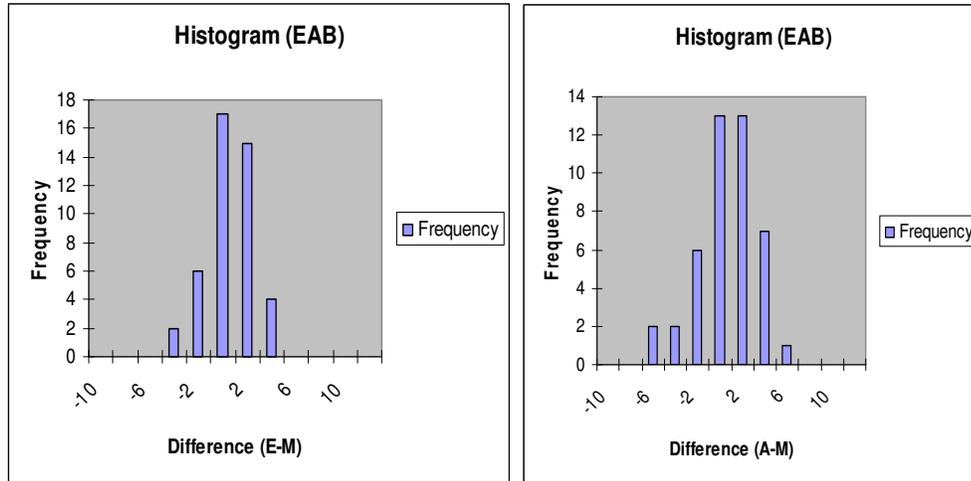
3 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0 and 8.

Table 5.12 - Measurement Results for Angle EAB (A4)

EAB Angle (A4)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	36	36	35	1	3	1	3
11215	36	34	32	4	13	2	6
11223	32	29	29	3	10	0	0
11224	35	38	34	1	3	4	12
11231	34	32	31	3	10	1	3
11235	31	31	29	2	7	2	7
11241	35	34	35	0	0	-1	-3
11244	29	27	29	0	0	-2	-7
11252	34	36	33	1	3	3	9
11253	31	34	31	0	0	3	10
12113	30	30	29	1	3	1	3
12115	34	31	32	2	6	-1	-3
12133	35	32	33	2	6	-1	-3
12137	34	36	36	-2	-6	0	0
12313	36	40	36	0	0	4	11
12314	33	34	33	0	0	1	3
12354	32	35	33	-1	-3	2	6
12356	32	34	30	2	7	4	13
22212	35	33	40	-5	-13	-7	-18
22242	32	31	32	0	0	-1	-3
22245	35	35	36	-1	-3	-1	-3
22255	36	34	35	1	3	-1	-3
22256	33	38	37	-4	-11	1	3
31111	29	28	29	0	0	-1	-3

Table 5.12 (Continued)

31113	30	31	28	2	7	3	11
31311	34	36	34	0	0	2	6
31313	34	29	34	0	0	-5	-15
33213	34	35	37	-3	-8	-2	-5
33215	37	31	33	4	12	-2	-6
33217	30	32	30	0	0	2	7
33231	31	31	30	1	3	1	3
33233	29	32	31	-2	-6	1	3
33237	31	32	33	-2	-6	-1	-3
41123	35	32	35	0	0	-3	-9
41126	33	34	31	2	6	3	10
42323	29	35	29	0	0	6	21
42324	33	32	33	0	0	-1	-3
42341	30	31	32	-2	-6	-1	-3
42344	33	28	31	2	6	-3	-10
51113	37	31	37	0	0	-6	-16
51121	35	34	33	2	6	1	3
51123	33	35	35	-2	-6	0	0
51132	35	31	34	1	3	-3	-9
51134	31	28	32	-1	-3	-4	-13
MEAN	33	33	33	0	1	0	0
STDEV	2	3	3	2	6	3	8



(a)

(b)

Figure 5.11 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle EAB (A4)

5.2.8 Angle EBA (B4)

The measurement results for honeybee feature Angle EBA are given in Table 5.13 and the difference histograms are shown in Figure 5.12. Mean of differences between Expert and Manual (E/M) measurements for feature Angle EBA is found to be -1 degrees and related standard deviation is 6 degrees. Mean and standard deviation of percentage differences (E/M) are -1 and 6 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

4 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0 and 4.

Table 5.13 - Measurement Results for Angle EBA (B4)

EBA Angle (B4)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	92	95	100	-8	-8	-5	-5
11215	102	103	105	-3	-3	-2	-2
11223	102	105	109	-7	-6	-4	-4
11224	90	89	94	-4	-4	-5	-5
11231	98	104	107	-9	-8	-3	-3
11235	105	106	110	-5	-5	-4	-4
11241	94	103	97	-3	-3	6	6
11244	104	111	111	-7	-6	0	0
11252	100	99	102	-2	-2	-3	-3
11253	96	105	101	-5	-5	4	4
12113	99	97	99	0	0	-2	-2
12115	95	94	92	3	3	2	2
12133	96	90	90	6	7	0	0
12137	100	102	97	3	3	5	5
12313	86	85	91	-5	-5	-6	-7
12314	98	96	101	-3	-3	-5	-5
12354	98	100	96	2	2	4	4
12356	99	101	100	-1	-1	1	1
22212	97	97	93	4	4	4	4
22242	103	98	99	4	4	-1	-1
22245	95	96	103	-8	-8	-7	-7
22255	85	84	86	-1	-1	-2	-2
22256	88	90	88	0	0	2	2
31111	99	107	100	-1	-1	7	7

Table 5.13 (Continued)

31113	97	109	103	-6	-6	6	6
31311	103	98	106	-3	-3	-8	-8
31313	101	115	109	-8	-7	6	6
33213	103	96	90	13	14	6	7
33215	96	109	101	-5	-5	8	8
33217	101	99	100	1	1	-1	-1
33231	96	102	98	-2	-2	4	4
33233	105	108	108	-3	-3	0	0
33237	99	101	105	-6	-6	-4	-4
41123	95	102	97	-2	-2	5	5
41126	100	88	90	10	11	-2	-2
42323	108	93	98	10	10	-5	-5
42324	97	112	108	-11	-10	4	4
42341	106	109	108	-2	-2	1	1
42344	102	109	108	-6	-6	1	1
51113	105	100	93	12	13	7	8
51121	99	101	100	-1	-1	1	1
51123	96	90	88	8	9	2	2
51132	84	83	85	-1	-1	-2	-2
51134	98	101	107	-9	-8	-6	-6
MEAN	98	100	99	-1	-1	0	0
STDEV	5	8	7	6	6	4	4

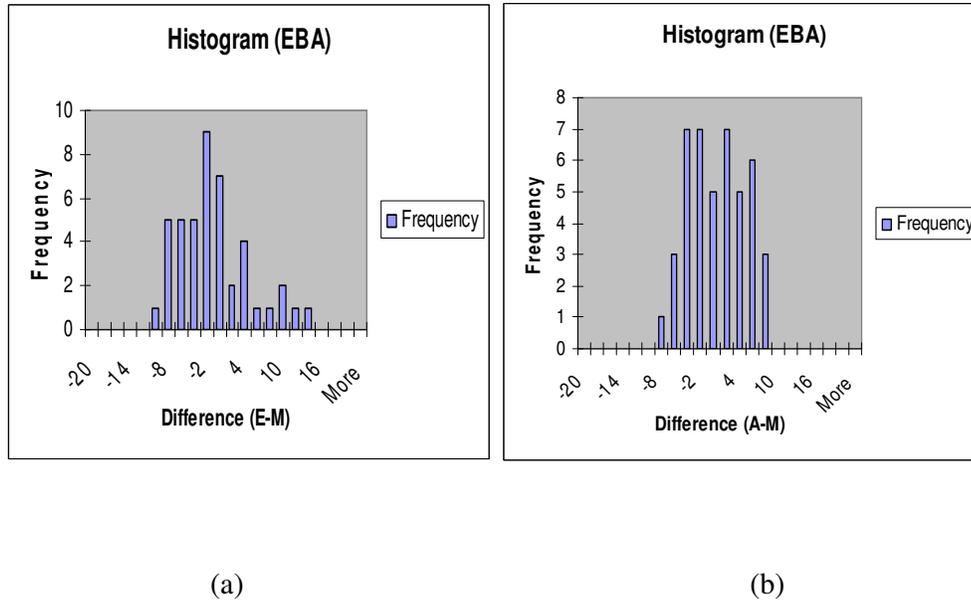


Figure 5.12 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle EBA (B4)

5.2.9 Angle BDG (D7)

The measurement results for honeybee feature Angle BDG are given in Table 5.14 and the difference histograms are shown in Figure 5.13. Mean of differences between Expert and Manual (E/M) measurements for feature Angle BDG is found to be 3 degrees and related standard deviation is 3 degrees. Mean and standard deviation of percentage differences (E/M) are 3 and 3 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 2 and

3 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 2 and 3.

Table 5.14 - Measurement Results for Angle BDG (D7)

BDG Angle (D7)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	101	99	101	0	0	-2	-2
11215	106	107	106	0	0	1	1
11223	104	102	104	0	0	-2	-2
11224	101	98	95	6	6	3	3
11231	104	105	103	1	1	2	2
11235	106	107	107	-1	-1	0	0
11241	108	109	107	1	1	2	2
11244	108	106	104	4	4	2	2
11252	100	102	98	2	2	4	4
11253	97	102	94	3	3	8	9
12113	97	97	96	1	1	1	1
12115	105	101	101	4	4	0	0
12133	101	102	101	0	0	1	1
12137	104	103	101	3	3	2	2
12313	102	105	100	2	2	5	5
12314	105	104	101	4	4	3	3
12354	102	104	103	-1	-1	1	1
12356	105	105	101	4	4	4	4
22212	106	108	103	3	3	5	5
22242	101	97	94	7	7	3	3
22245	104	103	102	2	2	1	1
22255	103	97	98	5	5	-1	-1
22256	101	100	99	2	2	1	1
31111	103	106	100	3	3	6	6

Table 5.14 (Continued)

31113	100	109	99	1	1	10	10
31311	107	106	107	0	0	-1	-1
31313	110	110	110	0	0	0	0
33213	101	98	100	1	1	-2	-2
33215	104	101	98	6	6	3	3
33217	106	105	100	6	6	5	5
33231	103	103	104	-1	-1	-1	-1
33233	103	104	103	0	0	1	1
33237	104	101	103	1	1	-2	-2
41123	101	105	97	4	4	8	8
41126	102	100	101	1	1	-1	-1
42323	104	100	102	2	2	-2	-2
42324	106	111	103	3	3	8	8
42341	106	104	103	3	3	1	1
42344	107	104	106	1	1	-2	-2
51113	116	103	102	14	14	1	1
51121	103	101	100	3	3	1	1
51123	102	97	95	7	7	2	2
51132	96	93	92	4	4	1	1
51134	106	106	106	0	0	0	0
MEAN	104	103	101	3	3	2	2
STDEV	4	4	4	3	3	3	3

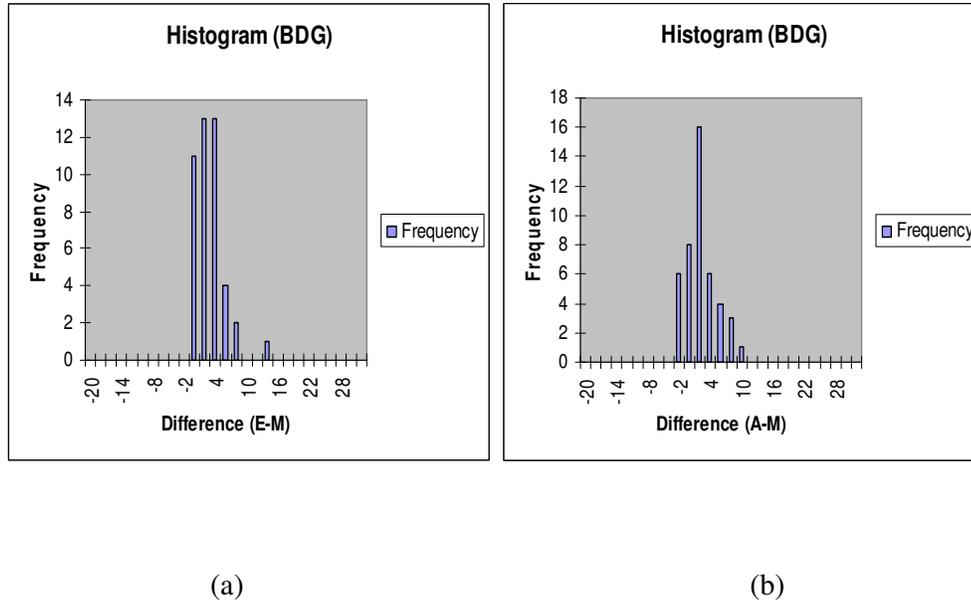


Figure 5.13 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle BDG (D7)

5.2.10 Angle FGD (G18)

The measurement results for honeybee feature Angle BDG are given in Table 5.15 and the difference histograms are shown in Figure 5.14. Mean of differences between Expert and Manual (E/M) measurements for feature Angle BDG is found to be 0 degrees and related standard deviation is 2 degrees. Mean and standard deviation of percentage differences (E/M) are 1 and 6 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

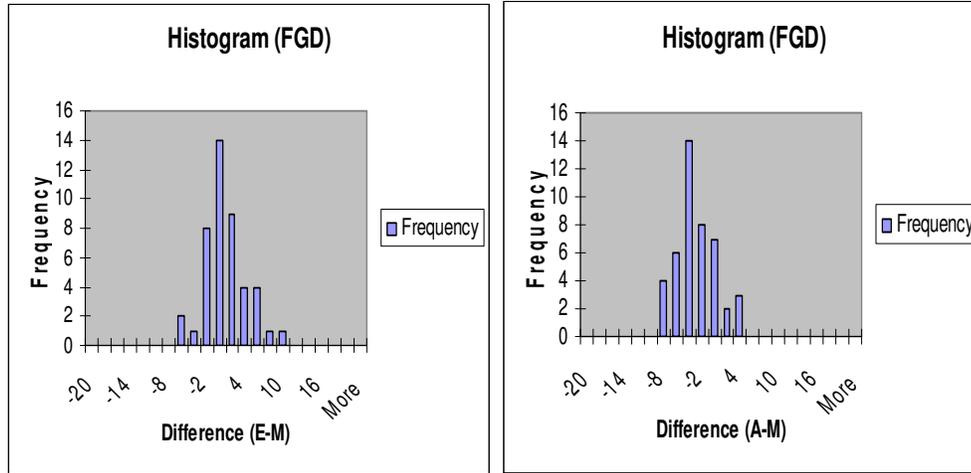
3 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0 and 8.

Table 5.15 - Measurement Results for Angle FGD (G18)

FGD Angle (G18)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	64	65	67	-3	-4	-2	-3
11215	64	63	64	0	0	-1	-2
11223	67	67	67	0	0	0	0
11224	63	59	66	-3	-5	-7	-11
11231	72	64	69	3	4	-5	-7
11235	71	67	72	-1	-1	-5	-7
11241	64	65	66	-2	-3	-1	-2
11244	66	62	64	2	3	-2	-3
11252	67	63	69	-2	-3	-6	-9
11253	65	63	67	-2	-3	-4	-6
12113	63	54	63	0	0	-9	-14
12115	66	67	64	2	3	3	5
12133	70	63	64	6	9	-1	-2
12137	67	64	67	0	0	-3	-4
12313	61	59	68	-7	-10	-9	-13
12314	65	60	67	-2	-3	-7	-10
12354	67	63	68	-1	-1	-5	-7
12356	70	63	70	0	0	-7	-10
22212	69	65	70	-1	-1	-5	-7
22242	71	67	71	0	0	-4	-6
22245	69	59	68	1	1	-9	-13
22255	64	58	66	-2	-3	-8	-12
22256	68	62	66	2	3	-4	-6
31111	64	58	64	0	0	-6	-9

Table 5.15 (Continued)

31113	70	67	69	1	1	-2	-3
31311	70	67	73	-3	-4	-6	-8
31313	63	65	62	1	2	3	5
33213	70	58	62	8	13	-4	-6
33215	65	61	65	0	0	-4	-6
33217	71	60	62	9	15	-2	-3
33231	71	71	76	-5	-7	-5	-7
33233	66	65	65	1	2	0	0
33237	69	64	64	5	8	0	0
41123	59	62	65	-6	-9	-3	-5
41126	62	59	63	-1	-2	-4	-6
42323	67	66	65	2	3	1	2
42324	66	61	63	3	5	-2	-3
42341	65	61	66	-1	-2	-5	-8
42344	66	62	67	-1	-1	-5	-7
51113	69	61	66	3	5	-5	-8
51121	66	63	60	6	10	3	5
51123	63	63	62	1	2	1	2
51132	65	59	61	4	7	-2	-3
51134	66	60	61	5	8	-1	-2
MEAN	67	63	66	1	1	-3	-5
STDEV	3	3	3	3	5	3	5



(a)

(b)

Figure 5.14 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle FGD (G18)

5.2.11 Angle OKF (K19)

The measurement results for honeybee feature Angle OKF are given in Table 5.16 and the difference histograms are shown in Figure 5.15. Mean of differences between Expert and Manual (E/M) measurements for feature Angle OKF is found to be -2 degrees and related standard deviation is 3 degrees. Mean and standard deviation of percentage differences (E/M) are -3 and 4 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

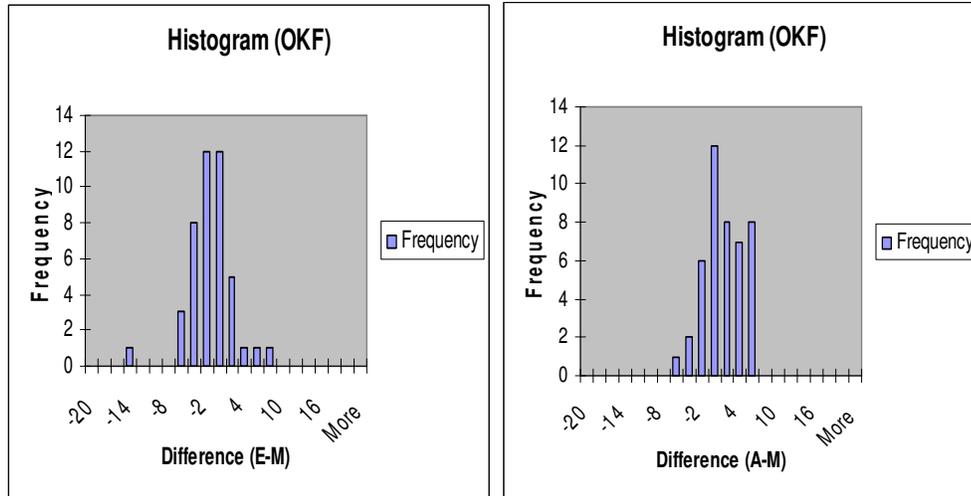
3 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 1 and 4.

Table 5.16 - Measurement Results for Angle OKF (K19)

OKF Angle (K19)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	73	77	74	-1	-1	3	4
11215	70	72	72	-2	-3	0	0
11223	75	79	79	-4	-5	0	0
11224	75	78	80	-5	-6	-2	-2
11231	70	71	72	-2	-2	-1	-1
11235	77	82	77	0	-1	5	6
11241	74	78	75	-1	-1	3	4
11244	73	75	77	-4	-5	-2	-2
11252	76	78	80	-4	-5	-2	-3
11253	73	74	75	-2	-2	0	0
12113	73	79	76	-3	-4	3	4
12115	76	78	74	2	3	4	5
12133	75	79	74	1	2	5	7
12137	74	74	79	-5	-6	-4	-5
12313	75	81	76	-1	-1	5	6
12314	76	80	81	-5	-6	-1	-1
12354	75	82	77	-2	-2	5	6
12356	75	82	78	-3	-4	4	5
22212	71	78	77	-6	-8	1	1
22242	74	77	73	1	1	4	5
22245	74	76	75	-1	-2	1	1
22255	76	81	79	-3	-4	2	3
22256	78	82	82	-4	-4	0	0
31111	77	81	76	1	1	5	6

Table 5.16 (Continued)

31113	70	82	85	-15	-17	-3	-3
31311	73	75	78	-5	-6	-3	-3
31313	76	75	75	1	1	-1	-1
33213	77	79	84	-7	-8	-5	-6
33215	75	78	77	-2	-3	1	2
33217	83	80	78	5	7	3	3
33231	78	80	82	-4	-5	-3	-3
33233	75	75	76	-1	-1	-1	-2
33237	76	76	79	-3	-3	-3	-4
41123	75	78	79	-4	-5	-2	-2
41126	75	78	78	-3	-4	0	0
42323	76	76	83	-7	-8	-7	-8
42324	77	79	77	0	0	2	2
42341	75	77	78	-3	-3	-1	-1
42344	75	76	73	2	2	3	4
51113	77	80	83	-6	-7	-3	-3
51121	74	78	76	-2	-2	2	2
51123	78	80	80	-2	-3	0	0
51132	75	77	76	-1	-2	1	1
51134	74	72	68	6	9	4	7
MEAN	75	78	77	-2	-3	0	1
STDEV	2	3	3	3	4	3	4



(a)

(b)

Figure 5.15 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle OKF (K19)

5.2.12 Angle ROQ (O26)

The measurement results for honeybee feature Angle ROQ are given in Table 5.17 and the difference histograms are shown in Figure 5.16. Mean of differences between Expert and Manual (E/M) measurements for feature Angle ROQ is found to be 0 degrees and related standard deviation is 5 degrees. Mean and standard deviation of percentage differences (E/M) are 0 and 11 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 1 and

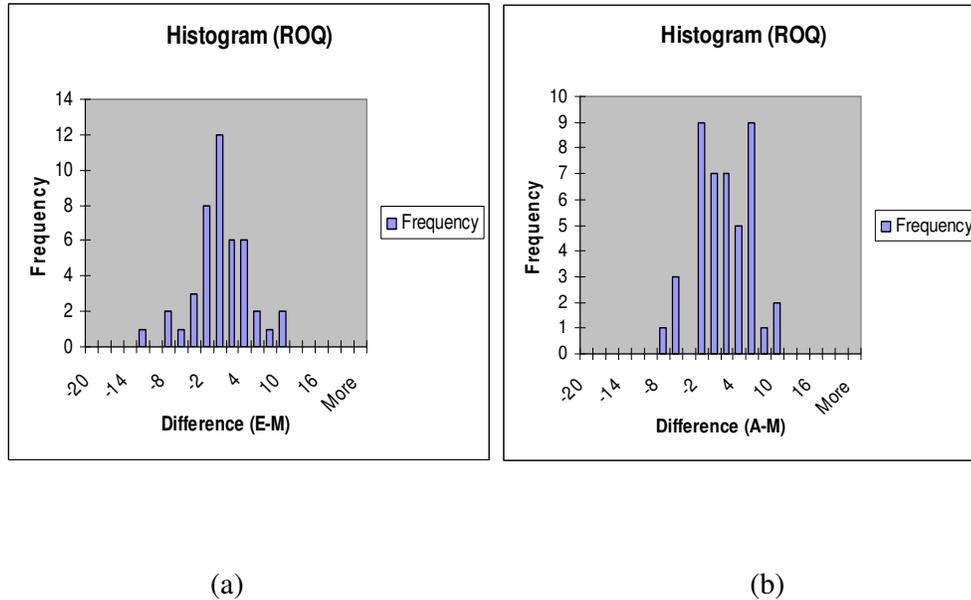
4 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 3 and 11.

Table 5.17 - Measurement Results for Angle ROQ (O26)

ROQ Angle (O26)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	42	42	45	-3	-7	-3	-7
11215	43	41	44	-1	-2	-3	-7
11223	38	43	38	0	0	5	13
11224	34	38	32	2	6	6	19
11231	40	41	42	-2	-5	-1	-2
11235	38	36	39	-1	-3	-3	-8
11241	41	44	43	-2	-5	1	2
11244	36	36	39	-3	-8	-3	-8
11252	49	54	44	5	11	10	23
11253	40	44	38	2	5	6	16
12113	36	39	39	-3	-8	0	0
12115	37	39	40	-3	-8	-1	-3
12133	45	44	35	10	29	9	26
12137	39	41	40	-1	-3	1	3
12313	37	42	37	0	0	5	14
12314	35	39	42	-7	-17	-3	-7
12354	46	52	48	-2	-4	4	8
12356	34	36	39	-5	-13	-3	-8
22212	42	41	40	2	5	1	3
22242	42	47	46	-4	-9	1	2
22245	43	41	43	0	0	-2	-5
22255	42	39	38	4	11	1	3
22256	50	49	46	4	9	3	7
31111	39	42	36	3	8	6	17

Table 5.17 (Continued)

31113	40	44	39	1	3	5	13
31311	40	42	41	-1	-2	1	2
31313	34	39	37	-3	-8	2	5
33213	42	41	37	5	14	4	11
33215	39	40	40	-1	-3	0	0
33217	47	44	37	10	27	7	19
33231	35	36	36	-1	-3	0	0
33233	38	37	34	4	12	3	9
33237	43	49	44	-1	-2	5	11
41123	31	37	43	-12	-28	-6	-14
41126	37	38	34	3	9	4	12
42323	41	40	41	0	0	-1	-2
42324	48	45	40	8	20	5	13
42341	42	40	41	1	2	-1	-2
42344	41	39	42	-1	-2	-3	-7
51113	44	46	40	4	10	6	15
51121	43	40	42	1	2	-2	-5
51123	31	31	40	-9	-23	-9	-23
51132	33	30	37	-4	-11	-7	-19
51134	35	37	44	-9	-20	-7	-16
MEAN	40	41	40	0	0	1	3
STDEV	5	5	3	5	11	4	11



(a) (b)

Figure 5.16 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle ROQ (O26)

5.2.13 Angle HEI (E9)

The measurement results for honeybee feature Angle HEI are given in Table 5.18 and the difference histograms are shown in Figure 5.17. Mean of differences between Expert and Manual (E/M) measurements for feature Angle HEI is found to be -2 degrees and related standard deviation is 7 degrees. Mean and standard deviation of percentage differences (E/M) are -5 and 13 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

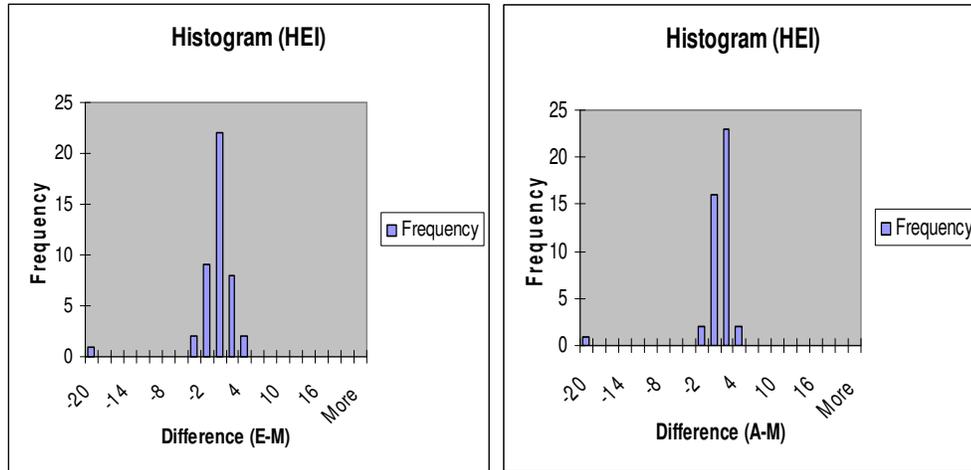
7 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 2 and 13.

Table 5.18 - Measurement Results for Angle HEI (E9)

HEI Angle (E9)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	18	19	18	0	0	1	6
11215	19	20	20	-1	-5	0	0
11223	17	20	21	-4	-19	-1	-5
11224	18	20	21	-3	-14	-1	-5
11231	21	20	20	1	5	0	0
11235	19	20	22	-3	-14	-2	-9
11241	18	18	18	0	0	0	0
11244	18	19	19	-1	-5	0	0
11252	22	22	19	3	16	3	16
11253	18	20	19	-1	-5	1	5
12113	18	20	19	-1	-5	1	5
12115	20	20	66	-46	-70	-46	-70
12133	19	20	19	0	0	1	5
12137	18	21	19	-1	-5	2	11
12313	18	19	20	-2	-10	-1	-5
12314	20	22	21	-1	-5	1	5
12354	18	20	20	-2	-10	0	0
12356	16	18	17	-1	-6	1	6
22212	16	19	15	1	7	4	27
22242	21	19	17	4	24	2	12
22245	17	19	17	0	0	2	12
22255	19	19	18	1	6	1	6
22256	17	20	18	-1	-6	2	11
31111	18	22	23	-5	-22	-1	-4

Table 5.18 (Continued)

31113	20	20	18	2	11	2	11
31311	18	21	21	-3	-14	0	0
31313	19	20	18	1	6	2	11
33213	16	19	18	-2	-11	1	6
33215	18	19	18	0	0	1	6
33217	18	22	21	-3	-14	1	5
33231	18	18	21	-3	-14	-3	-14
33233	18	20	19	-1	-5	1	5
33237	18	20	19	-1	-5	1	5
41123	20	21	21	-1	-5	0	0
41126	17	20	19	-2	-11	1	5
42323	18	18	18	0	0	0	0
42324	18	19	19	-1	-5	0	0
42341	21	22	20	1	5	2	10
42344	19	19	20	-1	-5	-1	-5
51113	19	18	18	1	6	0	0
51121	20	21	21	-1	-5	0	0
51123	16	18	16	0	0	2	13
51132	17	19	18	-1	-6	1	6
51134	19	19	18	1	6	1	6
MEAN	18	20	20	-2	-5	0	2
STDEV	1	1	7	7	13	7	13



(a)

(b)

Figure 5.17 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle HEI (E9)

5.2.14 Angle PNJ (N23)

The measurement results for honeybee feature Angle PNJ are given in Table 5.19 and the difference histograms are shown in Figure 5.18. Mean of differences between Expert and Manual (E/M) measurements for feature Angle PNJ is found to be 4 degrees and related standard deviation is 4 degrees. Mean and standard deviation of percentage differences (E/M) are 4 and 4 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and

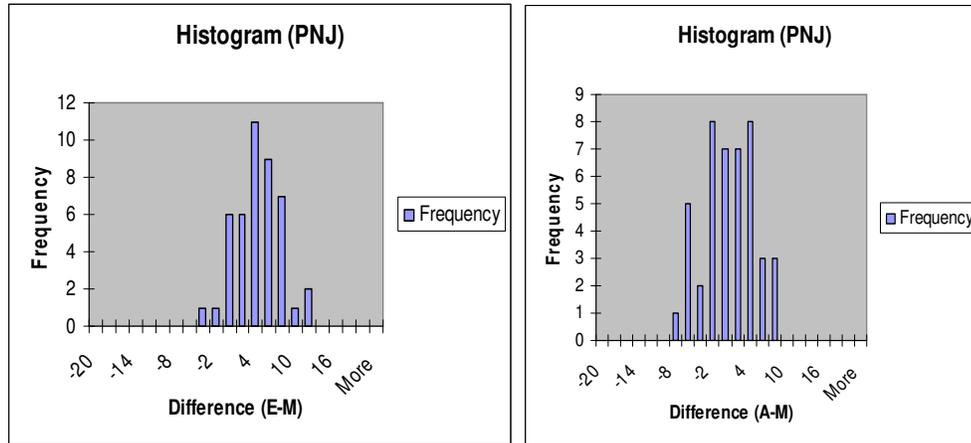
4 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0 and 4.

Table 5.19 - Measurement Results for Angle PNJ (N23)

PNJ Angle (N23)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	90	85	88	2	2	-3	-3
11215	86	86	86	0	0	0	0
11223	90	88	90	0	0	-2	-2
11224	90	80	85	5	6	-5	-5
11231	95	83	90	5	6	-7	-7
11235	97	84	93	4	4	-9	-9
11241	86	84	88	-2	-2	-4	-4
11244	90	85	86	4	5	-1	-1
11252	94	92	89	5	6	3	3
11253	93	87	86	7	8	1	1
12113	90	89	84	6	7	5	5
12115	90	87	89	1	1	-2	-2
12133	90	86	87	3	3	-1	-1
12137	89	93	86	3	3	7	7
12313	86	82	85	1	1	-3	-3
12314	82	83	81	1	1	2	2
12354	90	85	91	-1	-1	-6	-6
12356	92	88	88	4	5	0	0
22212	95	96	92	3	3	4	4
22242	88	83	80	8	10	3	3
22245	93	88	88	5	6	0	0
22255	93	86	88	5	6	-2	-2
22256	86	87	83	3	4	4	4
31111	91	86	85	6	7	1	1

Table 5.19 (Continued)

31113	95	90	88	7	8	2	2
31311	92	88	87	5	6	1	1
31313	90	85	91	-1	-1	-6	-6
33213	92	83	81	11	14	2	2
33215	88	86	89	-1	-1	-3	-3
33217	91	90	89	2	2	1	1
33231	95	89	90	5	6	-1	-1
33233	92	92	89	3	3	3	3
33237	88	84	86	2	2	-2	-2
41123	87	88	83	4	5	5	5
41126	93	86	81	12	15	5	5
42323	90	94	87	3	3	7	7
42324	93	88	84	9	11	4	4
42341	88	84	81	7	9	3	3
42344	90	89	95	-5	-5	-6	-6
51113	94	85	91	3	3	-6	-6
51121	98	93	90	8	9	3	3
51123	94	87	87	7	8	0	0
51132	95	94	87	8	9	7	7
51134	93	92	94	-1	-1	-2	-2
MEAN	91	87	87	4	4	0	0
STDEV	3	4	4	4	4	4	4



(a)

(b)

Figure 5.18 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle PNJ (N23)

5.2.15 Angle NJM (J16)

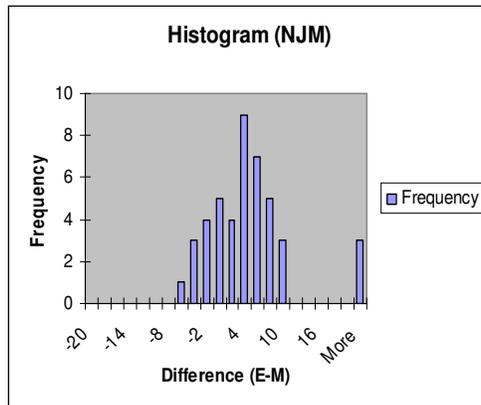
The measurement results for honeybee feature Angle NJM are given in Table 5.20 and the difference histograms are shown in Figure 5.19. Mean of differences between Expert and Manual (E/M) measurements for feature Angle NJM is found to be 5 degrees and related standard deviation is 9 degrees. Mean and standard deviation of percentage differences (E/M) are 7 and 18 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 8 and 10 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 11 and 19.

Table 5.20 - Measurement Results for Angle NJM (J16)

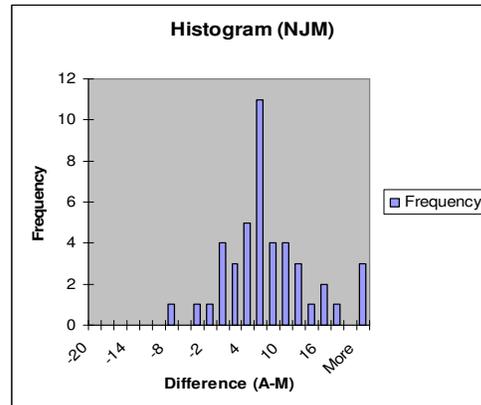
NJM Angle (J16)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	%Diff (E/M)	Diff (A/M)	%Diff (A/M)
11213	90	91	88	2	2	3	3
11215	90	93	93	-3	-3	0	0
11223	89	98	86	3	3	12	14
11224	85	95	83	2	2	12	14
11231	88	90	88	0	0	2	2
11235	86	104	93	-7	-8	11	12
11241	85	88	50	35	70	38	76
11244	88	96	91	-3	-3	5	5
11252	88	92	87	1	1	5	6
11253	90	98	92	-2	-2	6	7
12113	93	91	87	6	7	4	5
12115	96	97	91	5	5	6	7
12133	93	90	93	0	0	-3	-3
12137	101	104	98	3	3	6	6
12313	97	94	98	-1	-1	-4	-4
12314	87	97	80	7	9	17	21
12354	92	96	68	24	35	28	41
12356	95	100	100	-5	-5	0	0
22212	86	99	90	-4	-4	9	10
22242	93	93	86	7	8	7	8
22245	95	95	91	4	4	4	4
22255	91	96	90	1	1	6	7
22256	92	95	89	3	3	6	7
31111	94	96	91	3	3	5	5
31113	100	105	91	9	10	14	15
31311	93	95	87	6	7	8	9
31313	90	95	47	43	91	48	102
33213	97	90	88	9	10	2	2
33215	93	97	97	-4	-4	0	0
33217	92	93	92	0	0	1	1
33231	97	104	94	3	3	10	11
33233	96	99	91	5	5	8	9
33237	97	99	94	3	3	5	5
41123	93	104	88	5	6	16	18

Table 5.20 (Continued)

41126	88	97	81	7	9	16	20
42323	98	102	93	5	5	9	10
42324	99	99	90	9	10	9	10
42341	95	95	88	7	8	7	8
42344	86	86	83	3	4	3	4
51113	96	88	96	0	0	-8	-8
51121	95	97	97	-2	-2	0	0
51123	99	95	91	8	9	4	4
51132	100	100	94	6	6	6	6
51134	95	96	91	4	4	5	5
MEAN	93	96	88	5	7	8	11
STDEV	4	5	10	9	18	10	19



(a)



(b)

Figure 5.19 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle NJM (J16)

5.2.16 Angle IJH (J10)

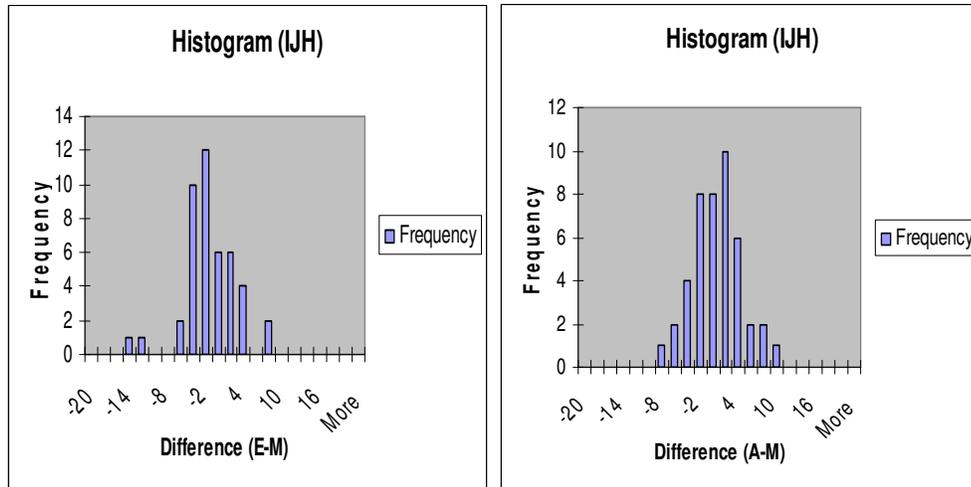
The measurement results for honeybee feature Angle IJH are given in Table 5.21 and the difference histograms are shown in Figure 5.20. Mean of differences between Expert and Manual (E/M) measurements for feature Angle IJH is found to be -2 degrees and related standard deviation is 4 degrees. Mean and standard deviation of percentage differences (E/M) are -3 and 8 respectively. Mean and standard deviation of differences between Automatic and Manual (A/M) are 0 and 4 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 0 and 7.

Table 5.21 - Measurement Results for Angle IJH (J10)

IJH Angle (J10)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	55	56	56	-1	-2	0	0
11215	50	50	52	-2	-4	-2	-4
11223	52	50	55	-3	-5	-5	-9
11224	54	56	53	1	2	3	6
11231	48	53	52	-4	-8	1	2
11235	48	44	49	-1	-2	-5	-10
11241	54	47	47	7	15	0	0
11244	48	44	47	1	2	-3	-6
11252	39	45	53	-14	-26	-8	-15
11253	50	51	49	1	2	2	4
12113	50	55	54	-4	-7	1	2
12115	49	48	54	-5	-9	-6	-11
12133	50	53	54	-4	-7	-1	-2

Table 5.21 (Continued)

12137	49	48	49	0	0	-1	-2
12313	49	45	45	4	9	0	0
12314	44	44	46	-2	-4	-2	-4
12354	52	46	45	7	16	1	2
12356	55	63	54	1	2	9	17
22212	52	50	48	4	8	2	4
22242	48	56	51	-3	-6	5	10
22245	50	55	54	-4	-7	1	2
22255	51	53	54	-3	-6	-1	-2
22256	46	48	50	-4	-8	-2	-4
31111	49	49	50	-1	-2	-1	-2
31113	47	55	51	-4	-8	4	8
31311	52	54	52	0	0	2	4
31313	47	43	43	4	9	0	0
33213	49	56	54	-5	-9	2	4
33215	46	52	59	-13	-22	-7	-12
33217	51	59	55	-4	-7	4	7
33231	44	48	46	-2	-4	2	4
33233	54	55	52	2	4	3	6
33237	48	46	51	-3	-6	-5	-10
41123	49	53	46	3	7	7	15
41126	53	46	51	2	4	-5	-10
42323	47	56	49	-2	-4	7	14
42324	45	47	49	-4	-8	-2	-4
42341	52	59	55	-3	-5	4	7
42344	55	58	61	-6	-10	-3	-5
51113	47	47	49	-2	-4	-2	-4
51121	49	52	55	-6	-11	-3	-5
51123	43	48	45	-2	-4	3	7
51132	46	51	49	-3	-6	2	4
51134	48	53	48	0	0	5	10
MEAN	49	51	51	-2	-3	0	0
STDEV	3	5	4	4	8	4	7



(a)

(b)

Figure 5.20 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle IJH (J10)

5.2.17 Angle ILE (L13)

The measurement results for honeybee feature Angle ILE (L13) are given in Table 5.22 and the difference histograms are shown in Figure 5.21. Mean of differences between Expert and Manual (E/M) measurements for feature Angle ILE is found to be -1 degrees and related standard deviation is 2 degrees. Mean and standard deviation of percentage differences (E/M) are -3 and 12 respectively. Mean and standard deviation of differences between Automatic and Manual

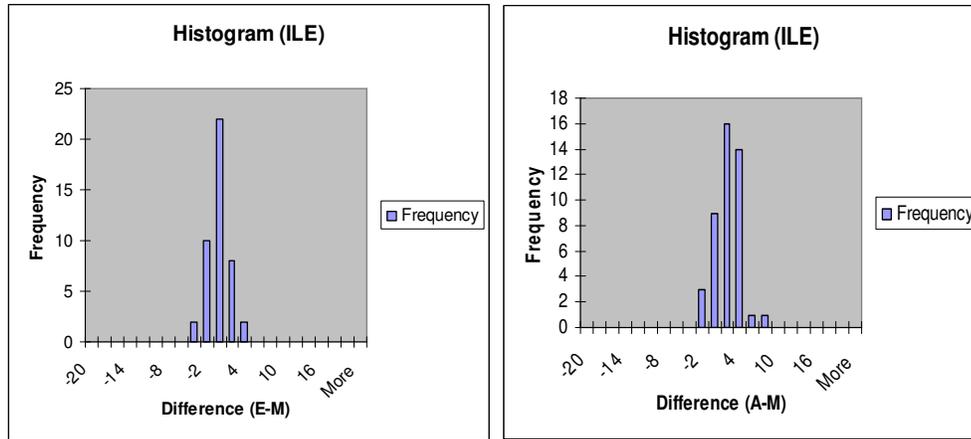
(A/M) are 2 and 2 degrees. Mean and standard deviation of percentage differences between Automatic and Manual (A/M) are 13 and 15.

Table 5.22 - Measurement Results for Angle ILE (L13)

ILE Angle (L13)							
Honey bee	Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
11213	13	17	14	-1	-7	3	21
11215	12	13	13	-1	-8	0	0
11223	13	16	12	1	8	4	33
11224	12	15	13	-1	-8	2	15
11231	12	12	12	0	0	0	0
11235	13	18	14	-1	-7	4	29
11241	13	14	11	2	18	3	27
11244	12	14	14	-2	-14	0	0
11252	12	13	15	-3	-20	-2	-13
11253	13	17	15	-2	-13	2	13
12113	13	17	13	0	0	4	31
12115	13	15	14	-1	-7	1	7
12133	13	18	16	-3	-19	2	13
12137	13	14	14	-1	-7	0	0
12313	14	16	16	-2	-13	0	0
12314	16	18	16	0	0	2	13
12354	14	21	14	0	0	7	50
12356	13	15	13	0	0	2	15
22212	13	16	15	-2	-13	1	7
22242	14	13	13	1	8	0	0
22245	15	17	15	0	0	2	13
22255	12	13	16	-4	-25	-3	-19
22256	14	16	15	-1	-7	1	7
31111	12	20	16	-4	-25	4	25

Table 5.22 (Continued)

31113	13	19	15	-2	-13	4	27
31311	14	18	14	0	0	4	29
31313	12	16	13	-1	-8	3	23
33213	13	15	14	-1	-7	1	7
33215	15	14	12	3	25	2	17
33217	14	15	12	2	17	3	25
33231	14	15	14	0	0	1	7
33233	17	20	14	3	21	6	43
33237	15	18	14	1	7	4	29
41123	12	16	14	-2	-14	2	14
41126	13	19	15	-2	-13	4	27
42323	14	14	13	1	8	1	8
42324	13	11	13	0	0	-2	-15
42341	16	16	16	0	0	0	0
42344	13	16	13	0	0	3	23
51113	13	13	12	1	8	1	8
51121	14	13	14	0	0	-1	-7
51123	14	16	12	2	17	4	33
51132	13	15	14	-1	-7	1	7
51134	13	15	15	-2	-13	0	0
MEAN	13	16	14	-1	-3	2	13
STDEV	1	2	1	2	12	2	15



(a)

(b)

Figure 5.21 - Difference (E-M) Histogram (a) and Difference (A-M) Histogram (b) for Angle ILE (L13)

5.3 Summary of the Difference Mean and Standard Deviation

The difference between the automatic measurement and manual measurements and the difference between the expert measurement and manual measurements are calculated and given in the previous sections 5.1 and 5.2. Mean and standard deviation of each feature is summarized for forewing in Table 5.23 and in Table 5.24, for hindleg in Table 5.25.

Table 5.23 - Mean and Standard Deviation Summary of the Forewing Length Features

		Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
FL	MEAN	28.3	679.4	28.3	679.7	28.3	0.1	0.2	0.0	0.0
	STDEV	0.8	12.7	0.5	12.1	0.5	0.6	2.0	0.2	0.5
FB	MEAN	9.1	220.6	9.2	228.3	9.5	-0.4	-4.3	-0.3	-3.3
	STDEV	0.3	9.7	0.4	8.9	0.4	0.4	4.1	0.2	2.4
a	MEAN	1.6	38.2	1.6	38.5	1.6	0.0	-1.2	0.0	0.5
	STDEV	0.2	4.1	0.2	4.7	0.2	0.2	10.9	0.2	14.6
b	MEAN	0.8	20.6	0.9	20.2	0.8	-0.1	-9.7	0.0	3.5
	STDEV	0.1	2.3	0.1	2.5	0.1	0.1	9.1	0.1	14.1
c	MEAN	2.6	69.8	2.9	66.1	2.8	-0.1	-5.2	0.2	6.2
	STDEV	0.1	3.6	0.1	2.0	0.1	0.1	2.9	0.1	5.1
d	MEAN	5.9	147.9	6.2	147.1	6.1	-0.2	-2.8	0.0	0.5
	STDEV	0.2	4.1	0.2	4.4	0.2	0.1	1.9	0.1	1.4

Table 5.24 - Mean and Standard Deviation Summary of the Forewing Angle Features

		Expert (E) (degree)	Auto. (A) (degree)	Manual (M) (degree)	Diff (E/M)	% Diff (E/M)	Diff (A/M)	% Diff (A/M)
A4 (EAB)	MEAN	33	33	33	0	1	0	0
	STDEV	2	3	3	2	6	3	8
B4(EBA)	MEAN	98	100	99	-1	-1	0	0
	STDEV	5	8	7	6	6	4	4
D7(BDG)	MEAN	104	103	101	3	3	2	2
	STDEV	4	4	4	3	3	3	3
E9(HEI)	MEAN	18	20	20	-2	-5	0	2
	STDEV	1	1	7	7	13	7	13
G18 (FGD)	MEAN	67	63	66	1	1	-3	-5
	STDEV	3	3	3	3	5	3	5
J10 (IJH)	MEAN	49	51	51	-2	-3	0	0
	STDEV	3	5	4	4	8	4	7
J16 (NJM)	MEAN	93	96	88	5	7	8	11
	STDEV	4	5	10	9	18	10	19
K19 (OKF)	MEAN	75	78	77	-2	-3	0	1
	STDEV	2	3	3	3	4	3	4
L13 (ILE)	MEAN	13	16	14	-1	-3	2	13

Table 5.24 (Continued)

	STDEV	1	2	1	2	12	2	15
N23 (PNJ)	MEAN	91	87	87	4	4	0	0
	STDEV	3	4	4	4	4	4	4
O26 (ROQ)	MEAN	40	41	40	0	0	1	3
	STDEV	5	5	3	5	11	4	11

Table 5.25 - Mean and Standard Deviation Summary of the Hindleg Features

		Expert (E) (cm)	Auto (A) (pixels)	Auto (A) (cm)	Manual (M) (pixels)	Manual (M) (cm)	Diff (E/M) (cm)	% Diff (E/M)	Diff (A/M) (cm)	% Diff (A/M)
Fe	MEAN	7.7	190.9	7.9	190.6	7.9	-0.2	-3.0	0.0	0.3
	STDEV	0.2	5.9	0.2	5.4	0.2	0.2	2.3	0.1	1.4
Ti	MEAN	9.9	238.0	9.9	236.8	9.9	0.0	0.3	0.1	0.6
	STDEV	0.4	8.7	0.4	8.4	0.4	0.2	2.4	0.2	1.6
ML	MEAN	6.2	148.2	6.2	149.4	6.2	0.0	-0.2	0.0	-0.8
	STDEV	0.2	7.3	0.3	6.1	0.3	0.2	3.6	0.1	2.1
MW	MEAN	3.5	91.7	3.8	89.7	3.7	-0.2	-6.1	0.1	2.2
	STDEV	0.1	4.2	0.2	3.4	0.1	0.1	3.1	0.2	4.1

5.4 Factors effecting the angle measurements

The angle measurements are very sensitive to the placement of the critical (i.e. vertex) points because some critical points are so close to each other. The factors that effect the angle measurements are explained below:

1. The quality of the images especially for forewings is very important in getting the good results from the measurements carried out with the proposed system. The quality of the forewing image is increased mostly by increasing the luminance of the microscope to increase the lightening of the image and to eliminate the undesired effect of the transparent tapes on the honeybee microscopic slides. In Figure 5.22 a high quality forewing image is shown in which the background is uniform gray color and in Figure 5.24 a low quality, dark forewing image is shown in which the background is not uniform consisting of both white pixels inside the forewing and dark pixels near the thick vein of the forewing. These dark gray level pixels near the thick vein segmented in some of the images during the binarization which cause the wing template partially converges to the desired edges in this region. Figure 5.23 shows the successful convergence of the wing template on the edges of the forewing after edge detection for the high quality image (Figure 5.22). Figure 5.25 shows the convergence of the wing template on the edges of the forewing after edge detection for the low quality image (Figure 5.24). Because of the dark gray level pixels near the thick vein, the unwanted edges are formed and the wing template in this region does not converge well.

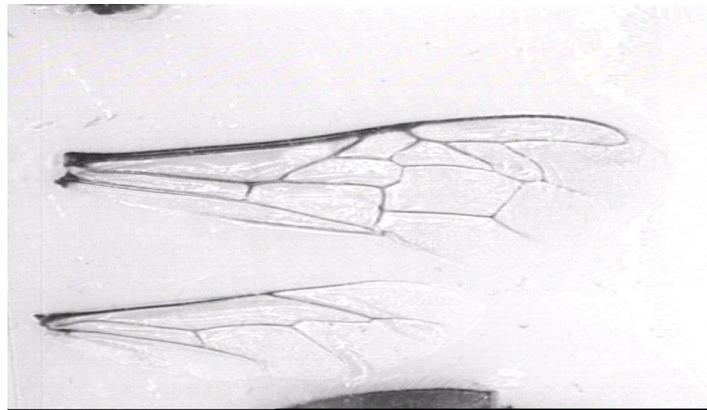


Figure 5.22 - A high quality forewing image

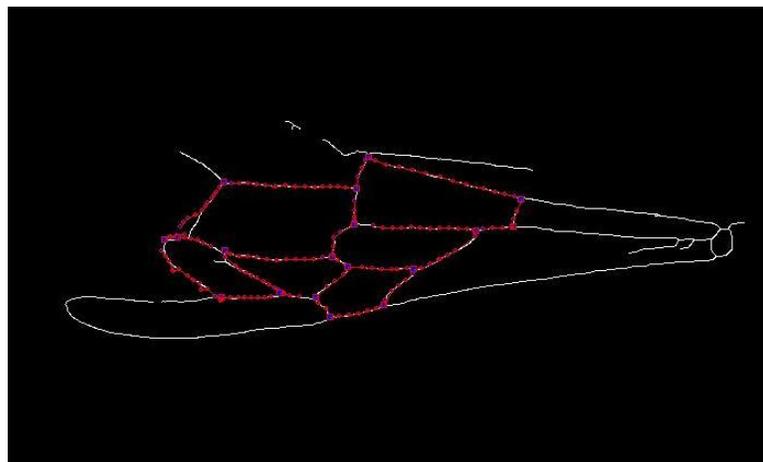


Figure 5.23 - High quality forewing image after applying edge detection and snake algorithms

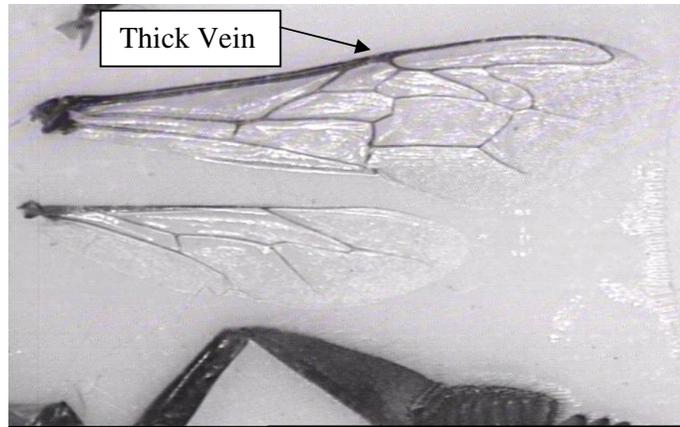


Figure 5.24 - A low quality forewing image

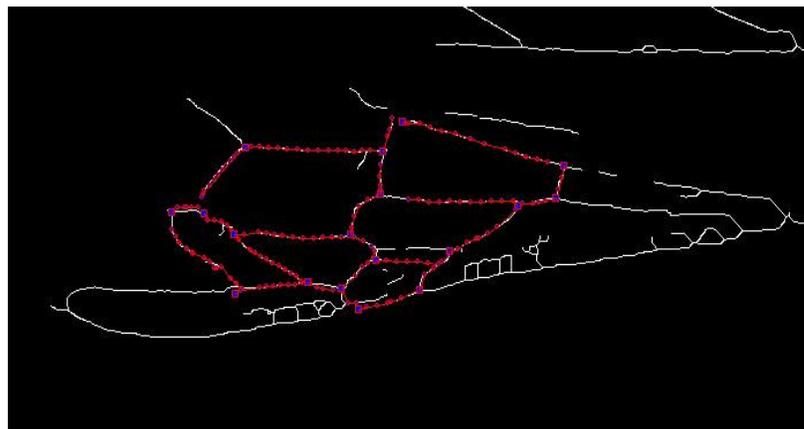


Figure 5.25 - Low quality forewing image after applying edge detection and snake algorithms

Besides the quality of the image, the differences in angles with respect to the expert measurements are emerged due to the factors below:

2. First, the expert measurements cannot be completely reliable because the feature measurements are handled by putting the rulers on the hindleg or forewing image on the monitor and taking the length and angle measurements. This monitor screen is not flat and on the contrary, it is convex that increase the probability of human errors.
3. The distances between critical points are very small. This condition affects especially the angle measurements and causes differences between automatic and expert measurements by up to 5 degrees.
4. Due to the quality of the images, some edges especially the AE, EI, IM, ML, LN, NP edges on contours of 3, 5, 6 (Figure 4.21) can partially segmented or unwanted edges are formed. For that reason, the snake algorithm applied to wing templates converges partially for these edges, which cause differences in the calculations.
5. In very low quality images, even the vertex points that determine the critical points used in the calculations of wing lengths and angles cannot be segmented. The user of the system may correct the disappeared critical point by dragging this point to the most suitable place. However, this may also increase the differences. The best way to measure the features of that kind of critical points is to reload the image and measure the feature manually in the system.
6. The angles that give the big percentage difference are measured again manually in the system and the differences between the expert and manual measurements found to be different up to 4-5 degrees, showing that there can be errors also in the expert measurement.

7. LN edge of the wing is very thick compared to the other edges. After thinning algorithm, the vertex points can shift from actual critical points. The shift can be observed as follows for the forewing image 31111:

- a. The distance between the vertex points L and N is measured to be 42 pixels by using the manual measuring of the system as shown below.

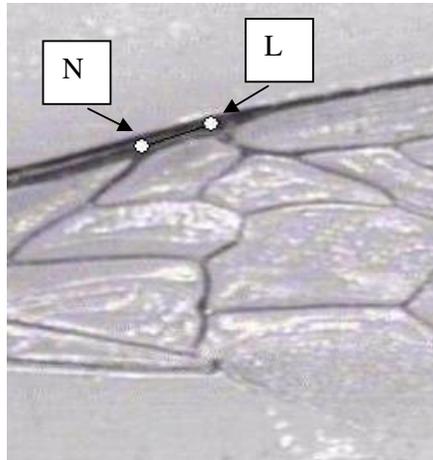


Figure 5.26 - Manual LN distance measurement with L and N points are placed on actual places

- b. By using automatic measurement (i.e. applying binarization and thinning), the LN distance is measured manually to be 58.69 pixels as shown below.

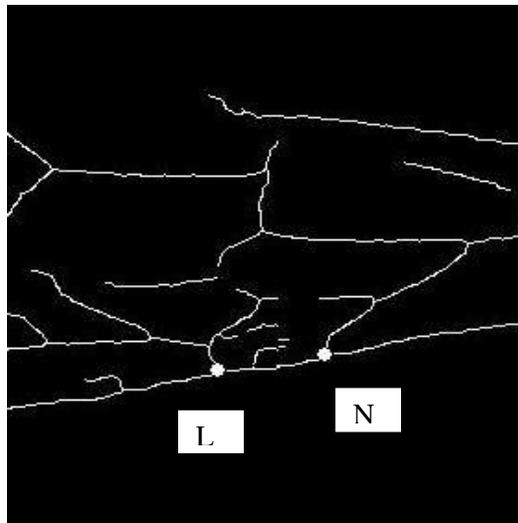


Figure 5.27 - Manual LN distance measurement after applying binarization and thinning

- c. When the L and N vertex points are marked manually as in Figure 5.28, the manual measurement for LN distance is 57.56 pixels which is closer to the value found in case (b).

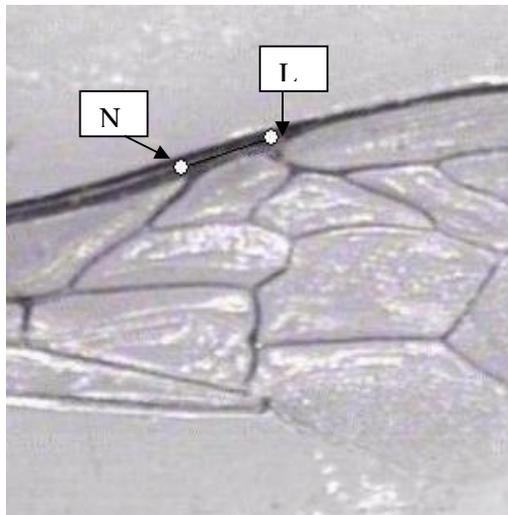


Figure 5.28 - Manual LN distance measurement with L and N points are placed far from the actual places

For that reason L and N vertex points in some images can shift with respect to the actual points that cause some of the ILE angle (L13)

measurements to be wrong. In addition to that, there is a difference between the manual measurement and expert measurement results for ILE angle up to 3-4 degrees. For forewing image 31111, ILE angle manual measurement is 15.86 degrees, expert measurement is 12 degrees, and this difference occurs for that kind of small angles.

8. The placement of the L and N critical points are very important for the angle features that use them. They must be placed in the middle of the thick veins in the direction of ML and NP edges as shown in Figure 2.3. After binarization and thinning of the thick vein NL, the critical points L and N can be placed to near but not to actual points. This situation causes the angle features especially the small angled features that use these points to produce high percentage differences.

CHAPTER 6

CONCLUSIONS

In this thesis, a semi-automatic honeybee forewing and hindleg feature extraction system is proposed. It is semi-automatic because the user is prompted to enter the critical points on the hindleg and forewing images only at the initial phase of image processing operations. These reference points are used in different stages of the system; placing the honeybee images in the same position and orientation with the templates used in snake algorithm are examples to this. The entered reference points speed up the system and make it more accurate. The honeybee feature extraction system has also the manual mode for taking the length and angle measurements.

The honeybee forewing and hindleg images used in the system are captured using the TV card in the setup shown in Figure 2.7. The samples used in measurements are collected from different villages of Ula, Marmaris, Milas, Bodrum and Muğla. A total of 50 hindleg and 50 forewing images are used for the measurements. 4 length features of hindleg, 8 length features and 11 angle features of forewing are measured for testing. As the expert measurements do not contain the L1 and L2 length features of the forewing, these are not used in difference and percentage difference calculations.

The measurements carried out in the system and the measurements made by the expert are compared with respect to the manual measurements by calculating the differences and percentage differences for each feature. The mean and standard deviation of differences, percentage differences and all measurements for each feature are also calculated and summarized in Table 5.23 and in Table 5.24 for forewing and in Table 5.25 for hindleg. As can be seen from the results, the mean difference and standard deviation of length features is less than 0.5 cm and most of them are 0.1 cm. Forewing length features “cubital index a” and “cubital index b” are very small compared to the long length features “FL” and “FW” which results high difference percentage mean for these features (i.e. between %10 - %15). Among the length features, FL, Fe, Ti and ML are extracted using the reference points entered by the user in the initial phase of the program. Moreover, the reference point 2 is used for measuring of FW, which increases the accuracy of the data.

The mean and standard deviation of differences and percentage differences for angle measurements are also in reasonable ranges. The difference mean and percentage difference mean are low for large angles, measurements of which are between 80-100 degrees. In contrast, angles of ILE and HEI are examples to lower valued angles between 13 and 20 degrees, which cause high percentage differences.

The angle measurements are very sensitive to the placement of the critical (i.e. vertex) points. Some critical points used in angle calculations are very close to each other, such that they cause high percentage of difference even after a few pixels of movement.

Finding the vertex points after applying the snake algorithm to the edges of the forewing improves the correctness of the angle measurements greatly. However,

some critical points cannot be determined automatically with this process. Because, the critical points have to be searched in a limited neighborhood or some edges may not be segmented in the binarization step due to the low quality of the images. So, a final correction to the critical points with wrong placements can be applied by user. In this way, the final touch of the user reduces the differences in forewing measurements.

Another alternative method to measure the features of disappearing critical points is to reload the image and use the manual measurement mode of the system. In this mode, the user simply marks proper points on the image which are later used to calculate length or angle features. Even this mode greatly speeds up the measurement process compared to the method of measuring by directly placing rulers on the screen.

The quality of the images -especially for forewings- is very important to obtain fine results from the measurements carried out with the proposed system. Mainly, the lightening and the transparent tapes on the honeybee microscopic slides affect the quality of the forewing images. On the other hand, these factors do not affect the performance of the hindleg images. The quality of the forewing image can be improved by increasing the luminance of the microscope. This enhances the brightness of the image and eliminates the undesired effects of the transparent tapes stick on the honeybee microscopic slides. If the lightening of the microscope is adjusted well, then the background of the forewing image becomes uniform which provides a fine segmentation followed by successful edge detection and the convergence of the wing template in active contour (snake) algorithm.

In some of the low quality forewing images, some edges -especially the AE, EI, IM, ML, LN, NP- on contours of 3, 5, 6 (Figure 4.21) can be segmented partially or unwanted edges may even be formed. Due to partial segmentation and extra

edges, the snake algorithm applied to wing templates converges partially for these edges, which inevitably leads to errors in calculations.

The distances between most of the critical points are very small, which affects especially the angle measurements and causes differences between automatic and expert measurements by up to 5 degrees.

For the angles, which contain high percentage of differences, measurements are repeated by using the manual measurement mode in the system. Even after the repeated manual measurements, it is observed that the differences between the expert and manual measurements may differ up to 4-5 degrees. These results show that, also the expert measurement may contain errors. The expert measurements cannot be completely reliable because the feature measurements are handled by placing the rulers on the hindleg or forewing image on the monitor and taking the length and angle measurements. This monitor screen is convex, which may well increase the probability of human errors.

LN edge of the wing is very thick compared to the other edges. After binarization and thinning algorithm, the vertex points can shift from the actual critical points. However, the placement of the L and N critical points are very important for the angle features that use them. They must be placed in the middle of the thick veins in the direction of ML and NP edges as shown in Figure 2.3. This situation causes the angle features -especially the small angled features that use these points- to produce high percentage differences. High percentage differences of ILE angle arise from that factor.

As a future work, for improving the correctness of angle ILE, user can enter the critical points L and N as reference points in the beginning of the program. This will also decrease the percentage difference in angles PNJ and NJM that use the critical point N.

REFERENCES

- [1] Aykut Kence, Aslı Zeynep Eryümlü, “Determination of Morphometric and Electrophoretic Variation in Honey Bee (*Apis Mellifera* L.) Populations of Aegean Region of Turkey”, METU, January 1999

- [2] Matrox Imaging Library, Version 5.1, Command Reference, Manual No. 10512-MS-0501, February 10, 1998

- [3] Matrox Imaging Library, Version 5.1, User Guide, Manual No. 10513-MS-0501, February 10, 1998

- [4] Efford Nick (2000), “Digital Image Processing, A Practical Introduction Using Java”, Addison Wesley.

- [5] Friedrich M. Wahl (1987), “Digital Image Signal Processing”, Artech House.

- [6] M. Kass, A, Witkin, and D, Terzopoulos, “Snake: Active Contour Models”, International Journal of Computer Vision, vol. 1, no.4, pp321-331, 1988

- [7] A.A. Amini, T. E. Weymouth, and T. C. Jain, “Using Dynamic Programming for Solving Variation Problems in Vision”, IEEE Trans on Pattern Analysis and Machine Intelligence, vol. 12, no. 9, September 1990

- [8] Williams, D. J. and Shah, M. (1992) "A fast algorithm for active contours and curvature estimation". CVGIP: Image Understanding, 55(1), pp14-26.
- [9] Ayşegül Gönülşen, Uğur Halıcı, Kemal Leblebicioğlu, Aykut Kence, "Bal Arıları Kanat ve Bacaklarının Belirli Karakterlerinin İmge İşleme Algoritmaları ile Ölçülmesi", Sinyal İşleme Uygulamaları Kurultayı, 2003.
- [10] David F. Rogers, J. Alan Adams, "Mathematical Elements For Computer Graphics"
- [11] John R.Rankin, "Computer Graphics Software Constructions", Prentice Hall, Australia, 1989
- [12] Ruttner, F. 1988. Biogeography and Taxonomy of Honey Bees. Springer-Verlag, Berlin.
- [13] Gould, J.L. and Gould, C.G. 1988. The Honey Bee. W.H Freeman and Company, N.Y.
- [14] Wilson, E.O. 1971. Insect Societies. Harward University Press, Cambridge, Moss.
- [15] Yılmaz, M.A 1990. Bal Arılarında (*Apis mellifera* L.) protein poliformizmi. Yüksek Lisans Tezi, Ankara Üniversitesi.

- [16] Darendeliođlu, Y. and Kence, A. 1993. Morhometric study on population structure on honey bee, *Apis mellifera* L. (Hymenoptera: Apidae). Türkiye 2. Entomoloji Kongresi Bildirileri, 387-396.
- [17] Kandemir, İ., Kence, A. 1995. Allozyme variability in a central Anatolian Honey bee (*Apis mellifera* L) population. *Apidologie*, 26, 503-510.
- [18] Bumgardner Wendy, 1996; <http://www.ava.org/bees.htm>.
- [19] Ana Britannica. 1993. Ana Yayıncılık A.Ş. ve Encyclopaedia Britannica, Inc.
- [20] Damus, M.S. and Otis G.W. 1997. A morhometric analysis of *Apis cerana* F. and *Apis nigrocicta* Smith populations from Southeast Asia. *Apidologie*, 28, 309-323.
- [21] Bodenheimer, F.S. 1941. Studies on the Honey Bee and Beekeeping in Turkey. Merkez Ziraat Mücadele Enstitüsü, Ankara.
- [22] Maa, T. An injury into the systematics of the tribus *Apidini* of honey bees (Hym.) Treubia, 21, 525-640.
- [23] Adam, B. 1983. In Search of the best Strains of Honey Bees. Second edition. Northern Bee Books, UK.

[24] Karacaođlu, M. 1989. Orta Anadolu, Karadeniz geit ve Ardahan izole blgeleri arılarının bazı morfometrik zellikleri zerine bir arařtırma. Doktora Tezi. Ankara niversitesi Ziraat Fakltesi, Ankara.

[25] Settar, A. 1983. Ege blgesi arı tipleri ve gezginci arıcılık zerine arařtırmalar. Doktora Tezi. Ege Blge Zirai Arařtırma Enstits.

APPENDIX A

MATROX IMAGING LIBRARY

The Matrox Imaging Library (MIL) package is a hardware-independent modular imaging library. It has an extensive set of commands for image processing and specialized operations such as blob analysis, measurement, pattern recognition, and optical feature recognition. It also supports a basic graphics set. In general, MIL can manipulate either grayscale or color images. However, statistical, blob analysis, measurement, pattern matching, and optical feature recognition operations are done on grayscale images only.

It allows for platform-independent applications. This means that a MIL application can run on any VESA-compatible VGA board or Matrox imaging board under different environments (DOS or Windows).

MIL is used in this project in order to decrease the software development cycle. MIL functions are mainly used for;

1. Image Acquisition (e.g. loading Images from disk and storing the processed form of the image in disk.)
2. Statistical information and binarization (e.g. histogram modification, thresholding etc.)

3. Morphological operations (e.g. dilation, erosion, opening, closing, thinning etc.)
4. Filtering operations (e.g. edge detection, smoothing etc.)
5. Blob Analysis (e.g. identifying and measuring blobs such as the blob area, center of gravity of the blobs, eliminating blobs within an image etc.)
6. Graphical operations (e.g. drawing lines on the center of gravity of the blobs etc)

MIL functions that are used in the implementation of this system are given briefly in Table A.6.1 [2] and they will be explained in the rest of this appendix.

Table A.6.1 - Brief definitions of matrox functions used in the application.

Function Name :	Description :
<i>MappAllocDefault()</i>	Allocate MIL application defaults.
<i>MappFreeDefault()</i>	Free MIL application defaults.
<i>MblobAllocFeatureList()</i>	Allocate a blob analysis feature list.
<i>MblobAllocResult()</i>	Allocate a blob analysis result buffer.
<i>MblobCalculate()</i>	Perform blob analysis calculations.
<i>MblobFill()</i>	Fill blobs that meet a given criteria.
<i>MblobFree()</i>	Free the blob analysis result buffer or the feature list.
<i>MblobGetNumber()</i>	Get the number of currently included blobs.
<i>MblobGetResult()</i>	Read feature values of the included blobs.
<i>MblobReconstruct()</i>	Reconstruct blobs in an image.
<i>MblobSelect()</i>	Select blobs for calculations and result retrieval.
<i>MblobSelectFeature()</i>	Select feature(s) to be calculated.
<i>MbufAllocColor()</i>	Allocate a color data buffer.
<i>MbufAlloc2d()</i>	Allocate a 2D data buffer.
<i>MbufChild2d()</i>	Allocate a 2D child data buffer.

Table A6.1 (Continued)

<i>MbufCopy()</i>	Copy data from one buffer to another.
<i>MbufExport()</i>	Export a data buffer to a file.
<i>MbufFree()</i>	Free a data buffer.
<i>MbufGet2d()</i>	Get data from a 2D area of a buffer and place it in a user supplied array
<i>MbufImport()</i>	Import data from a file into a data buffer.
<i>MbufPut2d()</i>	Put data from a user-supplied array into a 2D area of a buffer.
<i>MdispSelectWindow()</i>	Select an image buffer to display in a user-defined window
<i>MgraAlloc()</i>	Allocate a graphics context.
<i>MgraColor()</i>	Associate a foreground color with a graphics context.
<i>MgraFree()</i>	Free a graphics context
<i>MgraLine()</i>	Draw a line.
<i>MimAllocResult()</i>	Allocate an image processing result buffer.
<i>MimBinarize()</i>	Perform a point-to-point binary-thresholding operation.
<i>MimClose()</i>	Perform a closing-type morphological operation.
<i>MimConvert()</i>	Perform a color conversion.
<i>MimConvolve()</i>	Perform a general convolution operation.
<i>MimDilate()</i>	Perform a dilation-type morphological operation.
<i>MimEdgeDetect()</i>	Perform a specific edge detection operation.
<i>MimErode()</i>	Perform an erosion-type morphological operation.
<i>MimFlip()</i>	Perform a horizontal or vertical image flipping.
<i>MimFree()</i>	Free an image processing result buffer.
<i>MimGetResult()</i>	Get values from an image processing result buffer.
<i>MimHistogram()</i>	Generate the intensity histogram of an image buffer.
<i>MimOpen()</i>	Perform an opening-type morphological operation.
<i>MimThick()</i>	Thicken blobs in an image.
<i>MimThin()</i>	Thin blobs in an image.

The description of MIL functions used in this application is given as below [2]:

MappAllocDefault

Format:

void MappAllocDefault(InitFlag, ApplicationIdPtr, SystemIdPtr, DisplayIdPtr,
DigIdPtr, ImageBufIdPtr)

long InitFlag;	Initialization flag
MIL_ID *ApplicationIdPtr;	Storage location for application identifier
MIL_ID *SystemIdPtr;	Storage location for system identifier
MIL_ID *DisplayIdPtr;	Storage location for display identifier
MIL_ID *DigIdPtr;	Storage location for digitizer identifier
MIL_ID *ImageBufIdPtr;	Storage location for image buffer identifier

Description:

This macro sets up the requested MIL and processing environments using the defaults specified in the "milsetup.h" file. It can allocate and initialize a MIL application, allocate the system to receive the MIL commands, allocate the digitizer and display, and allocate and clear a displayable image buffer on this target system, depending on what is requested.

MappFreeDefault

Format:

void MappFreeDefault(ApplicationId, SystemId, DisplayId, DigitizerId,
ImageBufId)

MIL_ID ApplicationId;	Application identifier
MIL_ID SystemId;	System identifier
MIL_ID DisplayId;	Display identifier
MIL_ID DigId;	Digitizer identifier
MIL_ID ImageBufId;	Image buffer identifier

Description:

This macro frees the MIL application defaults that were allocated with the MappAllocDefault macro (located in milsetup.h). Note, this command does not affect what is being displayed on the system's display; if you want to clear the display, you should do so, using MdispDeselect, before calling MappFreeDefault.

MblobAllocFeatureList

Format:

MIL_ID MblobAllocFeatureList(SystemId, FeatureListIdPtr)

MIL_ID SystemId;	System identifier
MIL_ID *FeatureListIdPtr;	Storage location for feature list identifier

Description:

This function allocates a feature list. The feature list holds the feature(s) to be calculated by MblobCalculate(). You must specify which feature(s) to calculate, using MblobSelectFeature(). Immediately after allocation, no features are selected in the feature list. When the feature list is no longer required, release it, using blobFree().

MblobAllocResult

Format:

MIL_ID MblobAllocResult(SystemId, BlobResIdPtr)

MIL_ID SystemId;	System identifier
MIL_ID *BlobResIdPtr;	Storage location for blob analysis result buffer identifier

Description:

This function allocates a result buffer used to store blob analysis results. Each blob creates a separate result entry in the blob analysis result buffer. You can retrieve blob analysis results from a result buffer, using MblobGetResult(). When the result buffer is no longer required, release it, using MblobFree().

MblobCalculate

Format:

void MblobCalculate(BlobIdentImageId, GrayImageId, FeatureListId, BlobResId)

MIL_ID BlobIdentImageId;	Blob identifier image identifier
MIL_ID GrayImageId;	Optional grayscale image identifier

MIL_ID FeatureListId;	Feature list identifier
MIL_ID BlobResId;	Blob analysis result buffer identifier

Description:

This function calculates the features specified in the given feature list for all currently included blobs in the blob identifier image and stores results in the specified result buffer. Features are added to the feature list with MblobSelectFeature().

MblobFill

Format:

void MblobFill(BlobResId, DestImageBufId, Criterion, Value)

MIL_ID BlobResId;	Blob analysis result buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long Criterion;	Fill criterion
long Value;	Fill value

Description:

This function draws, in an image, those blobs that meet a specified criterion. Note, only those that meet the criterion are drawn; other blobs and background pixels are not drawn.

This function is often used to remove unwanted (excluded or deleted) blobs from the identifier image (by filling them with the background color), or to highlight included blobs in a different color. Therefore, an appropriate destination image is the blob identifier image (or a copy of it) associated with the result buffer, or another image buffer that has been cleared.

MblobCalculate() must have been called prior to using this function.

MblobFree

Format:

void MblobFree(BlobId)

MIL_ID BlobId;	Blob analysis result buffer identifier or feature list
buffer identifier	

Description:

This function deletes the specified blob analysis result buffer or feature list and releases any memory associated with it. The BlobId parameter specifies the identifier of the blob analysis result buffer or feature list buffer to free. The buffer must have been successfully allocated, using MblobAllocResult() or MblobAllocFeatureList(), prior to calling this function.

MblobGetNumber

Format:

long MblobGetNumber(BlobResId, CountVarPtr)

MIL_ID BlobResId;	Blob analysis result buffer identifier
long *CountVarPtr;	Storage location for the count

Description:

This function reads the number of currently included blobs from the specified blob analysis result buffer. All blobs are included unless their status is changed, using MblobSelect(). Included blobs will be included in future operations and result retrievals.

This function must be used to determine the number of blob results that will be returned by MblobGetResult().

A call to MblobCalculate() must have been made prior to using this function.

MblobGetResult

Format:

void MblobGetResult(BlobResId, Feature, TargetArrayPtr)

MIL_ID BlobResId;	Blob analysis buffer identifier
long Feature;	Type of feature for which to get results
void *TargetArrayPtr;	Array in which to return results

Description:

This function obtains the results for a specified feature from the blob analysis result buffer.

The BlobResId parameter specifies the identifier of the blob analysis result buffer from which to get results.

The Feature parameter specifies the feature for which results will be retrieved. The specified feature must have already been calculated with MblobCalculate().

MblobReconstruct

Format:

void MblobReconstruct(SrcImageBufId, SeedImageBufId, DestImageBufId, Operation, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID SeedImageBufId;	Seed image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long Operation;	Type of operation to perform
long ProcMode;	Operation mode

Description:

This function copies (or reconstructs) blobs or blob holes from the source to the destination buffer, according to the type of operation specified. All non-zero pixels in the source buffer are considered to be part of a blob.

The Operation parameter specifies the type of operation to perform. In our application, M_FILL_HOLES is used.

MblobSelect

Format:

void MblobSelect(BlobResId, Operation, Feature, Condition, CondLow, CondHigh)

MIL_ID BlobResId;	Blob analysis result buffer identifier
long Operation;	Operation to perform on specified blobs
long Feature;	Feature to be used for selection
long Condition;	Conditional operator for selection
double CondLow;	Low compare value for the condition
double CondHigh;	High compare value for the condition

Description:

This function selects blobs that meet a specified criterion. These blobs will be included in or excluded from future operations (calculations or result retrieval), or deleted entirely from the result buffer.

If this function is not called at least once, all blobs are included by default.

If there is more than one call to this function, the effect of the calls is cumulative unless `M_INCLUDE_ONLY` or `M_EXCLUDE_ONLY` is specified as the operation to perform.

Once a blob has been excluded, it can normally be re-included only by specifying `M_INCLUDE` or `M_INCLUDE_ONLY` in a future call to this function (with the correct criterion). However, if you change the processing mode of a result buffer (with `MblobControl()`), or use the result buffer with different images (in a call to `MblobCalculate()`), all results in the buffer are discarded and all blobs are re-included.

MblobSelectFeature

Format:

```
void MblobSelectFeature(FeatureListId, Feature)
```

MIL_ID FeatureListId;	Feature list identifier
long Feature;	Feature to be selected

Description:

This function selects the feature(s) to be calculated by `MblobCalculate()` when using the specified feature list.

The `FeatureListId` parameter specifies the identifier of the feature list buffer.

The `Feature` parameter specifies the feature to add to the feature list. To select several features, you must call this function for each feature you want to add to the list (certain commonly used groups of features can be selected in a single call).

In our application `M_AREA`, `M_CENTER_OF_GRAVITY_X`, `M_CENTER_OF_GRAVITY_Y` are used as the `Feature` parameter.

MbufAllocColor

Format:

```
MIL_ID MbufAllocColor(SystemId, SizeBand, SizeX, SizeY, DataType,  
Attribute, BufIdPtr)
```

MIL_ID SystemId	System identifier
long SizeBand;	Number of color bands
long SizeX;	X dimension
long SizeY;	Y dimension
long DataType;	Pixel depth and range
long Attribute;	Buffer attributes
MIL_ID *BufIdPtr;	Storage location for buffer identifier

Description:

This function allocates a data buffer with multiple color bands on the specified system. This type of buffer allows the representation of color images (for example, RGB).

This function creates buffers that have a two-dimensional surface for each specified color band. You can use MbufAlloc1d and MbufAlloc2d to create single band one- or two-dimensional data buffers, respectively.

After allocating a buffer, we recommend that you check if the operation was successful, using MappGetError, or by verifying that the buffer identifier returned is not M_NULL.

MbufAlloc2d

Format:

MIL_ID MbufAlloc2d(SystemId, SizeX, SizeY, DataType, Attribute, BufIdPtr)

MIL_ID SystemId	System identifier
long SizeX;	X dimension
long SizeY;	Y dimension
long DataType;	Pixel depth and range
long Attribute;	Buffer attributes
MIL_ID *BufIdPtr;	Storage location for buffer identifier

Description:

This function allocates a two-dimensional data buffer on the specified system.

After allocating a buffer, we recommend that you check if the operation was successful, using MappGetError or by verifying that the buffer identifier returned is not M_NULL. When a buffer is no longer required, release it, using MbufFree.

MbufChild2d

Format:

MIL_ID MbufChild2d(ParentBufId, OffX, OffY, SizeX, SizeY, BufIdPtr)

MIL_ID ParentBufId;	Parent buffer identifier
long OffX;	X pixel offset relative to the parent buffer
long OffY;	Y pixel offset relative to the parent buffer
long SizeX;	Child buffer width
long SizeY;	Child buffer height
MIL_ID *BufIdPtr;	Storage location for child buffer identifier

Description:

This function allocates a child buffer from a two-dimensional region of all bands of the specified, previously allocated data buffer. The child buffer is not allocated its own memory space; it remains part of the parent buffer. Any modification to the child buffer affects the parent and vice versa. Note, a parent buffer can have several child buffers.

A child buffer is considered a data buffer in its own right, and can be used in the same circumstances as its parent buffer. A child buffer inherits its type and attributes from the parent buffer.

Note that when the parent buffer is multi-band, this function allocates a multi-band child buffer. The child is created from the specified region in each color band. To allocate a child region in one specific band, or specifically in all bands, use MbufChildColor2d instead of MbufChild2d.

When this buffer is no longer required, it can be released, using MbufFree.

MbufExport

Format:

void MbufExport(FileName, FileFormatBufId, SrcBufId)

char *FileName;	Destination file name
MIL_ID FileFormatBufId;	File format specification identifier
MIL_ID SrcBufId;	Source data buffer identifier

Description:

This function exports a data buffer to a file, using the specified output file format. Note, you can also save a buffer in an M_MIL file format, using MbufSave. The M_MIL file format is TIFF compatible.

MbufImport

Format:

MIL_ID MbufImport(FileName, FileFormatBufId, Operation, SystemId, BufIdPtr)

char *FileName;	Source file name
MIL_ID FileFormatBufId;	File format specification identifier
long Operation;	Import operation
MIL_ID SystemId;	System identifier
MIL_ID *BufIdPtr;	Buffer identifier (returned or given)

Description:

This function imports data, of the specified format, from a file into a MIL data buffer on the specified system.

Note, you can also import data that has been saved in an M_MIL file format, using MbufLoad or MbufRestore.

MbufPut2d

Format:

void MbufPut2d(DestBufId, OffX, OffY, SizeX, SizeY, UserArrayPtr)

MIL_ID DestBufId;	Destination buffer identifier
long OffX;	X pixel offset relative to destination buffer origin
long OffY;	Y pixel offset relative to the destination buffer origin
long SizeX;	Width of destination buffer area in which to put data
long SizeY;	Height of destination buffer area in which to put data
void *UserArrayPtr;	Source user array

Description:

This function copies data from a user-supplied array to a two-dimensional area of the specified MIL destination buffer.

MdispSelectWindow

Format:

void MdispSelectWindow(DisplayId, ImageBufId, ClientWindowHandle)

MIL_ID DisplayId;	Display identifier
MIL_ID ImageBufId;	Image buffer identifier

HWND ClientWindow Handle User-defined window handle

Description:

This function displays the specified image buffer contents in the specified user window, using the specified MIL display.

MgraColor

Format:

void MgraColor(GraphContId, ForegroundColor)

MIL_ID GraphContId; Graphics context identifier
double ForegroundColor; Foreground drawing and text color

Description:

This function sets the foreground color of a specified graphics context.

MgraFree

Format:

void MgraFree(GraphContId)

MIL_ID GraphContId; Graphics context identifier

Description:

This function deallocates a graphics context previously allocated with MgraAlloc.

MgraLine

Format:

void MgraLine(GraphcontId, DestImageBufId, XStart, YStart, XEnd, YEnd)

MIL_ID GraphContId; Graphics context identifier
MIL_ID DestImageBufId; Destination image buffer identifier
long XStart; X-coordinate of start of line position
long YStart; Y-coordinate of start of line position
long XEnd; X-coordinate of end of line position
long YEnd; Y-coordinate of end of line position

Description:

This function draws a line starting and ending at the specified coordinates, using the foreground color specified in the graphics context.

MimAllocResult

Format:

MIL_ID MimAllocResult(SystemId, NbEntries, ResultType, ImResultIdPtr)

MIL_ID SystemId;	System identifier
long NbEntries;	Number of result buffer entries
long ResultType;	Type of result buffer
MIL_ID *ImResultIdPtr;	Storage location for image processing result buffer identifier

Description:

This function allocates a result buffer with the specified number of entries, for use with the image processing module statistical functions.

MimBinarize

Format:

void MimBinarize(SrcImageBufId, DestImageBufId, Condition, CondLow, CondHigh)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long Condition;	Conditional operator for selection
double CondLow;	Low compare value for the condition
double CondHigh;	High compare value for the condition

Description:

This function performs binary thresholding on the specified image. Each pixel that meets the specified condition is set to the highest unsigned destination buffer value, while other pixels are set to 0. For example, the highest buffer value for an 8-bit buffer is 0xff.

MimClose

Format:

void MimClose(SrcImageBufId, DestImageBufId, NbIteration, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer
MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode;	Processing mode

Description:

This function performs a binary or grayscale closing operation on the given source image for the specified number of iterations. A closing is a dilation followed by an erosion.

MimConvert

Format:

MIL_ID MimConvert(SrcImageId, DestImageId, ConversionType)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long ConversionType;	Type of conversion to perform

Description:

This function performs a color conversion on the source image and places the result in the destination buffer.

MimConvolve

Format:

void MimConvolve(SrcImageBufId, DestImageBufId, KernelBufId)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
MIL_ID KernelBufId;	Kernel buffer identifier

Description:

This function performs a general convolution operation on the source buffer using the specified kernel, storing results in the specified destination buffer.

MimDilate

Format:

void MimDilate(SrcImageBufId, DestImageBufId, NbIteration, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode;	Processing mode

Description:

This function performs a binary or grayscale dilation on the given source image for the specified number of iterations.

MimEdgeDetect

Format:

void MimEdgeDetect(SrcImageBufId, DestIntensityImageBufId, DestAngleImageBufId, KernelId, ControlFlag, Threshold)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestIntensityImageBufId;	Destination gradient intensity image buffer identifier
MIL_ID DestAngleImageBufId;	Destination gradient angle image buffer identifier
MIL_ID KernelId;	Kernel identifier
long ControlFlag;	Flag to control operation
long Threshold;	Threshold value for gradient intensity

Description:

This function performs an edge detection operation on the specified source image, using the specified kernel. It produces a gradient intensity image and/or a gradient angle image in the specified image buffer(s). If one of the destination images is not required, specify M_NULL as its image buffer identifier.

MimErode

Format:

void MimErode(SrcImageBufId, DestImageBufId, NbIteration, Procmode)

MIL_ID SrcImageBufId;	Source image buffer identifier
-----------------------	--------------------------------

MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode;	Processing mode

Description:

This function performs a binary or grayscale erosion on the given source image for the specified number of iterations.

MimFlip

Format:

void MimFlip(SrcImageId, DestImageId, Operation, OpFlag)

MIL_ID SrcImageId;	Source image buffer identifier
MIL_ID DestImageId;	Destination image buffer identifier
long Operation;	Operation to perform
long OpFlag;	Flag for the operation

Description:

This function performs a flipping of a source image to a destination image according to the specified operation.

The Operation parameter specifies the operation to perform. This parameter can be set to one of the following:

M_FLIP_HORIZONTAL Flip the image in a horizontal direction (left to right, along a vertical axis).

M_FLIP_VERTICAL Flip the image in a vertical direction (top to bottom, along a horizontal axis).

MimFree

Format:

void MimFree(ImResultId)

MIL_ID ImResultId;	Image processing result buffer identifier
--------------------	---

Description:

This function deallocates a result buffer previously allocated with MimAllocResult.

MimGetResult

Format:

void MimGetResult(ImResultId, ResultType, UserArrayPtr)

MIL_ID ImResultId;	Image processing result buffer identifier
long ResultType;	Type of result to read
void *UserArrayPtr;	Array in which to return results

Description:

This function copies all the results from the specified result buffer to the specified one-dimensional destination user array.

MimHistogram

Format:

void MimHistogram(SrcImageBufId, HistImResultId)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID HistImResultId;	Histogram image processing result buffer identifier

Description:

This function calculates the histogram (or pixel intensity distribution) of the specified source buffer and stores results in the specified histogram result buffer.

We can read the histogram values from the result buffer, using MimGetResultId or MimGetResult, specifying M_VALUE as the result type.

MimOpen

Format:

void MimOpen(SrcImageBufId, DestImageBufId, NbIteration, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode;	Processing mode

Description:

This function performs a binary or grayscale opening operation on the given source image for the specified number of iterations. An opening is an erosion followed by a dilation.

MimThick

Format:

void MimThick(SrcImageBufId, DestImageBufId, NbIteration, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode;	Processing mode

Description:

This function performs a binary or grayscale thickening on the specified source image for the specified number of iterations.

MimThin

Format:

void MimThin(SrcImageBufId, DestImageBufId, NbIteration, ProcMode)

MIL_ID SrcImageBufId;	Source image buffer identifier
MIL_ID DestImageBufId;	Destination image buffer identifier
long NbIteration;	Number of operation iterations
long ProcMode	Processing mode

Description:

This function performs a binary or grayscale thinning on the specified source image for the specified number of iterations.

APPENDIX B

THE USER'S GUIDE OF THE PROGRAM

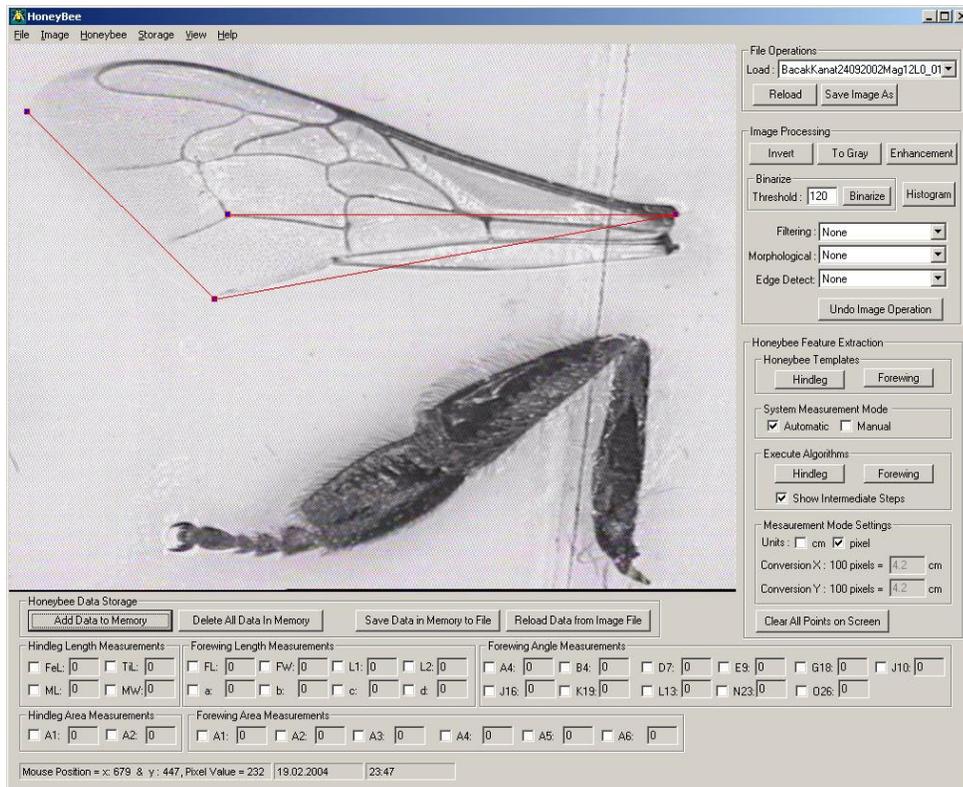


Figure B.1 - GUI of Honeybee Feature Extraction System

The GUI of the proposed system is given above. Each utilities accomplished by the program are explained in the following:

File Operations:

Load: Loads the selected image from the “Images” directory of the system. “Images” directory should be created in the upper directory of the executable.

Reload: Reloads the current image.

Save Image As: Saves the current image to the hard disk.

“Reload” and “Save Image As” can also be executed from the “File” menu.

Image Processing Operations:

Undo Image Operation: Loads the image just before the last image operation is executed. Undo operation caches the last three operations at most.

Invert: Inverts the current image. It replaces each pixel with its counterpart.

To Gray: Makes Color-to-Gray conversion on the current image.

Enhancement: Applies “brightness” and “contrast” image enhancement operations to the current image.

Binarize: Binarizes the current image with a given “Threshold” value. Before executing “Binarize” operation a suitable value for “Threshold” should be entered or default value of 120 is used. That is, for each pixel on the current image; if the pixel is lower than the given threshold, replaces it with 255, else replaces it with 0.

Histogram: Shows the histogram of the current image. Before using “Histogram”, “Smoothing” from the “Filtering” operations should be applied for the images. In this way, the fluctuations in the histogram are eliminated and a smoother histogram is obtained. The window for the histogram operation is shown in Figure B.2.

The region of the image for which the histogram is calculated is the whole image by default. However, by entering new (X1, Y1) and (X2, Y2) coordinates, it is possible to re-define the region on which histogram calculation is required. (X1, Y1) defines the top left corner of new the region and (X2, Y2) is the lower right corner of the new region. To draw the new histogram of the user specified region, “Update” button must be pressed.

The maximum value of the “Freq” axis of the histogram can be changed by entering the top value and checking the “Limit Freq” check box. Histogram will be automatically updated, when “Limit Freq” is checked.

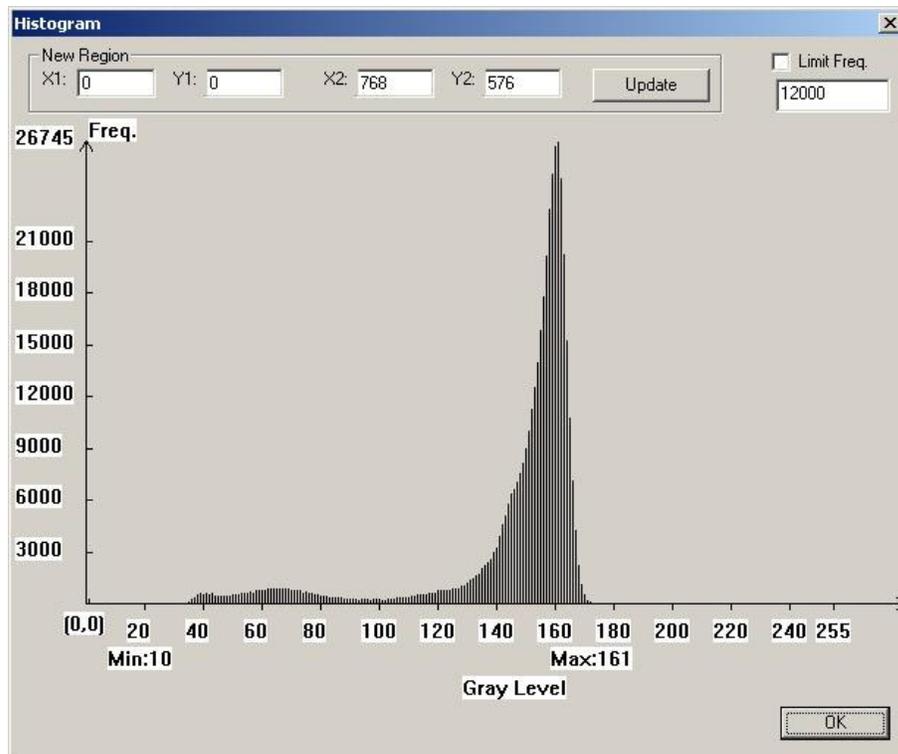


Figure B.2 - Histogram window

Filtering: Applies the following filtering operations on the current image:

- Smoothing
- Sharping
- More Sharping

Morphological: Applies the following morphological operations on the current image:

- Erode
- Dilate
- Open
- Close
- Thick
- Thin
- Thick to Idempotence
- Thin to Skeletons

Edge Detect: Detects the edges on the current image with the following methods:

- Horizontal Edge Detection
- Vertical Edge Detection
- Edge Detection
- More Edge Detection

“Undo”, “To Gray”, “Enhancement”, “Invert” and “Histogram” can be also executed from the “Image” menu.

Honeybee Feature Extraction Operations:

Honeybee Templates:

Hindleg: Shows the hindleg template on a new window. The reference points that should be entered by the user on the image in Automatic mode are displayed on this hindleg template. The window for hindleg template is shown below.

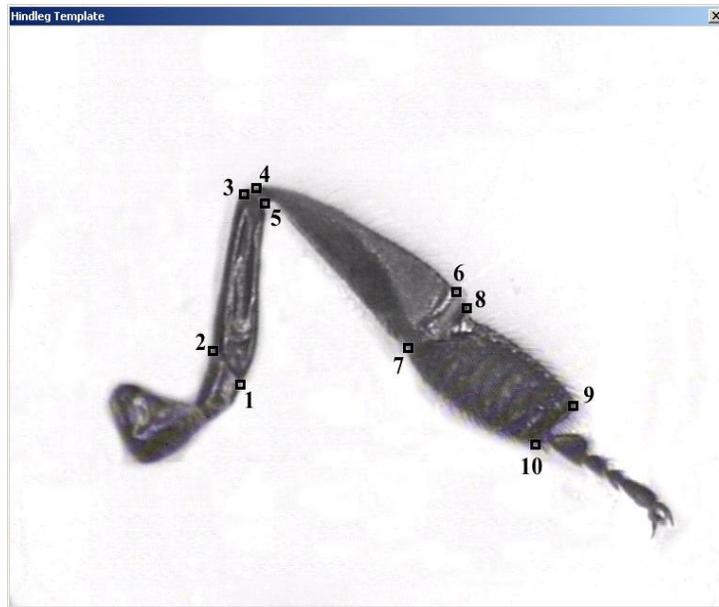


Figure B.3 - The hindleg template

Forewing: Shows the forewing template on a new window. The reference points that should be entered by the user on the image in Automatic mode are displayed on this forewing template. The window for forewing template is shown below.

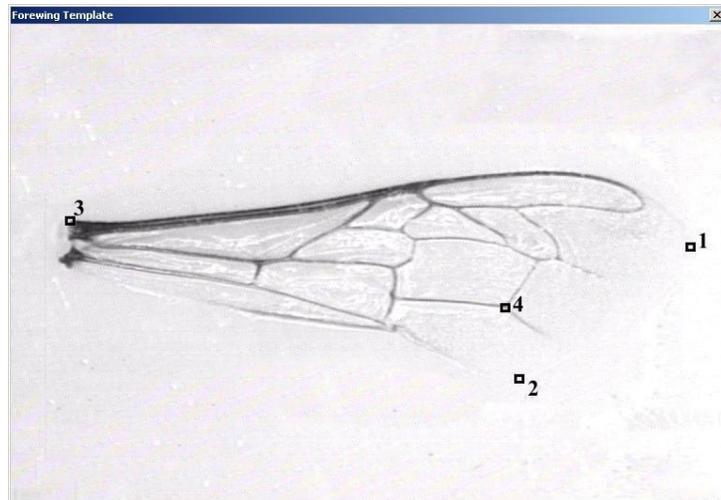


Figure B.4 - The forewing template

System Measurement Mode:

Automatic: Selects the system mode as Automatic, if this checkbox is checked. When the “Automatic” mode is active, the “Execute Algorithms” section is enabled.

Manual: Selects the system mode as Manual, if “Manual” check box is checked. If this mode is selected, the “Execute Algorithms” section is disabled. In this mode, user can measure angle and length features of the honeybee manually. For length measurements, the user must enter two points on the image by double clicking the left mouse

button and must select the appropriate feature check box in “Hindleg Length Measurements” or “Forewing Length Measurements”. For angle measurements, the user must enter three points on the image by double clicking the left mouse button and must select the appropriate feature check box in “Forewing Angle Measurements”.

The points entered by the user can be deleted by double clicking the entered point. A message box window is displayed to confirm the delete request.

Execute Algorithms:

Hindleg: Executes the algorithms for extracting the hindleg features in “Automatic Mode”. Before using this operation, the reference points shown in hindleg template (Honeybee Templates → Hindleg) should be entered on the hindleg image by the user.

Forewing: Executes the algorithms for extracting the forewing features in “Automatic Mode”. Before using this operation, the reference points shown in hindleg template (Honeybee Templates → Forewing) should be entered on the forewing image by the user.

The “Hindleg” and “Forewing” operations are activated only in Automatic mode of the system. The reference points used in Automatic mode are entered by a single click of left mouse button. The place of the reference points can be changed to another place on the image simply by dragging and dropping the points.

Show Intermediate Steps: If this is checked, the intermediate steps of the forewing or hindleg feature extraction algorithms are also shown.

After executing Hindleg or Forewing algorithms, the length and angle feature measurements are displayed automatically in “Hindleg Length Measurements” or “Forewing Length Measurements and Forewing Angle Measurements” fields respectively.

Measurement Mode Settings:

Units: The length measurements are handled in two units: pixels and cm. One of these units can be enabled by checking “pixel” or “cm” check boxes.

Conversion X: If “cm” is checked as the length unit, then the user should enter the equivalence of 100 pixels in cm from the conversion edit box for x-axis. The default value of this edit box is 4.2 cm.

Conversion Y: If “cm” is checked as the length unit, then the user should enter the equivalence of 100 pixels in cm from the conversion edit box for y-axis. The default value of this edit box is 4.2 cm.

Clear All Points on Screen: Clears all the reference points entered by the user and the template contour points drawn by executing “Execute Algorithms→Hindleg or Execute Algorithms→Forewing” operations.

“System Measurement Mode”, “Show Hindleg Templates”, “Show Forewing Templates”, “Automatic Feature Extraction” and “Clear All Points on Screen” can be also executed from the “Honeybee” menu.

Honeybee Data Storage Operations:

Add Data to Memory: Adds the measurement results displayed in “Hindleg Length Measurements” or “Forewing Length Measurements” and “Forewing Angle Measurements” sections, when measurements are handled in Automatic or Manual mode.

Delete All Data In Memory: Deletes the measurement results in the memory that are added by “Add Data to Memory” operation. This operation is disabled when no data is added to the memory.

Save Data in Memory to File: Saves all the measurement results in memory to a file in hard disk with extension “all”. The name of that file is given by user. For each hindleg and forewing measurements, separate files are created having the measurement result with the name same as image name and extension is “leg” for hindleg and “wng” for forewing measurements.

Reload Data from Image File: Reloads the measurement data and related forewing or hindleg image from the chosen “*.leg” or “*.wng” files. The data is displayed in “Hindleg Length Measurements” or “Forewing Length Measurements” and “Forewing Angle Measurements” sections and the image having that measured data is displayed on the screen.

“Add Data to Memory”, “Delete All Data In Memory”, “Save Data in Memory to File”, “Reload Data from Image File” can be also executed from the “Storage” menu.

Hindleg Length Measurements: This section shows the automatic or manual length measurements for the hindleg features.

Hindleg Area Measurements: This section shows the automatic area measurements for the hindleg.

Forewing Length Measurements: This section shows the automatic or manual length measurements for the forewing features.

Forewing Angle Measurements: This section shows the automatic or manual angle measurements for the forewing features.

Forewing Area Measurements: This section shows the automatic area measurements for the forewing.

Any one of the feature measurements can be repeated and changed in manual mode which is explained in Manual operation of System Measurement Mode.

By use of “Bee Measurements” in “View” menu, the “Hindleg Length Measurements” or “Forewing Length Measurements” and “Forewing Angle Measurements” sections can be hidden or displayed on the GUI.

The software can be used best with minimum 1152x864 resolutions. The application has been developed with Visual C++ 6.0 in Microsoft Windows 2000 but it is also tested in Windows 98/ME and Windows XP operating systems. The minimum system requirements for using this program are 128 MB RAM, Pentium 300 MHz or higher. The redistributable Matrox Imaging Library should also be installed before executing the program.

APPENDIX C

ALGORITHMS

In this appendix, “Vertex Point Finding Algorithm” is given for finding the intersection point of three edges (i.e. vertex point). This algorithm is used after applying the snake algorithm to the forewing images to correct the places of the critical points.

Algorithm C.1 - Vertex Point Finding Algorithm

Specify a neighborhood of the critical point for which search of vertex point is required. This neighborhood is named as δ_{Big} .

Specify a second neighborhood, which determines the borders on which the vertex point will be searched. This secondary neighborhood is named as δ_{Border} .

Construct the dynamic array –namely “arrayCandidates”-, which will be used to store the candidate vertex points.

for all pixel coordinates (x, y) in δ_{Big} do

*//Find all vertex points on the border of the neighborhood δ_{Border} taking the
//center point as (x, y) . This function checks the border of δ_{Border} . If there
//exists edge points such that they are all connected neighbours on the
//border, they are counted as 1 single edge point. After all checks are done,*

```

//if the total number of edge points is 3 (i.e. (x,y) is a vertex point),
//function returns TRUE.
if (FindVertex_CheckSinglePoint(x,y, NeighbourWidth of  $\delta_{Border}$ , NeighbourHeight of
 $\delta_{Border}$ ) == TRUE)
{
    //Add the vertex points to arrayCandidates.
    arrayCandidates.Add( (x,y));
}
end for

//Now we have candidate points recorded in "arrayCandidates" array.
//So, the next step is to decrement the neighbourhood and check if those //candidates still satisfy
our criteria!
int i;
BOOL bContinueDecrementing = TRUE;
while(bContinueDecrementing) {
    //The following "if" structure actually determines the new values of //our neighbourhood.
    if(NeighbourWidth of  $\delta_{Border}$  > 5) {
        NeighbourWidth of  $\delta_{Border}$  -= 2;
        if(NeighbourHeight of  $\delta_{Border}$  > 5) {
            NeighbourHeight of  $\delta_{Border}$  =  $\delta_{Border}$  - 2;
        }
    }
    else {
        if(NeighbourHeight of  $\delta_{Border}$  > 5) {
            NeighbourHeight of  $\delta_{Border}$  =  $\delta_{Border}$  - 2;
        }
        else {
            //We cannot decrement any more! So quit!
            bContinueDecrementing = FALSE;
            break;
        }
    }
}
//Now we have the "NEW" neighbourhood ranges.
//Check the candidate points for these "NEW" ranges!
for(i=0; i<arrayCandidates.GetSize(); i++)
{
    if(FindVertex_CheckSinglePoint(x,y, NeighbourWidth of  $\delta_{Border}$ ,
    NeighbourHeight of  $\delta_{Border}$ ) == FALSE)
    {
        arrayCandidates.RemoveAt(i); //Remove this point!
        i--; //We need to decrement since we had removed
        //an element within the array!
    }
}

```

```
        }  
    }  
} // End of while(bContinueDecrementing)  
  
//Check if we have found the vertex points  
if(arrayCandidates.GetSize() > 0) {  
    //get the first element (x,y) in the arrayCandidate.  
    return TRUE;  
}  
else {  
    return FALSE; // No vertex point is found  
}
```